



Informatica (Versión 9.6.0)

# Guía de transformación para desarrolladores

Copyright (c) 2009-2013 Informatica Corporation. Todos los derechos reservados.

Este software y esta documentación contienen información privada de Informatica Corporation y se facilitan bajo un acuerdo de licencia que incluye restricciones sobre el uso y la divulgación y también están protegidos por leyes de copyright. Está prohibida la ingeniería inversa del software. Ninguna parte de este documento puede ser reproducida o transmitida de cualquier forma o manera (electrónica, fotocopia, grabación o mediante otros métodos) sin el consentimiento previo de Informatica Corporation. Este Software puede estar protegido por Patentes de los Estados Unidos y/o patentes internacionales y otras patentes pendientes.

El uso, la duplicación o la divulgación del software por parte del gobierno de los Estados Unidos están sujetos a las restricciones puntualizadas en el acuerdo de licencia de software vigente y según se establece en la reglamentación DFARS 227.7202-1(a) y 227.7702-3(a) (1995), DFARS 252.227-7013<sup>(1)</sup>(ii) (OCT 1988), FAR 12.212(a) (1995), FAR 52.227-19 o FAR 52.227-14 (ALT III), según proceda.

La información contenida en este producto o documentación está sujeta a cambios sin previo aviso. Si encuentra algún problema en este producto o documentación, infórmenos por escrito.

Informatica, Informatica Platform, Informatica Data Services, PowerCenter, PowerCenterRT, PowerCenter Connect, PowerCenter Data Analyzer, PowerExchange, PowerMart, Metadata Manager, Informatica Data Quality, Informatica Data Explorer, Informatica B2B Data Transformation, Informatica B2B Data Exchange Informatica On Demand, Informatica Identity Resolution, Informatica Application Information Lifecycle Management, Informatica Complex Event Processing, Ultra Messaging e Informatica Master Data Management son marcas comerciales o marcas comerciales registradas de Informatica Corporation en Estados Unidos y en las jurisdicciones de todo el mundo. Los demás nombres de productos y empresas pueden ser nombres o marcas comerciales de sus respectivos titulares.

Hay fragmentos de este software y/o documentación que están sujetos a copyright perteneciente a terceros, incluido, entre otros: Copyright DataDirect Technologies. Todos los derechos reservados. Copyright © Sun Microsystems. Todos los derechos reservados. Copyright © RSA Security Inc. Todos los derechos reservados. Copyright © Ordinal Technology Corp. Todos los derechos reservados. Copyright © Aandacht c.v. Todos los derechos reservados. Copyright Genivia, Inc. Todos los derechos reservados. Copyright Isomorphic Software. Todos los derechos reservados. Copyright © Meta Integration Technology, Inc. Todos los derechos reservados. Copyright © Intalio. Todos los derechos reservados. Copyright © Oracle. Todos los derechos reservados. Copyright © Adobe Systems Incorporated. Todos los derechos reservados. Copyright © DataArt, Inc. Todos los derechos reservados. Copyright © ComponentSource. Todos los derechos reservados. Copyright © Microsoft Corporation. Todos los derechos reservados. Copyright © Rogue Wave Software, Inc. Todos los derechos reservados. Copyright © Teradata Corporation. Todos los derechos reservados. Copyright © Yahoo! Inc. Todos los derechos reservados. Copyright © Glyph & Cog, LLC. Todos los derechos reservados. Copyright © Thinkmap, Inc. Todos los derechos reservados. Copyright © Clearpace Software Limited. Todos los derechos reservados. Copyright © Information Builders, Inc. Todos los derechos reservados. Copyright © OSS Nokalva, Inc. Todos los derechos reservados. Copyright Edifecs, Inc. Todos los derechos reservados. Copyright Cleo Communications, Inc. Todos los derechos reservados. Copyright © International Organization for Standardization 1986. Todos los derechos reservados. Copyright © ej-technologies GmbH. Todos los derechos reservados. Copyright © Jaspersoft Corporation. Todos los derechos reservados. Copyright © International Business Machines Corporation. Todos los derechos reservados. Copyright © yWorks GmbH. Todos los derechos reservados. Copyright © Lucent Technologies. Todos los derechos reservados. Copyright © University of Toronto. Todos los derechos reservados. Copyright © Daniel Veillard. Todos los derechos reservados. Copyright © Unicode, Inc. Copyright IBM Corp. Todos los derechos reservados. Copyright © MicroQuill Software Publishing, Inc. Todos los derechos reservados. Copyright © PassMark Software Pty Ltd. Todos los derechos reservados. Copyright © LogiXML, Inc. Todos los derechos reservados. Copyright © 2003-2010 Lorenzi Davide. Todos los derechos reservados. Copyright © Red Hat, Inc. Todos los derechos reservados. Copyright © The Board of Trustees of the Leland Stanford Junior University. Todos los derechos reservados. Copyright © EMC Corporation. Todos los derechos reservados. Copyright © Flexera Software. Todos los derechos reservados. Copyright © Jinfonet Software. Todos los derechos reservados. Copyright © Apple Inc. Todos los derechos reservados. Copyright © Telerik Inc. Todos los derechos reservados. Copyright © BEA Systems. Todos los derechos reservados. Copyright © PDFlib GmbH. Todos los derechos reservados. Copyright © Orientation in Objects GmbH. Todos los derechos reservados. Copyright © Tanuki Software, Ltd. Todos los derechos reservados. Copyright © Ricebridge. Todos los derechos reservados. Copyright © Sencha, Inc. Todos los derechos reservados.

Este producto incluye software desarrollado por la Apache Software Foundation (<http://www.apache.org/>) y/u otro software protegido por varias versiones de la licencia Apache License ("Licencia"). Puede obtener una copia de estas licencias en <http://www.apache.org/licenses/>. A menos que las leyes aplicables lo requieran o se haya acordado por escrito, el software distribuido bajo estas licencias se distribuye "TAL CUAL", SIN GARANTÍAS NI CONDICIONES DE NINGÚN TIPO, ya sea expresas o implícitas. Consulte las licencias del idioma específico para conocer los permisos y las limitaciones que rigen según las licencias.

Este producto incluye software desarrollado por Mozilla (<http://www.mozilla.org/>), copyright del software de The JBoss Group, LLC, todos los derechos reservados; copyright del software © 1999-2006 de Bruno Lowagie y Paulo Soares y otro software protegido con licencia por el acuerdo GNU Lesser General Public License Agreement, que se puede encontrar en la dirección <http://www.gnu.org/licenses/lgpl.html>. Los materiales se facilitan gratuitamente por parte de Informatica, "tal cual", sin garantía de ningún tipo, ya sea expresa o implícita, incluidas, entre otras, las garantías implícitas de adecuación para un propósito determinado y de validez para el comercio.

El producto incluye software ACE(TM) y TAO(TM) con copyright de Douglas C. Schmidt y su grupo de investigación de la Washington University, University of California, Irvine y Vanderbilt University, Copyright (©) 1993-2006, todos los derechos reservados.

Este producto incluye software desarrollado por el OpenSSL Project para uso en el OpenSSL Toolkit (copyright The OpenSSL Project. Todos los derechos reservados) y la redistribución de este software está sujeta a los términos especificados en <http://www.openssl.org> y <http://www.openssl.org/source/license.html>.

Este producto incluye software Curl con Copyright 1996-2013, Daniel Stenberg, <daniel@haxx.se>. Todos los derechos reservados. Los permisos y las limitaciones relativos a este software están sujetos a los términos disponibles en la dirección <http://curl.haxx.se/docs/copyright.html>. La autorización para utilizar, copiar, modificar y distribuir este software para cualquier propósito con o sin tasas se concede por el presente, siempre que el aviso de copyright anterior y este aviso de permiso aparezcan en todas las copias.

El producto incluye copyright de software 2001-2005 (©) MetaStuff, Ltd. Todos los derechos reservados. Los permisos y las limitaciones relativos a este software están sujetos a los términos disponibles en la dirección <http://www.dom4j.org/license.html>.

El producto incluye copyright de software © 2004-2007, The Dojo Foundation. Todos los derechos reservados. Los permisos y las limitaciones relativos a este software están sujetos a los términos disponibles en la dirección <http://dojotoolkit.org/license>.

Este producto incluye software ICU con copyright de International Business Machines Corporation y otros. Todos los derechos reservados. Los permisos y las limitaciones relativos a este software están sujetos a los términos disponibles en la dirección <http://source.icu-project.org/repos/icu/icu/trunk/license.html>.

Este producto incluye copyright de software © 1996-2006 Per Bothner. Todos los derechos reservados. Su derecho a utilizar estos materiales está establecido en la licencia que puede encontrarse en la dirección <http://www.gnu.org/software/kawa/Software-License.html>.

Este producto incluye software OSSP UUID con Copyright © 2002 Ralf S. Engelschall, Copyright © 2002 The OSSP Project Copyright © 2002 Cable & Wireless Deutschland. Los permisos y las limitaciones relativos a este software están sujetos a los términos disponibles en la dirección <http://www.opensource.org/licenses/mit-license.php>.

Este producto incluye software desarrollado por Boost (<http://www.boost.org/>) o protegido por la licencia de software de Boost. Los permisos y las limitaciones relativos a este software están sujetos a los términos disponibles en la dirección [http://www.boost.org/LICENSE\\_1\\_0.txt](http://www.boost.org/LICENSE_1_0.txt).

Este producto incluye copyright de software © 1997-2007 University of Cambridge. Los permisos y las limitaciones relativos a este software están sujetos a los términos disponibles en la dirección <http://www.pcre.org/license.txt>.

Este producto incluye copyright de software © 2007 The Eclipse Foundation. Todos los derechos reservados. Los permisos y las limitaciones relativos a este software están sujetos a los términos especificados en <http://www.eclipse.org/org/documents/epl-v10.php> y <http://www.eclipse.org/org/documents/edl-v10.php>.

Este producto incluye software protegido por licencia según los términos que aparecen en <http://www.tcl.tk/software/tcltk/license.html>, <http://www.bosrup.com/web/overlib?License>, <http://www.stlport.org/doc/license.html>, <http://asm.ow2.org/license.html>, <http://www.cryptix.org/LICENSE.TXT>, <http://hsqldb.org/web/hsqldbLicense.html>, <http://httpunit.sourceforge.net/doc/license.html>, <http://jung.sourceforge.net/license.txt>, [http://www.gzip.org/zlib/zlib\\_license.html](http://www.gzip.org/zlib/zlib_license.html), <http://www.openldap.org/software/release/license.html>, <http://www.libssh2.org>, <http://slf4j.org/license.html>, <http://www.sente.ch/software/OpenSourceLicense.html>, <http://fusesource.com/downloads/license-agreements/fuse-message-broker-v-5-3-license-agreement>, <http://antlr.org/license.html>, <http://aopalliance.sourceforge.net/>, <http://www.bouncycastle.org/license.html>, <http://www.jgraph.com/jgraphdownload.html>, <http://www.jcraft.com/jsch/LICENSE.txt>, [http://jotm.objectweb.org/bsd\\_license.html](http://jotm.objectweb.org/bsd_license.html), <http://www.w3.org/Consortium/Legal/2002/copyright-software-20021231>, <http://www.sl4j.org/license.html>, <http://nanoxml.sourceforge.net/orig/copyright.html>, <http://www.json.org/>, <http://forge.ow2.org/projects/javaservice/>, <http://www.postgresql.org/about/license.html>, <http://www.sqlite.org/copyright.html>, <http://www.tcl.tk/software/tcltk/license.html>, <http://www.jaxen.org/faq.html>, <http://www.jdom.org/docs/faq.html>, <http://www.sl4j.org/license.html>, <http://www.iodbc.org/dataspace/iodbc/wiki/IODBC/License>, <http://www.keplerproject.org/md5/license.html>, <http://www.toedter.com/en/calendar/license.html>, <http://www.edankert.com/bounce/index.html>, <http://www.net-snmp.org/about/license.html>, <http://www.openmdx.org/#FAQ>, [http://www.php.net/license/3\\_01.txt](http://www.php.net/license/3_01.txt), <http://srp.stanford.edu/license.txt>, <http://www.schneier.com/blowfish.html>, <http://www.jmock.org/license.html>, <http://xsom.java.net>, <http://benalman.com/about/license/>, <https://github.com/CreateJS/EaselJS/blob/master/src/easeljs/display/Bitmap.js>, <http://www.h2database.com/html/license.html#summary>, <http://jsoncpp.sourceforge.net/LICENSE>, <http://jdbc.postgresql.org/license.html>, <http://protobuf.googlecode.com/svn/trunk/src/google/protobuf/descriptor.proto>, <https://github.com/rantav/hector/blob/master/LICENSE>, <http://web.mit.edu/Kerberos/krb5-current/doc/mitK5license.html> y <http://jibx.sourceforge.net/jibx-license.html>.

Este producto incluye software desarrollado por la Academic Free License (<http://www.opensource.org/licenses/afl-3.0.php>), la Common Development and Distribution License (<http://www.opensource.org/licenses/cddl1.php>), la Common Public License (<http://www.opensource.org/licenses/cpl1.0.php>), la Sun Binary Code License Agreement Supplemental License Terms, la BSD License (<http://www.opensource.org/licenses/bsd-license.php>), la nueva BSD License (<http://opensource.org/licenses/BSD-3-Clause>), la MIT License (<http://www.opensource.org/licenses/mit-license.php>), la Artistic License (<http://www.opensource.org/licenses/artistic-license-1.0>) y la Initial Developer's Public License Version 1.0 (<http://www.firebirdsql.org/en/initial-developer-s-public-license-version-1-0/>).

Este producto incluye copyright de software © 2003-2006 Joe Walnes, 2006-2007 XStream Committers. Todos los derechos reservados. Los permisos y las limitaciones relativos a este software están sujetos a los términos disponibles en la dirección <http://xstream.codehaus.org/license.html>. Este producto incluye software desarrollado por Indiana University Extreme! Lab. Para obtener más información, visite <http://www.extreme.indiana.edu/>.

Este producto incluye software Copyright © 2013 Frank Balluffi y Markus Moeller. Todos los derechos reservados. Los permisos y las limitaciones relativas a este software están sujetos a los términos de la licencia MIT.

Este Software está protegido por los números de patentes de Estados Unidos 5.794.246, 6.014.670, 6.016.501, 6.029.178, 6.032.158, 6.035.307, 6.044.374, 6.092.086, 6.208.990, 6.339.775, 6.640.226, 6.789.096, 6.823.373, 6.850.947, 6.895.471, 7.117.215, 7.162.643, 7.243.110, 7.254.590, 7.281.001, 7.421.458, 7.496.588, 7.523.121, 7.584.422, 7.676.516, 7.720.842, 7.721.270, 7.774.791, 8.065.266, 8.150.803, 8.166.048, 8.166.071, 8.200.622, 8.224.873, 8.271.477, 8.327.419, 8.386.435, 8.392.460, 8.453.159, 8.458.230 y RE44.478, patentes internacionales y otras patentes pendientes.

**EXENCIÓN DE RESPONSABILIDAD:** Informatica Corporation proporciona esta documentación "tal cual" sin garantía de ningún tipo, ya sea expresa o implícita, incluidas, entre otras, las garantías implícitas de no incumplimiento, de adecuación para un propósito determinado y de validez para el comercio. Informatica Corporation no garantiza que este software o esta documentación estén libres de errores. La información proporcionada en este software o en esta documentación puede contener imprecisiones técnicas o errores tipográficos. La información de este software y esta documentación está sujeta a cambios en cualquier momento sin previo aviso.

#### AVISOS

Este producto de Informatica (el "Software") incluye ciertos controladores (los "Controladores DataDirect") de DataDirect Technologies, una empresa operativa de Progress Software Corporation ("DataDirect") que están sujetos a los términos y condiciones siguientes:

1. LOS CONTROLADORES DATADIRECT SE PROPORCIONAN "TAL CUAL" SIN GARANTÍA DE NINGÚN TIPO, YA SEA EXPRESA O IMPLÍCITA, INCLUIDAS, ENTRE OTRAS, LAS GARANTÍAS IMPLÍCITAS DE NO INCUMPLIMIENTO, DE ADECUACIÓN PARA UN PROPÓSITO DETERMINADO Y DE VALIDEZ PARA EL COMERCIO.
2. EN NINGÚN CASO DATADIRECT NI SUS PROVEEDORES DE TERCEROS SERÁN RESPONSABLES ANTE EL USUARIO FINAL POR NINGÚN DAÑO DIRECTO, INDIRECTO, FORTUITO, ESPECIAL, CONSECUENTE, NI DE NINGÚN OTRO TIPO, RESULTANTE DEL USO DE LOS CONTROLADORES ODBC, INDEPENDIENTEMENTE DE SI SE HA AVISADO O NO DE LOS POSIBLES DAÑOS POR ADELANTADO. ESTAS LIMITACIONES SE APLICAN A TODAS LAS DEMANDAS JUDICIALES, INCLUIDAS, ENTRE OTRAS, AQUELLAS POR INCUMPLIMIENTO DE CONTRATO, INCUMPLIMIENTO DE LA GARANTÍA, NEGLIGENCIA, RESPONSABILIDAD ESTRICTA, TERGIVERSACIÓN Y OTROS AGRAVIOS.

Número de parte: IN-DTG-96000-0001

# Tabla de contenido

<b>Prefacio .....</b>	<b>xvii</b>
Documentación de Informatica .....	xvii
Portal de atención al cliente (My Support) de Informatica. ....	xvii
Documentación de Informatica .....	xvii
Sitio web de Informatica .....	xvii
Biblioteca de asistencia de Informatica .....	xviii
Base de conocimiento de Informatica. ....	xviii
Canal de YouTube de atención al cliente de Informatica .....	xviii
Catálogo de soluciones de Informatica. ....	xviii
Informatica Velocity. ....	xviii
Servicio internacional de atención al cliente de Informatica. ....	xviii
 <b>Capítulo 1: Introducción a las transformaciones.....</b>	<b>1</b>
Introducción al resumen de las transformaciones. ....	1
Transformaciones activas. ....	1
Transformaciones pasivas. ....	2
Transformaciones no conectadas. ....	2
Transformaciones multiestrategia. ....	2
Descripciones de transformaciones. ....	3
Cómo desarrollar una transformación. ....	5
Transformaciones reutilizables. ....	5
Instancias de transformación reutilizable y cambios heredados. ....	6
Cómo editar una transformación reutilizable. ....	6
Expresiones de las transformaciones. ....	6
El editor de expresiones. ....	8
Nombres de puerto en una expresión. ....	8
Cómo añadir una expresión a un puerto. ....	8
Comentarios en una expresión. ....	9
Validación de expresiones. ....	9
Cómo crear una transformación. ....	9
Niveles de seguimiento. ....	9
 <b>Capítulo 2: Transformación del validador de direcciones.....</b>	<b>11</b>
Resumen de la transformación del validador de direcciones. ....	11
Datos de referencia de direcciones. ....	12
Tipos de datos de referencia de direcciones. ....	12
Grupos de puertos y selección de puertos. ....	14
Plantillas. ....	14
Grupos de puertos de entrada de la transformación del validador de direcciones. ....	15
Grupos de puertos de salida de transformación del validador de direcciones. ....	15

Puertos de varias instancias. . . . .	17
Proyectos de validación de direcciones. . . . .	18
Direcciones con formato y estándares del proveedor de servicios de correo. . . . .	19
Puertos de estado del validador de direcciones. . . . .	20
Definiciones de código de estado de elemento. . . . .	21
Valores del puerto de salida del código de resolución de direcciones . . . . .	24
Valores del puerto de salida del estado de entrada de elementos. . . . .	24
Valores del puerto de salida de relevancia de elementos. . . . .	25
Valores del puerto de salida del estado de resultado de elementos. . . . .	25
Valores del puerto de salida del estado de elementos extendido. . . . .	27
Valores del puerto de salida de la puntuación de viabilidad de envío de correo. . . . .	28
Valores del puerto de salida del código de coincidencia. . . . .	29
Valores del puerto de salida del estado de geocodificación. . . . .	30
Estado del archivo de datos de referencia de direcciones. . . . .	31
Ajustes generales de la transformación del validador de direcciones. . . . .	32
Modo de lista de sugerencias. . . . .	33
Propiedades avanzadas de la transformación del validador de direcciones. . . . .	33
Alias de calle. . . . .	33
Uso de mayúsculas y minúsculas. . . . .	34
País de origen. . . . .	34
Tipo de país. . . . .	35
País predeterminado. . . . .	36
Prioridad de dirección dual. . . . .	36
Abreviatura de elemento. . . . .	36
Instancias de ejecución. . . . .	37
Expansión de rango flexible. . . . .	37
Tipo de datos de geocodificación. . . . .	37
Longitud de campo máxima global. . . . .	38
Tipo de formato de entrada. . . . .	38
Formato de entrada con país . . . . .	39
Separador de línea. . . . .	39
Alternativas coincidentes. . . . .	39
Ámbito coincidente. . . . .	40
Recuento máximo de resultados. . . . .	40
Modo . . . . .	40
Nivel de optimización. . . . .	41
Tipo de formato de salida. . . . .	42
Formato de salida con país. . . . .	42
Idioma preferido. . . . .	42
Codificación de caracteres preferida. . . . .	43
Rangos para expandir. . . . .	44
Estandarizar direcciones no válidas. . . . .	44

Nivel de seguimiento. . . . .	45
Informes de certificación. . . . .	45
Campos de informe AMAS. . . . .	46
Campos de informe CASS. . . . .	47
Informe SendRight. . . . .	47
Campos de informe SERP. . . . .	48
Cómo configurar una transformación del validador de direcciones.. . . .	49
Cómo añadir puertos a la transformación del validador de direcciones. . . . .	50
Cómo crear plantillas definidas por el usuario. . . . .	50
Cómo definir modelos del validador de direcciones. . . . .	51
Cómo definir un informe de certificación. . . . .	51

### **Capítulo 3: Transformación de excepción de registros incorrectos. . . . . 53**

Resumen de la transformación de excepción de registros incorrectos. . . . .	53
Tipos de registros de salida de excepción de registros incorrectos. . . . .	54
Flujo de proceso de administración de excepciones de registros incorrectos. . . . .	55
Asignaciones de excepción de registros incorrectos. . . . .	56
Problemas de calidad de excepción de registros incorrectos. . . . .	57
Tareas humanas. . . . .	57
Puertos de excepción de registros incorrectos . . . . .	58
Puertos de entrada de transformación de excepción de registros incorrectos. . . . .	58
Salida de transformación de excepción de registros incorrectos. . . . .	59
Vista de configuración de excepción de registros incorrectos. . . . .	59
Cómo generar la tabla de registros incorrectos y la tabla de problemas. . . . .	61
Asignación problemática de excepción de registros incorrectos . . . . .	61
Asignar puertos a problemas de calidad. . . . .	62
Propiedades avanzadas de Transformación de excepción. . . . .	62
Cómo configurar una transformación de excepción de registros incorrectos. . . . .	63
Ejemplo de asignación de excepción de registros incorrectos. . . . .	64
Mapplet de excepción de registros incorrectos. . . . .	64
Grupos de entrada de ejemplo de excepción de registros incorrectos. . . . .	65
Configuración de ejemplo de excepción de registros incorrectos . . . . .	65
Salida de asignación de ejemplo de excepción de registros incorrectos. . . . .	66

### **Capítulo 4: Transformación de asociación. . . . . 69**

Resumen de la transformación de asociación. . . . .	69
Asignación de memoria. . . . .	70
Propiedades avanzadas de transformación de asociación. . . . .	71

### **Capítulo 5: Transformación de agregación. . . . . 72**

Resumen de la transformación de agregación. . . . .	72
Cómo desarrollar una transformación de agregación. . . . .	72
Puertos de transformación de agregación. . . . .	73

Propiedades avanzadas de transformación de agregación. . . . .	73
Memorias cachés de agregado. . . . .	74
Expresiones de agregado. . . . .	74
Funciones de agregado. . . . .	75
Funciones de agregado anidadas. . . . .	75
Cláusulas condicionales en expresiones de agregado. . . . .	75
Puertos Agrupar por. . . . .	76
Expresiones de no agregado. . . . .	77
Valores predeterminados de puertos Agrupar por. . . . .	77
Entrada ordenada para una transformación de agregación. . . . .	77
Condiciones de entrada ordenada. . . . .	78
Cómo ordenar datos en una transformación de agregación. . . . .	78
Cómo crear una transformación de agregación no reutilizable. . . . .	79
Cómo crear una transformación de agregación no reutilizable. . . . .	79
Consejos para transformaciones de agregación. . . . .	80
Solución de problemas de transformaciones de agregación. . . . .	80
 <b>Capítulo 6: Transformación de conversión de mayúsculas y minúsculas. . . . .</b>	 <b>81</b>
Resumen de la transformación de conversión de mayúsculas y minúsculas. . . . .	81
Propiedades de la estrategia de distinción de mayúsculas y minúsculas. . . . .	82
Cómo configurar una estrategia de conversión de mayúsculas y minúsculas. . . . .	82
Propiedades avanzadas de Transformación de conversión de mayúsculas y minúsculas. . . . .	83
 <b>Capítulo 7: Transformación de clasificador. . . . .</b>	 <b>84</b>
Resumen de la transformación de clasificador. . . . .	84
Modelos de clasificador. . . . .	85
Algoritmos clasificadores. . . . .	85
Opciones de la transformación de clasificador. . . . .	85
Estrategias de clasificador. . . . .	86
Propiedades avanzadas de Transformación de clasificador. . . . .	86
Cómo configurar una estrategia de clasificador. . . . .	87
Ejemplo de análisis de clasificador. . . . .	87
Crear la asignación de clasificador. . . . .	88
Muestra de datos de entrada. . . . .	89
Configuración de origen de datos. . . . .	89
Configuración de transformación de clasificador. . . . .	89
Configuración de transformación de enrutador. . . . .	89
Configuración de destino de datos. . . . .	91
Resultados de la asignación de clasificador. . . . .	91
 <b>Capítulo 8: Transformación de comparación. . . . .</b>	 <b>93</b>
Resumen de la transformación de comparación. . . . .	93
Estrategias para la coincidencia de campos. . . . .	94

Bigram. . . . .	94
Distancia de Hamming. . . . .	94
Distancia de edición. . . . .	95
Distancia de Jaro. . . . .	95
Invertir la distancia de Hamming. . . . .	96
Estrategias de coincidencia de identidades. . . . .	96
Cómo configurar una estrategia de comparación. . . . .	97
Propiedades avanzadas de Transformación de comparación. . . . .	97
<b>Capítulo 9: Transformación de consolidación. . . . .</b>	<b>98</b>
Resumen de la transformación de consolidación. . . . .	98
Asignaciones de consolidación. . . . .	99
Puertos de transformación de consolidación. . . . .	99
Vistas de transformación de consolidación. . . . .	99
Vista Estrategias de la transformación de consolidación. . . . .	99
Propiedades avanzadas de Transformación de consolidación. . . . .	100
Tamaño de archivo caché. . . . .	101
Estrategias simples. . . . .	102
Estrategias basadas en filas. . . . .	103
Estrategias avanzadas. . . . .	103
Funciones de consolidación simple. . . . .	104
CONSOL_AVG. . . . .	104
CONSOL_LONGEST. . . . .	104
CONSOL_MAX. . . . .	105
CONSOL_MIN. . . . .	106
CONSOL_MOSTFREQ. . . . .	106
CONSOL_MOSTFREQ_NB. . . . .	107
CONSOL_SHORTEST. . . . .	107
Funciones de consolidación basada en filas. . . . .	108
CONSOL_GETROWFIELD. . . . .	108
CONSOL_MODALEXACT. . . . .	109
CONSOL_MOSTDATA. . . . .	110
CONSOL_MOSTFILLED. . . . .	111
Ejemplo de asignación de consolidación. . . . .	112
Datos de entrada. . . . .	112
Transformación de generador de claves. . . . .	112
Transformación de consolidación. . . . .	113
Salida de la asignación de consolidación. . . . .	113
Cómo configurar una transformación de consolidación. . . . .	113
<b>Capítulo 10: Transformación de enmascaramiento de datos. . . . .</b>	<b>115</b>
Resumen de transformación de enmascaramiento de datos. . . . .	115
Técnicas de enmascaramiento. . . . .	116

Enmascaramiento aleatorio. . . . .	116
Enmascaramiento de expresiones. . . . .	118
Enmascaramiento de claves. . . . .	120
Máscara de sustitución. . . . .	121
Enmascaramiento dependiente. . . . .	125
Reglas de enmascaramiento. . . . .	127
Formato de máscara. . . . .	128
Caracteres de cadena de origen. . . . .	129
Caracteres de reemplazo de cadena de resultado. . . . .	130
Intervalo. . . . .	131
Desenfoque. . . . .	131
Formatos de máscara especiales. . . . .	132
Enmascaramiento del número de la tarjeta de crédito. . . . .	132
Enmascaramiento de la dirección de correo electrónico. . . . .	133
Enmascaramiento de correo electrónico avanzado. . . . .	133
Enmascaramiento de direcciones IP. . . . .	134
Enmascaramiento del número de teléfono. . . . .	135
Enmascaramiento del número de seguridad social. . . . .	135
Enmascaramiento de direcciones URL. . . . .	136
Enmascaramiento del número de seguro social. . . . .	136
Archivo de valores predeterminados. . . . .	136
Configuración de la transformación de enmascaramiento de datos. . . . .	137
Configure el servicio de integración de datos. . . . .	137
Cómo definir los puertos. . . . .	138
Cómo configurar un enmascaramiento de datos para cada puerto. . . . .	138
Vista previa de los datos enmascarados. . . . .	138
Cómo crear una transformación de enmascaramiento de datos. . . . .	139
Propiedades de la sesión de transformación de enmascaramiento de datos. . . . .	139
Ejemplo de enmascaramiento de datos. . . . .	140
Read_Customer Data. . . . .	141
Transformación de enmascaramiento de datos de clientes. . . . .	141
Resultados de datos de prueba de cliente. . . . .	142
Propiedades avanzadas de Transformación de enmascaramiento de datos. . . . .	142
<b>Capítulo 11: Transformación de decisión. . . . .</b>	<b>144</b>
Resumen de la transformación de decisión. . . . .	144
Funciones de transformación de decisión. . . . .	145
Instrucciones condicionales de transformación de decisión . . . . .	146
Operadores de transformación de decisión. . . . .	147
Control de valores NULL en transformaciones de decisión. . . . .	148
Cómo configurar una estrategia de decisión . . . . .	148
Propiedades avanzadas de Transformación de decisión. . . . .	149

<b>Capítulo 12: Transformación de excepción de registros duplicados.....</b>	<b>150</b>
Resumen de la transformación de excepción de registros duplicados. . . . .	150
Flujo del proceso de la excepción de registros duplicados. . . . .	151
Excepciones de registros duplicados. . . . .	151
Vista de configuración de excepción de registros duplicados . . . . .	152
Cómo generar una tabla de registros duplicados. . . . .	153
Puertos. . . . .	154
Puertos de entrada de transformación de excepción de registros duplicados. . . . .	154
Puertos de salida de transformación de excepción de registros duplicados. . . . .	155
Cómo crear puertos . . . . .	156
Propiedades avanzadas de la transformación de excepción de registros duplicados. . . . .	156
Ejemplo de asignación de excepción de registros duplicados. . . . .	157
Asignación de excepción de registros duplicados. . . . .	157
Transformación de coincidencia. . . . .	158
Grupos de entrada de excepción de registros duplicados. . . . .	158
Vista de configuración de ejemplo de excepción de registros duplicados. . . . .	159
Registros de tabla de salida estándar. . . . .	160
Salida del clúster. . . . .	161
Cómo crear una transformación de excepción de registros duplicados. . . . .	163
 <b>Capítulo 13: Transformación de expresión.....</b>	 <b>164</b>
Resumen de la transformación de expresión. . . . .	164
Cómo desarrollar una transformación de expresión. . . . .	164
Propiedades avanzadas de la transformación de expresión. . . . .	165
Puertos de transformación de expresión. . . . .	165
 <b>Capítulo 14: Transformación de filtro.....</b>	 <b>166</b>
Resumen de la transformación de filtro. . . . .	166
Propiedades avanzadas de la transformación de filtro. . . . .	166
Condición de filtro. . . . .	167
Cómo filtrar filas con valores nulos. . . . .	167
Consejos sobre el rendimiento de la transformación de filtro. . . . .	167
 <b>Capítulo 15: Transformación de Java.....</b>	 <b>168</b>
Resumen de la transformación de Java. . . . .	168
Transformaciones de Java reutilizables y no reutilizables. . . . .	169
Transformaciones de Java activas y pasivas. . . . .	170
Conversión de tipos de datos. . . . .	170
Cómo diseñar una transformación de Java. . . . .	171
Puertos de transformación de Java. . . . .	172
Cómo crear puertos. . . . .	172
Cómo configurar valores de puerto predeterminados. . . . .	172

Propiedades avanzadas de la transformación de Java. . . . .	173
Cómo configurar la classpath para el cliente de Developer Tool. . . . .	175
Cómo configurar la classpath para el servicio de integración de datos. . . . .	175
Cómo desarrollar código Java. . . . .	176
Cómo crear snippets de código Java. . . . .	177
Cómo importar paquetes de Java. . . . .	178
Cómo definir el código de una aplicación auxiliar. . . . .	179
Propiedades de Java de una transformación de Java. . . . .	180
Ficha Importaciones. . . . .	180
Ficha Aplicaciones auxiliares. . . . .	180
Ficha En entrada. . . . .	181
Ficha Al final. . . . .	181
Ficha Funciones. . . . .	182
Ficha Código completo. . . . .	182
Optimización de filtro con la transformación de Java. . . . .	183
Optimización de primera selección con la transformación de Java. . . . .	183
Optimización de inserción con la transformación de Java. . . . .	184
Cómo crear una transformación de Java. . . . .	185
Cómo crear una transformación de Java reutilizable. . . . .	185
Cómo crear una transformación de Java no reutilizable. . . . .	186
Cómo compilar una transformación de Java. . . . .	186
Solución de problemas de una transformación de Java. . . . .	187
Cómo buscar el origen de errores de compilación. . . . .	187
Cómo identificar el origen de errores de compilación. . . . .	188
<b>Capítulo 16: Referencia API de transformación de Java. . . . .</b>	<b>189</b>
Resumen de los métodos API de transformación de Java. . . . .	189
defineJExpression. . . . .	190
failSession. . . . .	191
generateRow. . . . .	192
getInRowType. . . . .	192
getMetadata. . . . .	193
incrementErrorCount. . . . .	193
invokeJExpression. . . . .	194
isNull. . . . .	195
logError. . . . .	196
logInfo. . . . .	196
resetNotification. . . . .	197
setNull. . . . .	197
storeMetadata. . . . .	198
<b>Capítulo 17: Expresiones de Java. . . . .</b>	<b>200</b>
Resumen de las expresiones de Java. . . . .	200

Tipos de función de expresión. . . . .	201
Cómo utilizar el cuadro de diálogo Definir función para definir una expresión. . . . .	201
Paso 1. Configurar la función. . . . .	202
Paso 2. Crear y validar la expresión. . . . .	202
Paso 3. Generar código Java para la expresión. . . . .	202
Cómo crear una expresión y generar código Java mediante el cuadro de diálogo Definir función. . . . .	202
Plantillas de expresiones de Java. . . . .	203
Cómo trabajar con la interfaz simple. . . . .	203
invokeJExpression. . . . .	203
Ejemplo de interfaz simple. . . . .	204
Cómo trabajar con la interfaz avanzada. . . . .	205
Cómo invocar una expresión con la interfaz avanzada. . . . .	205
Reglas y directrices para trabajar con la interfaz avanzada. . . . .	205
Clase EDataType. . . . .	206
Clase JExprParamMetadata. . . . .	206
defineJExpression. . . . .	207
Clase JExpression. . . . .	208
Ejemplo de interfaz avanzada. . . . .	208
Referencia API de la clase JExpression. . . . .	209
getBytes. . . . .	209
getDouble. . . . .	209
getInt. . . . .	210
getLong. . . . .	210
getResultDataType. . . . .	210
getResultMetadata. . . . .	210
getStringBuffer. . . . .	211
invoke. . . . .	211
isResultNull. . . . .	211
<b>Capítulo 18: Transformación de incorporación. . . . .</b>	<b>213</b>
Resumen de la transformación de incorporación. . . . .	213
Propiedades avanzadas de Transformación de combinación. . . . .	214
Puertos de transformación de incorporación. . . . .	214
Cómo definir una condición de unión. . . . .	215
Tipo de condición simple. . . . .	215
Tipo de condición avanzada. . . . .	216
Tipos de unión. . . . .	216
Unión normal. . . . .	216
Outer Join principal. . . . .	217
Outer Join de detalles. . . . .	218
Outer Join completo. . . . .	218
Entrada ordenada para una transformación de incorporación. . . . .	219

Cómo configurar el orden de clasificación. . . . .	219
Cómo añadir transformaciones a la asignación. . . . .	219
Reglas y directrices para las condiciones de unión. . . . .	220
Ejemplo de una condición de unión y del orden de clasificación. . . . .	221
Cómo unir datos de un mismo origen. . . . .	221
Cómo unir dos ramificaciones del mismo canal. . . . .	222
Cómo unir dos instancias del mismo origen. . . . .	223
Pautas para unir datos de un mismo origen. . . . .	223
Cómo bloquear los canales de origen. . . . .	223
Transformación de incorporación no ordenada. . . . .	223
Transformación de incorporación ordenada. . . . .	224
Consejos sobre el rendimiento de la transformación de incorporación. . . . .	224
Reglas y directrices para una transformación de incorporación. . . . .	225
<b>Capítulo 19: Transformación de generador de claves. . . . .</b>	<b>226</b>
Resumen de la transformación de generador de claves. . . . .	226
Estrategia Soundex. . . . .	227
Propiedades de estrategia Soundex. . . . .	227
Estrategia de cadena. . . . .	228
Propiedades de la estrategia de cadena. . . . .	228
Estrategia NYSIIS. . . . .	228
Puertos de salida del generador de claves. . . . .	229
Cómo configurar una estrategia de agrupación. . . . .	229
Propiedades de la creación de claves. . . . .	230
Propiedades avanzadas de Transformación de generador de claves. . . . .	230
<b>Capítulo 20: Transformación de etiquetador. . . . .</b>	<b>231</b>
Resumen de la transformación de etiquetador. . . . .	231
Cuando utilizar una transformación de etiquetador. . . . .	232
Uso de los datos de referencia en la transformación de etiquetador. . . . .	233
Juegos de caracteres. . . . .	234
Modelos probabilísticos. . . . .	234
Tablas de referencia. . . . .	235
Expresiones regulares. . . . .	235
Conjuntos de tokens. . . . .	235
Estrategias de transformación de etiquetador. . . . .	236
Operaciones de etiquetado de caracteres. . . . .	236
Operaciones de etiquetado con tokens. . . . .	236
Puertos de transformación de etiquetador. . . . .	237
Propiedades de etiquetado de caracteres. . . . .	237
Propiedades generales. . . . .	237
Propiedades de la tabla de referencia. . . . .	238
Propiedades de los conjuntos de caracteres. . . . .	238

Propiedades de filtro. . . . .	239
Propiedades de etiquetado con tokens. . . . .	240
Propiedades generales. . . . .	240
Propiedades de conjuntos de tokens. . . . .	241
Propiedades de etiqueta personalizada. . . . .	241
Propiedades de coincidencia probabilística. . . . .	242
Propiedades de la tabla de referencia. . . . .	242
Cómo configurar una estrategia de etiquetado de caracteres. . . . .	243
Cómo configurar una estrategia de etiquetado de tokens. . . . .	244
Propiedades avanzadas de Transformación de etiquetador. . . . .	244
<b>Capítulo 21: Transformación de búsqueda. . . . .</b>	<b>245</b>
Resumen de la transformación de búsqueda. . . . .	245
Búsquedas conectadas y no conectadas. . . . .	246
Búsquedas conectadas. . . . .	247
Búsquedas no conectadas. . . . .	247
Cómo desarrollar una transformación de búsqueda. . . . .	248
Condición de búsqueda. . . . .	248
Reglas y directrices para las condiciones de una transformación de búsqueda. . . . .	249
Memoria caché de búsqueda. . . . .	249
Memoria caché compartida. . . . .	250
Memoria caché persistente. . . . .	250
Propiedades del tiempo de ejecución para Búsquedas en archivos sin formato, tablas de referencia o datos relacionales. . . . .	251
Propiedades avanzadas para transformaciones de búsqueda de archivo sin formato. . . . .	252
Propiedades avanzadas para la tabla de referencia o transformaciones de búsqueda de datos relacionales. . . . .	253
Propiedades avanzadas de transformaciones de búsqueda de objetos de datos lógicos. . . . .	254
Cómo crear una transformación de búsqueda reutilizable. . . . .	254
Cómo crear una transformación de búsqueda no reutilizable. . . . .	255
Cómo crear una transformación de búsqueda no conectada. . . . .	256
Ejemplo de búsqueda no conectada. . . . .	257
<b>Capítulo 22: Transformación de coincidencia. . . . .</b>	<b>259</b>
Resumen de la transformación de coincidencia. . . . .	259
Coincidencia de campos y coincidencia de identidades. . . . .	260
Flujo del proceso del análisis de coincidencia. . . . .	260
Cálculos de puntuación de coincidencia. . . . .	261
Puntuaciones ponderadas. . . . .	262
Puntuaciones de coincidencia nulas. . . . .	262
Opciones de puntuación del clúster. . . . .	262
Rendimiento de asignación de coincidencias. . . . .	264
Componentes de transformación de coincidencia. . . . .	265

Vistas de transformación de coincidencia. . . . .	265
Puertos de transformación de coincidencia. . . . .	265
Coincidencia de campos. . . . .	267
Algoritmos de coincidencia de campos. . . . .	267
Propiedades de la estrategia para la coincidencia de campos. . . . .	269
Propiedades de salida de coincidencia de campos. . . . .	270
Propiedades de coincidencias de campo de origen dual. . . . .	271
Propiedades avanzadas de la coincidencia de campos. . . . .	271
Coincidencia de identidades. . . . .	272
Propiedades de tipo de coincidencia de identidades. . . . .	272
Estrategias de coincidencia de identidades. . . . .	274
Propiedades de la estrategia para la coincidencia de identidades. . . . .	275
Propiedades de salida de coincidencia de identidades. . . . .	275
Propiedades avanzadas de la coincidencia de identidades. . . . .	276
Mapplets de coincidencia. . . . .	277
Cómo crear un mapplet de coincidencia. . . . .	277
Cómo usar un mapplet de coincidencia. . . . .	278
Cómo configurar una operación de coincidencia. . . . .	278
<b>Capítulo 23: Transformación de fusión. . . . .</b>	<b>280</b>
Resumen de la transformación de fusión. . . . .	280
Cómo configurar una estrategia de fusión. . . . .	280
Propiedades avanzadas de Transformación de fusión. . . . .	281
<b>Capítulo 24: Transformación de analizador. . . . .</b>	<b>282</b>
Resumen de la transformación de analizador. . . . .	282
Modos de transformación del analizador. . . . .	283
Cuando utilizar una transformación de analizador. . . . .	283
Uso de los datos de referencia en la transformación de analizador. . . . .	284
Conjuntos de patrones. . . . .	285
Modelos probabilísticos. . . . .	285
Tablas de referencia. . . . .	286
Expresiones regulares. . . . .	286
Conjuntos de tokens. . . . .	286
Operaciones de análisis de tokens. . . . .	287
Puertos del análisis de tokens. . . . .	287
Propiedades del análisis de tokens. . . . .	288
Propiedades generales. . . . .	288
Propiedades del modelo probabilístico. . . . .	289
Propiedades de la tabla de referencia. . . . .	290
Propiedades de conjuntos de tokens. . . . .	290
Modo de análisis basado en patrones. . . . .	291
Puertos de análisis basado en patrones. . . . .	292

Configurar una estrategia de análisis de tokens. . . . .	292
Configurar una estrategia de análisis de patrones. . . . .	293
Propiedades avanzadas de Transformación de analizador. . . . .	293
<b>Capítulo 25: Transformación de rango. . . . .</b>	<b>295</b>
Resumen de la transformación de rango. . . . .	295
Valores de las cadenas de establecimiento de rango. . . . .	296
Memorias caché de rango. . . . .	296
Propiedades de la transformación de rango. . . . .	296
Puertos de la transformación de rango. . . . .	296
Puerto de rango. . . . .	297
Índice de rango. . . . .	297
Cómo definir grupos. . . . .	298
Propiedades avanzadas de la transformación de rango. . . . .	298
<b>Capítulo 26: Transformación de enrutador. . . . .</b>	<b>300</b>
Resumen de la transformación de enrutador. . . . .	300
Cómo trabajar con grupos. . . . .	300
Grupo de entrada. . . . .	301
Grupos de salida. . . . .	301
Cómo utilizar las condiciones del filtro de grupo. . . . .	301
Cómo añadir grupos. . . . .	302
Cómo trabajar con puertos. . . . .	303
Cómo conectar transformaciones de enrutador de una asignación. . . . .	303
Propiedades avanzadas de la transformación de enrutador. . . . .	303
<b>Capítulo 27: Transformación de ordenación. . . . .</b>	<b>305</b>
Resumen de la transformación de ordenación. . . . .	305
Cómo desarrollar una transformación de incorporación. . . . .	306
Propiedades avanzadas de la transformación de ordenación. . . . .	306
Tamaño de la memoria caché de ordenación. . . . .	307
Puertos de transformación de ordenación. . . . .	308
Cómo crear una transformación de ordenación. . . . .	308
Cómo crear una transformación de ordenación reutilizable. . . . .	308
Cómo crear una transformación de ordenación no reutilizable. . . . .	308
Ejemplo de transformación de ordenación. . . . .	309
<b>Capítulo 28: Transformación de SQL. . . . .</b>	<b>311</b>
Resumen de la transformación de SQL. . . . .	311
Propiedades avanzadas de Transformación de SQL. . . . .	312
Puertos de transformación de SQL. . . . .	313
Puertos de entrada. . . . .	313
Puertos de salida. . . . .	314

Puertos de transferencia . . . . .	314
Puerto SQLError. . . . .	315
Número de filas afectadas. . . . .	315
Consulta de transformación de SQL. . . . .	316
Definir la consulta SQL. . . . .	316
Cardinalidad entre fila de entrada y fila de salida. . . . .	317
Cómo procesar instrucciones de consultas. . . . .	318
Configuración de puertos. . . . .	318
Recuento máximo de filas de salida. . . . .	319
Filas de error. . . . .	319
Continuar ante error de SQL. . . . .	320
Optimización del filtro con la transformación de SQL. . . . .	321
Optimización de primera selección con la transformación de SQL. . . . .	321
Optimización de inserción con la transformación de SQL. . . . .	322
Ejemplo de transformación de SQL. . . . .	323
Asignación de objeto de datos lógicos. . . . .	323
Tabla Salario. . . . .	323
Tabla Empleado . . . . .	324
Ejemplo de transformación de SQL. . . . .	324
Salida. . . . .	326
<b>Capítulo 29: Transformación de estandarizador. . . . .</b>	<b>327</b>
Resumen de la transformación de estandarizador. . . . .	327
Estrategias de estandarización. . . . .	327
Propiedades de estandarización. . . . .	328
Cómo configurar una estrategia de estandarización. . . . .	329
Propiedades avanzadas de Transformación de estandarizador. . . . .	329
<b>Capítulo 30: Transformación de unión. . . . .</b>	<b>330</b>
Resumen de la transformación de unión. . . . .	330
Grupos y puertos. . . . .	330
Propiedades avanzadas de la transformación de unión. . . . .	331
Cómo procesar una transformación de unión. . . . .	331
Cómo crear una transformación de unión. . . . .	331
Cómo crear una transformación de unión reutilizable. . . . .	331
Cómo crear una transformación de unión no reutilizable. . . . .	332
<b>Capítulo 31: Transformación de estrategia de actualización. . . . .</b>	<b>333</b>
Resumen de la transformación de estrategia de actualización. . . . .	333
Cómo configurar la estrategia de actualización. . . . .	333
Cómo marcar filas dentro de una asignación. . . . .	334
Expresiones de estrategia de actualización. . . . .	334
Propiedades avanzadas de la transformación de estrategia de actualización. . . . .	334

Transformaciones de estrategia de agregación y actualización. . . . .	335
Cómo especificar opciones de actualización para destinos individuales. . . . .	335
<b>Capítulo 32: Transformación de consumidor de servicio web. . . . .</b>	<b>337</b>
Resumen de la transformación de consumidor de servicio web. . . . .	337
Mensajes SOAP. . . . .	338
Archivos WSDL. . . . .	338
Operaciones. . . . .	338
Seguridad de servicio web. . . . .	339
Selección WSDL. . . . .	340
Puertos de transformación de consumidor de servicio web. . . . .	340
Puertos de entrada de encabezado HTTP. . . . .	340
Otros puertos de entrada. . . . .	341
Asignación de entrada de transformación de consumidor de servicio web. . . . .	342
Reglas y directrices para asignar puertos de entrada a nodos. . . . .	342
Personalizar opciones de vista. . . . .	343
Cómo asignar puertos de entrada a la entrada de operación. . . . .	343
Asignación de salida de transformación de consumidor de servicio web. . . . .	344
Reglas y directrices para asignar nodos a puertos de salida. . . . .	345
Cómo asignar el mensaje SOAP como XML . . . . .	346
Personalizar opciones de vista. . . . .	346
Cómo asignar la salida de operación a puertos de salida. . . . .	346
Propiedades avanzadas de la transformación de consumidor de servicio web. . . . .	348
Control de errores de servicio web. . . . .	350
Compresión de mensajes . . . . .	351
Simultaneidad. . . . .	351
Optimizaciones de filtro. . . . .	352
Habilitar la optimización de primera selección con la transformación del consumidor de servicio web. . . . .	352
Optimización de inserción con la transformación del consumidor de servicio web. . . . .	352
Cómo crear una transformación de consumidor de servicio web. . . . .	354
Ejemplo de transformación de consumidor de servicio web. . . . .	355
Archivo de entrada. . . . .	355
Modelo de objeto de datos lógicos. . . . .	355
Asignación de objetos de datos lógicos. . . . .	356
Transformación de consumidor de servicio web. . . . .	356
<b>Capítulo 33: Análisis de mensajes SOAP del servicio web. . . . .</b>	<b>359</b>
Resumen del análisis de mensajes SOAP del servicio web. . . . .	359
Interfaz de usuario de transformación. . . . .	360
Configuración de salidas de ocurrencia múltiple. . . . .	361
Salida relacional normalizada. . . . .	361
Claves generadas. . . . .	361

Salida relacional desnormalizada. . . . .	362
Salida relacional pivotada. . . . .	363
Cómo analizar elementos anyType. . . . .	363
Cómo analizar tipos derivados. . . . .	364
Análisis de elementos QName. . . . .	365
Cómo analizar grupos de sustitución. . . . .	365
Cómo analizar construcciones XML en mensajes SOAP. . . . .	366
Elemento de selección. . . . .	366
Elemento de lista. . . . .	366
Elemento de unión. . . . .	366
<b>Capítulo 34: Generación de mensajes SOAP del servicio web. . . . .</b>	<b>367</b>
Resumen de la generación de mensajes SOAP del servicio web. . . . .	367
Interfaz de usuario de transformación. . . . .	368
Área Puertos de entrada. . . . .	368
Área Operación. . . . .	369
Relaciones entre puerto y nivel de jerarquía . . . . .	369
Claves. . . . .	370
Asignar puertos. . . . .	371
Asignación de un puerto . . . . .	372
Asignación de un grupo. . . . .	373
Asignación de varios puertos. . . . .	373
Cómo pivotar puertos de ocurrencia múltiple . . . . .	373
Asignación de datos desnormalizados. . . . .	374
Tipos derivados y sustitución de elementos. . . . .	376
Cómo generar tipos derivados. . . . .	376
Cómo generar elementos anyType y atributos. . . . .	377
Cómo generar grupos de sustitución. . . . .	377
Cómo generar construcciones XML en mensajes SOAP. . . . .	378
Elemento de selección. . . . .	378
Elemento de lista. . . . .	378
Elemento de unión. . . . .	379
<b>Capítulo 35: Transformación de media ponderada. . . . .</b>	<b>380</b>
Resumen de la transformación de media ponderada. . . . .	380
Cómo configurar una transformación de media ponderada. . . . .	380
Ejemplo de puntuaciones de coincidencia ponderadas. . . . .	381
Propiedades avanzadas de Transformación de media ponderada. . . . .	381
<b>Apéndice A: Delimitadores de transformación. . . . .</b>	<b>382</b>
Resumen de delimitadores de transformación. . . . .	382
<b>Índice. . . . .</b>	<b>383</b>

# Prefacio

La *Guía de transformación* de Informatica Developer está destinada a desarrolladores de servicios de datos y de calidad de datos. Esta guía da por supuesto que usted entiende los conceptos de integración de datos, archivos sin formato y bases de datos relacionales, así como los motores de base de datos de su entorno. Esta guía también da por sentado que está familiarizado con los conceptos presentados en la *Guía del usuario* de Informatica Developer.

## Documentación de Informatica

### Portal de atención al cliente (My Support) de Informatica

Como cliente de Informatica, puede acceder al portal de atención al cliente de Informatica en <http://mysupport.informatica.com>.

El sitio contiene información sobre productos, información del grupo de usuarios, boletines de noticias, acceso al sistema de administración de casos de atención al cliente de Informatica (ATLAS), la biblioteca de asistencia de Informatica, la base de conocimiento de Informatica, la documentación de productos de Informatica y acceso a la comunidad de usuarios de Informatica.

### Documentación de Informatica

El equipo de Documentación de Informatica se esfuerza al máximo para crear documentación precisa y útil. Si le surgen preguntas o tiene comentarios o ideas relacionadas con esta documentación, póngase en contacto con el equipo de Documentación de Informatica enviando un correo electrónico a [infa\\_documentation@informatica.com](mailto:infa_documentation@informatica.com). Sus opiniones servirán para mejorar nuestra documentación. Por favor, indíquenos si podemos ponernos en contacto con usted en relación a sus comentarios.

El equipo de Documentación actualiza la documentación según sea necesario. Para obtener la documentación más reciente de su producto, visite la Documentación de Productos en la dirección <http://mysupport.informatica.com>.

### Sitio web de Informatica

Puede acceder al sitio web corporativo de Informatica a través de la dirección <http://www.informatica.com>. El sitio contiene información sobre Informatica, su historia, los próximos eventos y las oficinas de ventas. Asimismo, puede encontrar información sobre productos y socios. El área de servicio del sitio incluye información importante sobre soporte técnico, formación y cursos, así como servicios de implementación.

## Biblioteca de asistencia de Informatica

Como cliente de Informatica, puede acceder a la Biblioteca de asistencia de Informatica a través de la dirección <http://mysupport.informatica.com>. La Biblioteca de asistencia es una recopilación de documentos que le ayudarán a contar con más datos sobre los productos y las prestaciones de Informatica. Incluye artículos y demostraciones interactivas que ofrecen soluciones a problemas comunes, comparan prestaciones y conductas y le guían en la realización de tareas específicas del día a día.

## Base de conocimiento de Informatica

Como cliente de Informatica, puede acceder a la Base de conocimiento de Informatica a través de la dirección <http://mysupport.informatica.com>. Utilice la Base de conocimiento para buscar soluciones documentadas a problemas técnicos conocidos relacionados con los productos de Informatica. Asimismo, puede encontrar respuestas a las preguntas más frecuentes, libros blancos técnicos y sugerencias técnicas. Si le surgen preguntas o tiene comentarios o ideas relacionadas con la Base de conocimiento de Informatica, póngase en contacto con el equipo de la Base de conocimiento de Informatica enviando un correo electrónico a [KB\\_Feedback@informatica.com](mailto:KB_Feedback@informatica.com).

## Canal de YouTube de atención al cliente de Informatica

Puede acceder al canal de YouTube de atención al cliente de Informatica en <http://www.youtube.com/user/INFASupport>. El canal de YouTube del Soporte de Informatica incluye vídeos acerca de soluciones que le guiarán en la realización de tareas específicas. Si tiene alguna pregunta, comentarios o ideas acerca del canal de YouTube de atención al cliente de Informatica, póngase en contacto con el equipo de atención al cliente en YouTube de enviando un mensaje de correo electrónico a [supportvideos@informatica.com](mailto:supportvideos@informatica.com) o un tweet a @INFASupport.

## Catálogo de soluciones de Informatica

Informatica Marketplace es un forum donde los desarrolladores y asociados pueden compartir soluciones que aumentan, amplían o mejoran las implementaciones de integración de datos. Al aprovechar cualquiera de los cientos de soluciones disponibles en el Marketplace, puede mejorar la productividad y acelerar el tiempo de implementación en los proyectos. Puede acceder al Catálogo de soluciones de Informatica en <http://www.informaticamarketplace.com>.

## Informatica Velocity

Puede acceder a Informatica Velocity en <http://mysupport.informatica.com>. Con un desarrollo que parte de la experiencia real de cientos de proyectos de administración de datos, Informatica Velocity representa el conocimiento conjunto de nuestros asesores, lo cuales han trabajado con organizaciones de todo el mundo para planificar, desarrollar, implementar y mantener con éxito soluciones de administración de datos. Si tiene alguna pregunta, comentario o idea acerca de Informatica Velocity, póngase en contacto con los Servicios Profesionales de Informatica en [ips@informatica.com](mailto:ips@informatica.com).

## Servicio internacional de atención al cliente de Informatica

Puede ponerse en contacto con nuestro Centro de atención al cliente llamando por teléfono o a través del Soporte en línea.

El Soporte en línea requiere un nombre de usuario y una contraseña. Puede solicitar un nombre de usuario y una contraseña en la dirección <http://mysupport.informatica.com>.

Los números de teléfono del servicio internacional de atención al cliente de Informatica están disponibles en el sitio web de Informatica en

<http://www.informatica.com/us/services-and-training/support-services/global-support-centers/>.



# CAPÍTULO 1

## Introducción a las transformaciones

Este capítulo incluye los siguientes temas:

- [Introducción al resumen de las transformaciones, 1](#)
- [Cómo desarrollar una transformación, 5](#)
- [Transformaciones reutilizables, 5](#)
- [Expresiones de las transformaciones, 6](#)
- [Cómo crear una transformación, 9](#)
- [Niveles de seguimiento, 9](#)

## Introducción al resumen de las transformaciones

Una transformación es un objeto que genera, modifica o pasa datos.

Informatica Developer proporciona un conjunto de transformaciones que realizan funciones específicas. Por ejemplo, una transformación de agregación realiza cálculos en grupos de datos.

Las transformaciones de una asignación representan las operaciones que realiza el servicio de integración de datos en los datos. Los datos pasan por los puertos de transformación que se enlazan en una asignación o un mapplet.

Las transformaciones pueden ser activas o pasivas. Las transformaciones pueden conectarse al flujo de datos o pueden estar desconectadas.

### Transformaciones activas

Una transformación activa cambia el número de filas que pasan por una transformación. También puede cambiar el tipo de fila.

Por ejemplo, la transformación de filtro es activa porque quita filas que no cumplen la condición del filtro. La transformación de estrategia de actualización es activa porque marca filas para su inserción, eliminación, actualización o rechazo.

No se pueden conectar varias transformaciones activas o una transformación activa y otra pasiva a la misma transformación o grupo de entrada de transformación de nivel inferior porque es posible que el servicio de integración de datos no pueda concatenar las filas pasadas por las transformaciones activas.

Por ejemplo, una ramificación de una asignación contiene una transformación de estrategia de actualización que marca una fila para su eliminación. Otra ramificación contiene una transformación de estrategia de actualización que marca una fila para su inserción. Si conecta estas transformaciones a un grupo de entrada de transformación individual, el servicio de integración de datos no puede combinar las operaciones de eliminación y de inserción para la fila.

## Transformaciones pasivas

Una transformación pasiva no cambia el número de filas que pasan por la transformación y mantiene el tipo de fila.

Se pueden conectar varias transformaciones a una misma transformación o grupo de entrada de transformación de nivel inferior si todas las ramificaciones en un nivel superior son pasivas. La transformación que origina la ramificación puede ser activa o pasiva.

## Transformaciones no conectadas

Las transformaciones pueden conectarse al flujo de datos o pueden estar desconectadas. Una transformación no conectada no está conectada a otras transformaciones en la asignación. Se llama a una transformación no conectada dentro de otra transformación y devuelve un valor a esa transformación.

## Transformaciones multiestrategia

Puede definir varias estrategias de transformación en las siguientes transformaciones:

- Mayúsculas y minúsculas
- Clasificador
- Decisión
- Generador de claves
- Etiquetador
- Coincidencia
- Fusión
- Analizador
- Estandarizador

Puede asignar un conjunto de puertos de entrada y de salida diferentes a cada estrategia de la transformación. La transformación almacena las estrategias que se definen en un objeto de transformación individual.

**Nota:** Utilice la vista **Dependencias** para ver las entradas y salidas que utiliza cada una de las estrategias.

## Descripciones de transformaciones

Developer Tool contiene transformaciones comunes y de calidad de datos. Las transformaciones comunes están disponibles en Informatica Data Quality e Informatica Data Services. Las transformaciones de calidad de datos están disponibles en Informatica Data Quality

En la siguiente tabla se describen cada una de las transformaciones:

Transformación	Tipo	Descripción
Validador de direcciones	Activa o pasiva/ Conectada	Corrige los datos de direcciones y devuelve información de validación.
Asociación	Activa/ Conectada	Crea enlaces entre los registros duplicados que están asignados a diferentes clústeres de coincidencia.
Agregación	Activa/ Conectada	Realiza cálculos de agregados.
Conversión de mayúsculas y minúsculas	Pasiva/ Conectada	Estandariza el uso de mayúsculas y minúsculas de las cadenas.
Clasificador	Pasiva/ Conectada	Escribe etiquetas que resumen la información en los campos de puertos de entrada. Se utiliza cuando los campos de entrada contienen grandes cantidades de texto.
Comparación	Pasiva/ Conectada	Genera resultados numéricos que indican el grado de similitud entre pares de cadenas de entrada.
Consolidación	Activa/ Conectada	Crea un registro consolidado a partir de registros identificados como duplicados por parte de la transformación de coincidencia.
Enmascaramiento de datos	Pasiva/ Conectada o desconectada	Reemplaza los datos de producción confidenciales con datos de prueba realistas para entornos que no son de producción.
Decisión	Pasiva/ Conectada	Evalúa las condiciones de los datos de entrada y crea la salida en función de los resultados de dichas condiciones.
Excepción	Activa/ Conectada	Carga excepciones en las tablas que puede revisar y editar un analista. Una excepción es un registro que no pertenece a un conjunto de datos en su forma actual.
Expresión	Pasiva/ Conectada	Calcula un valor
Filtro	Activa/ Conectada	Filtra datos
Java	Activa o pasiva/ Conectada	Ejecuta lógica de usuario codificada en Java. El código byte para la lógica de usuario se encuentra en el repositorio.
Incorporación	Activa/ Conectada	Une datos de bases de datos o sistemas de archivos sin formato diferentes.

Transformación	Tipo	Descripción
Generador de claves	Activa/ Conectada	Organiza registros en grupos en función de los valores de los datos de una columna que usted seleccione.
Etiquetador	Pasiva/ Conectada	Escribe etiquetas que describen los caracteres o cadenas en un campo de puerto de entrada.
Búsqueda	Activa o pasiva/ Conectada o desconectada	Busca y devuelve datos de un archivo sin formato, objeto de datos lógicos, tabla de referencia, tabla relacional, vista o sinónimo.
Coincidencia	Activa/ Conectada	Genera resultados que indican el grado de similitud entre los registros de entrada y agrupa los registros con un alto grado de similitud.
Fusionar	Pasiva/ Conectada	Lee los valores de datos de varias columnas de entrada y crea una columna de salida.
Salida	Pasiva/ Conectada	Define filas de salida de mapplets.
Analizador	Pasiva/ Conectada	Analiza los valores en un campo de puerto de entrada en diferentes puertos de salida en función de los tipos de información que contienen.
Rango	Activa/ Conectada	Limita los registros a un intervalo superior o inferior.
Enrutador	Activa/ Conectada	Distribuye datos en varias transformaciones según condiciones de grupo.
Ordenación	Activa/ Conectada	Ordena datos según una clave de clasificación.
SQL	Activa o pasiva/ Conectada	Ejecuta consultas SQL en una base de datos.
Estandarizador	Pasiva/ Conectada	Genera versiones estandarizadas de cadenas de entrada.
Unión	Activa/ Conectada	Fusiona datos de bases de datos o sistemas de archivos sin formato diferentes.
Estrategia de actualización	Activa/ Conectada	Determina si deben insertarse, actualizarse o rechazarse filas.

Transformación	Tipo	Descripción
Consumidor de servicio web	Activa/ Conectada	Se conecta a un servicio web como un cliente del servicio web para acceder a o transformar datos.
Media ponderada	Pasiva/ Conectada	Lee puntuaciones de coincidencia de las estrategias coincidentes y produce una puntuación media de coincidencia. Puede aplicar diferentes ponderaciones numéricas a cada estrategia, según la importancia relativa de los datos en la estrategia.

## Cómo desarrollar una transformación

Cuando se crea una asignación, puede añadir transformaciones y configurarlas para manejar datos según un propósito empresarial.

Lleve a cabo las siguientes tareas para desarrollar una transformación e incorporarla a una asignación:

1. Añada una transformación no reutilizable a una asignación o mapplet. También puede crear una transformación reutilizable que puede añadir a varias asignaciones o mapplets.
2. Configure la transformación. Cada tipo de transformación tiene un conjunto de opciones único que puede configurarse.
3. Si la transformación es reutilizable, añádala a la asignación o al mapplet.
4. Vincule la transformación a otros objetos en la asignación o mapplet.

Puede arrastrar puertos desde objetos de nivel superior a los puertos de entrada de transformación.

Puede arrastrar puertos de salida desde la transformación a los puertos en objetos de nivel inferior.

Algunas transformaciones utilizan puertos predefinidos que puede seleccionar.

**Nota:** Si crea una transformación reutilizable, debe añadir los puertos de entrada y salida necesarios antes de enlazar la transformación a otros objetos. No puede añadir puertos a la instancia de transformación en el mapplet o en el lienzo de la asignación. Para actualizar los puertos en una transformación reutilizable, abra el objeto de transformación desde el proyecto del repositorio y añada los puertos.

## Transformaciones reutilizables

Las transformaciones reutilizables son transformaciones que se pueden usar en varias asignaciones o mapplets.

Por ejemplo, puede crear una transformación de expresión que calcule el impuesto de valor añadido en Canadá para analizar el coste de operar en ese país. En lugar de hacer el mismo trabajo cada vez, puede crear una transformación reutilizable. Cuando se desea incorporar esta transformación a una asignación, se añade una instancia de la misma a la asignación. Si se cambia la definición de la transformación, todas las instancias de la misma heredan los cambios.

Developer Tool almacena cada transformación reutilizable como metadatos aparte de cualquier asignación o mapplet que utilizan la transformación. Almacena las transformaciones reutilizables en un proyecto o una carpeta.

Cuando se añaden instancias de una transformación reutilizable a asignaciones, los cambios efectuados en la transformación pueden invalidar la asignación o generar datos inesperados.

## Instancias de transformación reutilizable y cambios heredados

Cuando se añade una transformación reutilizable a una asignación o un mapplet, se añade una instancia de la transformación. La definición de la transformación todavía existe fuera de la asignación o del mapplet, mientras que una instancia de la transformación aparece dentro de la asignación o el mapplet.

Cuando se cambia la transformación, las instancias de la transformación reflejan los cambios efectuados. En lugar de actualizar la misma transformación en todas las asignaciones que la utilicen, puede actualizar la transformación reutilizable una vez, con lo que todas las instancias de la transformación heredarán los cambios. Las instancias heredan los cambios realizados en puertos, expresiones, propiedades y el nombre de la transformación.

## Cómo editar una transformación reutilizable

Cuando se edita una transformación reutilizable, todas las instancias de esa transformación heredan los cambios. Algunos cambios pueden invalidar las asignaciones que utilizan la transformación reutilizable.

Puede abrir la transformación en el editor para editar una transformación reutilizable. No se puede editar una instancia de la transformación en una asignación. Sin embargo, puede editar las propiedades de tiempo de ejecución de la transformación.

Si efectúa alguno de los siguientes cambios en una transformación reutilizable, las asignaciones que utilizan instancias de la misma pueden no ser válidas:

- Cuando se elimina uno o más puertos de una transformación se desconecta la instancia de parte o todos los datos de la asignación.
- Cuando se cambia el tipo de datos de los datos de un puerto hace que sea imposible asignar los datos de ese puerto a otro puerto que utiliza un tipo de datos incompatible.
- Cuando se cambia el nombre de un puerto, las expresiones que se refieren al puerto dejan de ser válidas.
- Cuando se especifica una expresión que no es válida en la transformación reutilizable, las asignaciones que utilizan la transformación dejan de ser válidas. El servicio de integración de datos no puede ejecutar asignaciones que no son válidas.

## Expresiones de las transformaciones

Puede introducir expresiones en el **Editor de expresiones** de algunas transformaciones. Las expresiones modifican datos o comprueban si los datos coinciden con las condiciones.

Cree expresiones que utilicen funciones de lenguaje de transformación. Las funciones del lenguaje de transformación son funciones con aspecto SQL que transforman datos.

Introduzca una expresión en un puerto que utilice el valor de los datos de un puerto de entrada o de entrada/salida. Por ejemplo, tiene una transformación con un puerto de entrada IN\_SALARY que contiene los salarios de todos los empleados. Puede utilizar los valores de la columna IN\_SALARY más adelante en la asignación. Al mismo tiempo, también puede usar la transformación para calcular el salario medio y el total de los salarios. Developer Tool requiere que cree un puerto de salida independiente para cada valor calculado.

En la siguiente tabla se enumeran las transformaciones en las que puede introducir expresiones:

Transformación	Expresión	Valor de devolución
Agregador	Lleva a cabo un cálculo de agregación basado en todos los datos que pasen a través de la transformación. De forma alternativa, puede especificar un filtro para los registros del cálculo de agregación con el fin de excluir determinados tipos de registros. Por ejemplo, con esta transformación puede buscar el número total de salarios y el salario medio de todos los empleados de una sucursal.	El resultado es un cálculo de agregación para un puerto.
Expresión	Realiza un cálculo basado en los valores de una fila simple. Por ejemplo, en función del precio y de la cantidad de un elemento en concreto, puede calcular el precio total de la compra de dicha partida de un pedido.	El resultado es un cálculo de nivel de fila para un puerto.
Filtrar	Especifica una condición que se usa para filtrar las filas que pasan a través de esta transformación. Por ejemplo, si desea escribir datos del cliente en la tabla BAD_DEBT para los clientes con saldo pendiente, puede utilizar la transformación Filtrar para filtrar los datos de los clientes.	TRUE o FALSE en función de si la fila cumple la condición especificada. El servicio de integración de datos pasa a través de esta transformación las filas que devuelven el valor TRUE. La transformación aplica este valor a cada fila que pase a través de ella.
Unión	Especifica una condición avanzada que se utilice para unir datos de origen sin ordenar. Por ejemplo, puede concatenar los puertos principales de nombre y apellido y, a continuación, hacerlos coincidir con el puerto detallado de nombre completo.	TRUE o FALSE en función de si la fila cumple la condición especificada. En función del tipo de unión seleccionado, el servicio de integración de datos agrega la fila al conjunto de resultados o la descarta.
Rango	Establece las condiciones de las filas incluidas en un rango. Por ejemplo, puede crear un rango para los primeros diez vendedores contratados por la empresa.	El resultado es una condición o cálculo para un puerto.

Transformación	Expresión	Valor de devolución
Enrutador	Enruta los datos en varias transformaciones en función de una expresión de grupo. Por ejemplo, utilice esta transformación para comparar los salarios de los empleados de tres niveles de pago diferentes. Puede hacer esto creando tres grupos en la transformación de enrutador. Por ejemplo, cree una expresión de grupo para cada rango de salarios.	TRUE o FALSE en función de si la fila cumple la expresión de grupo especificada. El servicio de integración de datos pasa filas que devuelven el valor TRUE a través de cada grupo definido por el usuario en esta transformación. Las filas que dan como resultado FALSE pasan a través del grupo predeterminado.
Estrategia de actualización	Marca una fila para actualizar, insertar, suprimir o rechazar. Utilice esta transformación cuando desee controlar las actualizaciones de un destino, en función de alguna condición que aplique. Por ejemplo, puede usar la transformación de estrategia de actualización para marcar todas las filas de los clientes para su actualización cuando se realice una modificación en la dirección de envío. O bien, puede marcar todas las filas de empleados para rechazarlas en caso de que se trate de personas que ya no trabajan en la empresa.	Código numérico para actualizar, insertar, suprimir o rechazar. La transformación aplica este valor a cada fila que pase a través de ella.

## El editor de expresiones

Utilice el **Editor de expresiones** para crear instrucciones al estilo SQL.

Puede introducir una expresión manualmente o utilizar el método de apuntar y hacer clic. Seleccione funciones, puertos, variables y operadores desde la interfaz de apuntar y hacer clic para reducir al mínimo el número de errores en la construcción de expresiones. El número máximo de caracteres que se pueden incluir en una expresión es de 32.767.

## Nombres de puerto en una expresión

Se pueden introducir nombres de puerto de transformación en una expresión.

En el caso de transformaciones conectadas, si se utilizan nombres de puerto en una expresión, Developer Tool actualiza esa expresión cuando se cambian los nombres de puerto en la transformación. Por ejemplo, se escribe una expresión que determina la diferencia entre dos fechas: Date\_Promised y Date\_Delivered. Si se cambia el nombre de puerto Date\_Promised a Due\_Date, Developer Tool cambia el nombre de puerto Date\_Promised a Due\_Date en la expresión.

**Nota:** Se puede propagar el nombre Due\_Date a otras transformaciones no reutilizables que dependen de este puerto en la asignación.

## Cómo añadir una expresión a un puerto

Se puede añadir una expresión a un puerto de salida.

1. En la transformación, seleccione el puerto y abra el **Editor de expresiones**.
2. Introduzca la expresión.  
Utilice las fichas Funciones y Puertos y las claves del operador.
3. Si lo desea, añada comentarios a la expresión.

Utilice los indicadores de comentario -- o //.

4. Haga clic en el botón Validar para validar la expresión.
5. Haga clic en **Aceptar**.
6. Si la expresión no es válida, corrija los errores de validación y valide nuevamente la expresión.
7. Cuando la expresión sea válida, haga clic en **Aceptar** para cerrar el **Editor de expresiones**.

## Comentarios en una expresión

Se pueden añadir comentarios a una expresión para describirla o para especificar una URL válida para acceder a documentación empresarial sobre la expresión.

Para añadir comentarios dentro de la expresión, utilice los indicadores de comentario -- o //.

## Validación de expresiones

Es necesario validar una expresión para ejecutar una asignación u obtener una vista previa de una salida de mapplet.

Utilice el botón Validar del **Editor de expresiones** para validar una expresión. Si no se valida una expresión, Developer Tool la valida cuando se cierra el **Editor de expresiones**. Si la expresión no es válida, Developer Tool muestra una advertencia. Puede guardar la expresión no válida o modificarla.

## Cómo crear una transformación

Puede crear una transformación reutilizable para reutilizarla en varias asignaciones o mapplets. También puede crear una transformación no reutilizable para usarla una vez en una asignación o en un mapplet.

Para crear una transformación reutilizable, haga clic en **Archivo > Nueva > Transformación**, y complete el asistente.

Para crear una transformación no reutilizable en una asignación o en un mapplet, seleccione una transformación en la paleta Transformación y arrástrela hasta el editor.

Algunas transformaciones requieren que se elija un modo o que se lleve a cabo una configuración adicional cuando se crea la transformación. Por ejemplo, la transformación de analizador requiere que se elija el modo de análisis de tokens o el modo de análisis de patrones en el momento de crear la transformación.

Después de crear una transformación, ésta aparece en el editor. Algunas transformaciones contienen puertos y grupos predefinidos. Otras transformaciones están vacías.

## Niveles de seguimiento

Cuando configure una transformación, puede establecer la cantidad de detalles que el servicio de integración de datos escribe en el registro.

De manera predeterminada, el nivel de seguimiento para cada transformación es Normal. Cambie el nivel de seguimiento a una configuración de detalle si necesita solucionar problemas en una transformación que no

tiene el comportamiento esperado. Establezca el nivel de seguimiento en Simple cuando desee que aparezca la cantidad mínima de detalle en el registro.

Configure la siguiente propiedad en la ficha Avanzadas:

#### Nivel de seguimiento

Cantidad de detalles que aparecen en el registro para una transformación.

En la siguiente tabla se describen los niveles de seguimiento:

Nivel de seguimiento	Descripción
Simplificado	Registra información sobre la inicialización y mensajes de error y notificaciones de datos rechazados.
Normal	Registra información sobre la inicialización y el estado, los errores que aparecen y las filas que se omitieron debido a errores en filas de transformación. Resume los resultados de asignación, aunque no en el nivel de filas individuales.  El valor predeterminado es Normal.
Inicialización detallada	Además del seguimiento normal, registra detalles de inicialización adicionales, nombres de los archivos de datos e índices utilizados, y estadísticas de transformación detalladas.
Datos detallados	Además del seguimiento de inicialización detallada, registra cada fila que pasa por la asignación. También indica dónde se truncan los datos de la cadena para ajustarse a la precisión de una columna, y proporciona estadísticas detalladas de transformación.  Cuando se configura este nivel de seguimiento, los datos de fila de todas las filas de un bloque se escriben en el registro cuando se procesa una transformación.

## CAPÍTULO 2

# Transformación del validador de direcciones

Este capítulo incluye los siguientes temas:

- [Resumen de la transformación del validador de direcciones, 11](#)
- [Datos de referencia de direcciones, 12](#)
- [Grupos de puertos y selección de puertos, 14](#)
- [Plantillas, 14](#)
- [Grupos de puertos de entrada de la transformación del validador de direcciones, 15](#)
- [Grupos de puertos de salida de transformación del validador de direcciones, 15](#)
- [Puertos de varias instancias, 17](#)
- [Proyectos de validación de direcciones, 18](#)
- [Direcciones con formato y estándares del proveedor de servicios de correo, 19](#)
- [Puertos de estado del validador de direcciones, 20](#)
- [Estado del archivo de datos de referencia de direcciones, 31](#)
- [Ajustes generales de la transformación del validador de direcciones, 32](#)
- [Propiedades avanzadas de la transformación del validador de direcciones, 33](#)
- [Informes de certificación, 45](#)
- [Cómo configurar una transformación del validador de direcciones., 49](#)
- [Cómo añadir puertos a la transformación del validador de direcciones, 50](#)
- [Cómo crear plantillas definidas por el usuario, 50](#)
- [Cómo definir modelos del validador de direcciones, 51](#)
- [Cómo definir un informe de certificación, 51](#)

## Resumen de la transformación del validador de direcciones

La transformación del validador de direcciones compara datos de direcciones de entrada con datos de referencia de direcciones para determinar la exactitud de las direcciones y corregir los errores en las

direcciones. La transformación crea registros que cumplen los estándares de envío de correo para la estructura y el contenido de datos. La transformación también añade información de estado a cada dirección.

La transformación del validador de direcciones realiza las siguientes operaciones en datos de direcciones:

- Compara los registros de direcciones en los datos de origen con las definiciones de direcciones en los datos de referencia de direcciones.
- Genera informes de estado detallados sobre la validez de cada dirección de entrada, su estado de entrega y la naturaleza de los errores o ambigüedades que contiene.
- Corrige errores y completa registros de direcciones parciales. Para corregir una dirección, la transformación debe encontrar una coincidencia positiva con una dirección de los datos de referencia. La transformación copia los elementos de datos necesarios de los datos de referencia de direcciones en los registros de direcciones.
- Añade información que no aparece en la dirección estándar pero que ayuda a la entrega postal, como la información de punto de entrega y la información de geocodificación.
- Escribe las direcciones de salida en el formato que el proyecto de datos y el proveedor de servicios de correo requieran. Defina el formato cuando seleccione los puertos de salida en la transformación.

## Datos de referencia de direcciones

Es un conjunto de datos de referencia de direcciones que identifica las direcciones de entrega válidas de un país. Para realizar la validación de direcciones con la transformación del validador de direcciones, debe instalar los datos de referencia de direcciones en el equipo de servicios de Informática en el dominio. Adquiera y descargue los datos de referencia de direcciones de Informática.

Instale un archivo de datos de referencia de direcciones para cada país al que hagan referencia los datos de direcciones de origen. Un país grande puede necesitar varios archivos. El proveedor de servicios de correo nacional de cada país proporciona los datos para los archivos de datos de referencia. Además, también puede instalar archivos de datos que el proveedor de servicios de correo puede usar para certificar la exactitud de las direcciones y para agilizar el envío de correo.

Cuando ejecuta una asignación que contenga una transformación del validador de direcciones, la transformación compara cada fila de datos de entrada con los datos de referencia de direcciones. Si la transformación busca la dirección de entrada en los datos de referencia de direcciones, la transformación actualiza la fila de datos con los datos de direcciones correctos y completos.

Determine los tipos de datos que la validación de direcciones añade a la fila de datos de salida. Cuando configure la transformación del validador de direcciones, seleccione los puertos de salida que contienen los datos que puede necesitar. Puede utilizar los puertos de salida para crear diferentes estructuras de dirección y para añadir metadatos que los proveedores de servicios de correo pueden usar para mejorar la entrega de direcciones.

Utilice la ventana **Preferencias** en Developer Tool para ver información sobre los archivos de datos de referencia de direcciones en el equipo de servicios de Informática en el dominio.

## Tipos de datos de referencia de direcciones

Los archivos de datos de referencia de direcciones que se instalan dependen de los tipos de análisis de direcciones que realice.

Puede instalar los siguientes tipos de datos de referencia de direcciones:

### **Datos de lote**

Instale datos de lote para realizar la validación de direcciones en registros de direcciones. Utilice los datos de lote para comprobar que las direcciones de entrada son completamente válidas para la entrega y están completas basándose en los datos postales actuales del proveedor de servicios de correo nacional.

La transformación del validador de direcciones lee datos de lotes cuando configura la transformación para que se ejecute en el modo por lotes.

### **Datos de CAMEO**

Instale datos de CAMEO para añadir datos de segmentación de clientes en registros de direcciones residenciales. Los datos de segmentación de clientes indican el nivel de renta y las preferencias de estilo de vida probables de los residentes en cada dirección.

La transformación del validador de direcciones lee datos de CAMEO cuando configura la transformación para que se ejecute en el modo por lotes o en el modo de certificación.

### **Datos de certificación**

Instale datos de certificación para comprobar que los registros de direcciones cumplen los estándares de certificación que un proveedor de servicios de correo define. Una dirección cumple un estándar de certificación si contiene elementos de datos que pueden identificar un buzón único, como los elementos de datos de punto de entrega. Si una dirección cumple un estándar de certificación, el proveedor de servicios de correo cobra una tarifa de entrega reducida.

Los siguientes países definen estándares de certificación:

- Australia. Certifica el correo de acuerdo con el estándar del sistema de aprobación de coincidencia de direcciones (AMAS).
- Canadá. Certifica el correo de acuerdo con el estándar del programa evaluación y reconocimiento de software (SERP).
- Francia. Certifica el correo de acuerdo con el estándar del servicio de administración de direcciones nacional (SNA).
- Nueva Zelanda. Certifica el correo de acuerdo con el estándar SendRight.
- Estados Unidos. Certifica el correo de acuerdo con el estándar del sistema de soporte de precisión de codificación (CASS).

La transformación del validador de direcciones lee datos de lotes cuando configura la transformación para que se ejecute en el modo de certificación.

### **Datos de geocodificación**

Instale datos de geocodificación para añadir geocodificación a los registros de direcciones. Los datos de geocodificación son las coordenadas de latitud y longitud.

La transformación del validador de direcciones lee datos de geocodificación cuando configura la transformación para que se ejecute en el modo por lotes o en el modo de certificación.

### **Datos de la lista de sugerencias**

Instale datos de la lista de sugerencias para encontrar versiones válidas alternativas de un registro de dirección incompleto. Use los datos de la lista de sugerencias cuando configure una asignación de validación de direcciones para procesar registros de direcciones únicos en tiempo real. La transformación del validador de direcciones usa los elementos de datos de la dirección incompleta para realizar una comprobación de duplicados en los datos de la lista de sugerencias. La transformación devuelve cualquier dirección válida que incorpore la información en la dirección incompleta.

La transformación del validador de direcciones lee datos de la lista de sugerencias cuando configura la transformación para que se ejecute en el modo de lista de sugerencias.

### Datos complementarios

Instale datos complementarios para añadir datos a un registro de direcciones que sirvan de ayuda para diseñar una campaña de correo. Use los datos complementarios para identificar el área geográfica o el área de población que contiene la dirección. Los datos complementarios también pueden facilitar la entrega del correo al proveedor de servicios de correo.

La transformación del validador de direcciones puede añadir datos complementarios a las direcciones en los siguientes países:

- Japón. Identifica el distrito urbano que contiene la dirección.
- Serbia. Identifica el código postal a nivel de calle para la dirección.
- Reino Unido. Añade un identificador de punto de entrega para la dirección.
- Estados Unidos. Identifica el centro de la población que contiene la dirección.

## Grupos de puertos y selección de puertos

La transformación del validador de direcciones contiene grupos de puertos predefinidos que contienen los puertos de entrada y de salida que se pueden utilizar. Cuando configure una transformación del validador de direcciones, busque los grupos y seleccione los puertos que necesite.

Seleccione los puertos de entrada que correspondan a la estructura de los datos de entrada de las direcciones. Seleccione los puertos de salida que contengan los datos de direcciones que requiera el proyecto.

Puede añadir directamente a la transformación los puertos de entrada y salida, o bien puede crear un modelo predeterminado que contenga puertos de entrada y salida. Si añade los puertos directamente a la transformación, los puertos que seleccione se aplicarán únicamente a dicha transformación. Si añade los puertos al modelo predeterminado, los puertos que seleccione se aplicarán a las futuras transformaciones del validador de direcciones que cree.

**Nota:** El modelo predeterminado no es un objeto de repositorio. El modelo predeterminado reside en el equipo que utilice para crearlo.

Además, también puede añadir a la transformación puertos de transferencia para las columnas que no desee que procese la transformación del validador de direcciones.

## Plantillas

En la transformación puede crear una o varias plantillas de puertos. Una plantilla es un subconjunto de puertos de uno o varios grupos de puertos. Utilice las plantillas para organizar los puertos que espera utilizar con frecuencia en un proyecto.

Las plantillas que cree son visibles en todas las transformaciones del validador de direcciones en el repositorio de modelos.

# Grupos de puertos de entrada de la transformación del validador de direcciones

Antes de conectar los datos de direcciones con los puertos de entrada de la transformación, busque los grupos de entrada y seleccione los puertos que se corresponden con la estructura y el contenido de los datos de entrada. Busque los grupos de salida y seleccione los puertos que cumplan sus requisitos de datos.

La transformación del validador de direcciones muestra los grupos de puertos en un modelo básico y en un modelo avanzado. Puede definir la mayoría de direcciones utilizando un grupo de puertos en el modelo básico. Si las direcciones son complejas, utilice los puertos adicionales que están disponibles en el modelo avanzado.

**Nota:** Seleccione puertos solamente de un grupo de puertos de entrada.

La transformación tiene los siguientes grupos de puerto de entrada:

## Discreto

Lee columnas de datos que contienen información completa sobre un único elemento de datos, como un número de domicilio, un nombre de calle o un código postal. El grupo discreto está disponible en los modelos básico y avanzado.

## Híbrido

Lee columnas de datos que contienen uno o varios elementos de datos. El grupo híbrido combina puertos de los grupos discreto y de varias líneas. Use puertos híbridos para crear registros de direcciones que puede enviar a un proveedor de servicios de correo. Los puertos híbridos permiten estructurar una dirección según los estándares del proveedor de servicios de correo e identificar los tipos de datos de cada línea. El grupo híbrido está disponible en los modelos básico y avanzado.

## Varias líneas

Lee columnas de datos que contienen varios elementos de datos. Cada cadena de entrada se corresponde con una línea de una dirección estructurada con el formato requerido por el proveedor de servicios de correo. Conecte las columnas de dirección con puertos de varias líneas para crear un conjunto de registros de direcciones imprimible.

Cada puerto de varias líneas representa una línea de la dirección impresa, como la siguiente línea de datos de calle:

```
"123 Main Street Apartment 2"
```

Los puertos de varias líneas no especifican el tipo de datos que se muestra en cada línea de dirección. El grupo de varias líneas está disponible en los modelos básico y avanzado.

# Grupos de puertos de salida de transformación del validador de direcciones

Antes de conectar la transformación del validador de direcciones con otras transformaciones u objetos de datos, determine los tipos de información que necesite y la estructura que tendrán las direcciones de salida.

Busque los grupos de salida y seleccione los puertos que cumplan sus requisitos de datos.

**Nota:** Puede seleccionar puertos de varios grupos de salida y puede seleccionar puertos que tienen funciones comunes.

La transformación tiene los siguientes grupos de salida predefinidos:

#### **Elementos de dirección**

Escribe elementos de datos de dirección, como el número de domicilio, el número de apartamento y el nombre de calle en puertos independientes. Busque el grupo de elementos de dirección en los modelos básico y avanzado.

#### **Específico de Australia**

Genera datos de direcciones de Australia que permiten adaptar las direcciones a los estándares de Address Matching Approval System (AMAS) (sistema de aprobación de coincidencia de direcciones) de Australia Post. Busque el grupo específico de Australia en los modelos básico y avanzado.

#### **CAMEO**

Genera datos de resumen demográficos y de ingresos que puede usar en los análisis de segmentación de clientes. Busque el grupo CAMEO en el modelo básico.

#### **Específico de Canadá**

Genera datos de direcciones de Canadá que permiten adaptar las direcciones a los estándares de Software Evaluation and Recognition Program (SERP) (programa de evaluación y reconocimiento de software) de Canada Post. Busque el grupo específico de Canadá en el modelo básico.

#### **Elementos de contacto**

Escribe datos personales o de contacto, como nombres, tratamientos y puestos. Busque el grupo de elementos de contacto en el modelo avanzado.

#### **País**

Escribe el nombre o el código de país definido por la organización internacional de normalización (ISO). Busque el grupo de país en los modelos básico y avanzado.

#### **Línea de dirección formateada**

Escribe direcciones con el formato correspondiente para la impresión y el envío de correo. Busque el grupo de línea de dirección formateada en los modelos básico y avanzado.

#### **Específico de Francia**

Genera datos en direcciones de Francia que permiten que las direcciones cumplan con el servicio nacional de administración de direcciones (SNA) según los estándares de La Poste. Busque el grupo específico de Francia en el modelo básico.

#### **Geocodificación**

Genera datos de geocodificación, como las coordenadas de latitud y longitud, para una dirección. Busque el grupo de geocodificación en el modelo básico.

#### **Elementos de ID**

Escribe datos de ID de registro y clave de transacción. Busque el grupo de elementos de ID en el modelo avanzado.

#### **Suplementario de JP**

Escribe datos de identificación de distrito para direcciones de Japón. Busque el grupo Suplementario de JP en el modelo básico.

#### **Elementos de última línea**

Escribe datos que pueden aparecer en la última línea de una dirección nacional. Busque el grupo de elementos de última línea en los modelos básico y avanzado.

### Específicos de Nueva Zelanda

Genera datos de las direcciones de Nueva Zelanda que ayudan en la entrega postal, incluido el estado de datos SendRight. Busque el grupo específico de Nueva Zelanda en el modelo básico.

### Residuos

Escribe elementos de datos que la transformación no puede analizar para otros puertos. Busque el grupo de residuos en los modelos básico y avanzado.

### Complementario para RS

Escribe los datos de sufijo de código postal para direcciones de Serbia. Busque el grupo Complementario de RS en el modelo básico.

### Información de estado

Genera datos detallados sobre la calidad de cada dirección de entrada y salida. Busque el grupo de información de estado en el modelo básico.

### Complementario para el Reino Unido

Escribe datos de punto de entrega para direcciones de Reino Unido. Busque el grupo Complementario para el Reino Unido en el modelo básico.

### Específico de EE.UU.

Genera datos de direcciones de Estados Unidos que permiten la entrega postal, como datos de punto de entrega. Busque el grupo específico de EE.UU. en el modelo básico.

### Complementario para EE.UU.

Escribe datos geográficos y demográficos, como los códigos FIPS (Federal Information Processing Standards) para las direcciones de Estados Unidos. Busque el grupo Complementario para EE.UU. en el modelo básico.

## Puertos de varias instancias

Muchos tipos de datos de direcciones se pueden incluir varias veces en una dirección. Puede seleccionar varias instancias de un puerto si la dirección contiene varios formatos de mayúsculas y minúsculas de un elemento de datos.

Un puerto de varias instancias puede contener hasta seis instancias. Muchas direcciones usan una sola instancia de un puerto para cada uno de los elementos de datos que contienen. Algunas direcciones usan una segunda instancia de un puerto. Un conjunto de direcciones reducido usa varias instancias de un puerto.

En muchos casos, la primera instancia de un puerto es el nombre principal o el área más amplia identificada por el puerto. Debe comprobar la relación entre las instancias del puerto seleccionado.

### Ejemplo de puertos de calle completa

Un registro de direcciones del Reino Unido puede contener dos nombres de calle, donde un nombre de calle forma parte de un esquema de calle más amplio. La siguiente dirección usa dos puertos de calle completa:

Street Number Complete 1	1A
Street Complete 1	THE PHYGTLE
Street Complete 2	SOUTH STREET

Locality Name 1	NORFOLK
Postcode 1	NR25 7QE

En este ejemplo, los datos de calle del puerto de calle completa 1 dependen de los datos de calle del puerto de calle completa 2. Los datos del número de calle completo 1 hacen referencia a los datos de la calle completa 1.

**Nota:** Aunque el puerto de calle completa 1 especifica la ubicación del buzón, el puerto de calle completa 2 se puede corresponder con el esquema de calle más amplio.

### Ejemplos de puertos de contacto

Un registro de direcciones puede contener varios contactos, donde cada contacto forma parte de un domicilio. La siguiente dirección usa dos puertos de nombre de contacto:

Contact Name 1	MR. JOHN DOE
Contact Name 2	MS. JANE DOE
Formatted Address Line 1	2 MCGRATH PLACE EAST
Formatted Address Line 2	ST. JOHN'S NL A1B 3V4
Formatted Address Line 3	CANADA

En este ejemplo, la organización puede decidir la prioridad que se va a aplicar al nombre de contacto 1 o el nombre de contacto 2. La transformación del validador de direcciones no establece la prioridad de los datos de contacto.

En caso de aplicar un formato a las direcciones para su impresión, puede usar varias instancias de puertos de línea de dirección formateada. Puede seleccionar hasta 12 puertos de línea de dirección formateada.

## Proyectos de validación de direcciones

Puede usar la transformación del validador de direcciones en diversos tipos de proyectos. Puede crear una plantilla de dirección con distintos puertos para cada tipo de proyecto.

Puede definir un proyecto de validación de direcciones con uno o varios de los siguientes objetivos:

### Crear direcciones con formato conforme a los estándares del proveedor de servicios de correo

Puede preparar un conjunto de registros de direcciones de gran tamaño para una campaña de correo. Si crea las direcciones con el formato preferido por el proveedor de servicios de correo, los costos de correo se reducen considerablemente. Cuando prepare las direcciones para el envío de correo, seleccione los puertos de salida que escriben cada línea de la dirección con formato en un solo puerto. Puede seleccionar otro puerto para el nombre de contacto, las líneas de dirección y las líneas de localidad y código postal.

### Organizar las direcciones por indicadores de renta y estilo de vida

Puede añadir datos de segmentación de clientes a los registros de direcciones particulares. Los datos de segmentación de clientes indican el nivel de renta y las preferencias de estilo de vida probables de los residentes en cada dirección. Seleccione puertos del grupo de salida CAMEO para añadir datos de segmentación de clientes a registros de direcciones. Puede utilizar datos de segmentación de clientes en campañas de correo dirigidas a varios mercados de consumidores.

### Crear direcciones certificadas por el proveedor de servicios de correo

Si prepara un conjunto de registros para Australia Post, Canada Post o United States Postal Service (USPS), puede añadir datos para confirmar la garantía de entrega de cada dirección.

La transformación del validador de direcciones puede generar informes para certificar que los registros de direcciones son completos y exactos de conformidad con los estándares de cada proveedor de servicios de correo.

### Crear direcciones que cumplan los requisitos para el cumplimiento de la normativa

Puede comprobar si los registros de direcciones de su organización son exactos de conformidad con la normativa gubernamental o del sector. Seleccione puertos de salida que escriban cada elemento de datos de dirección en un campo independiente. Además, seleccione los puertos de estado de validación de direcciones que proporcionan información detallada acerca de la exactitud e integridad de los datos de salida.

### Mejorar la calidad de los datos de las direcciones

En conjunción con otros proyectos de datos, puede mejorar la estructura y la calidad general de los datos del conjunto de datos de direcciones. Por ejemplo, el conjunto de datos puede contener más columnas de las necesarias o puede contener el mismo tipo de datos en varias columnas. Puede reducir el número de columnas del conjunto de datos y simplificar las columnas usadas para los distintos tipos de datos.

## Direcciones con formato y estándares del proveedor de servicios de correo

Si prepara registros de direcciones para una campaña de correo, debe crear una estructura de direcciones imprimible que coincida con los estándares de formato del proveedor de servicios de correo.

Por ejemplo, USPS mantiene el siguiente formato de dirección para las direcciones nacionales de Estados Unidos:

Line 1	Person/Contact Data	JOHN DOE
Line 2	Street Number, Street, Sub-Building	123 MAIN ST NW STE 12
Line 3	Locality, State, ZIP Code	ANYTOWN NY 12345

Puede definir un formato de dirección imprimible que escriba cada línea de la dirección en un solo puerto. Puede usar puertos que reconozcan los tipos de datos de cada línea o puertos que rellenen la estructura de direcciones con independencia de los datos de cada línea.

En la siguiente tabla, se muestran los distintos métodos para aplicar un formato a una dirección de Estados Unidos para su impresión:

Para esta dirección	Usar estos puertos	O usar estos puertos
JOHN DOE	Línea de destinatario 1	Línea de dirección con formato 1
123 MAIN ST NW STE 12	Línea de dirección de entrega 1	Línea de dirección con formato 2
ANYTOWN NY 12345	Última línea específica de país 1	Línea de dirección con formato 3

Use los puertos de línea de dirección formateada si el conjunto de datos contiene distintos tipos de dirección, como direcciones comerciales o direcciones particulares. Es posible que para una dirección comercial se necesiten tres líneas de dirección para los datos de contacto y organización. La transformación del validador de direcciones garantiza la aplicación del formato correcto a cada dirección comercial o particular mediante el uso de puertos de línea de dirección formateada solamente si son necesarios. No obstante, los puertos de línea de dirección formateada no identifican el tipo de datos que contienen.

Use los puertos de línea de destinatario, línea de dirección de entrega y última línea específica de país si todas las direcciones tienen el mismo formato. Los puertos de línea de destinatario, línea de dirección de entrega y última línea específica de país separan los elementos de datos de direcciones según el tipo de información y facilitan la comprensión del conjunto de datos.

**Nota:** Puede seleccionar otros puertos para procesar esta dirección. En este ejemplo, se incluyen los puertos que permiten aplicar el formato correspondiente a las direcciones para la impresión y el envío.

### Datos demográficos y geográficos

Al crear un conjunto de registros para una campaña de correo, puede añadir varios tipos de datos que, en caso contrario, podrían no aparecer en la dirección. Use estos datos para revisar la extensión demográfica y geográfica de los elementos de correo.

Por ejemplo, puede identificar el distrito electoral al que pertenece una dirección de Estados Unidos. Además, puede generar las coordenadas de latitud y longitud si el país de destino incluye estas coordenadas en los datos de referencia de su sistema de correo.

## Puertos de estado del validador de direcciones

La transformación de validador de direcciones escribe información de estado sobre los elementos de dirección que lee y escribe en puertos de entrada y de salida. Utilice los puertos de información de estado para ver la información de estado.

Puede seleccionar los siguientes puertos de estado:

### Código de resolución de direcciones

Describe los elementos de dirección no válidos en la dirección. Seleccione el puerto en el grupo de puertos Información de estado del modelo básico.

### Tipo de dirección

Indica el tipo de dirección en casos en que el proveedor de servicios de correo reconoce más de una forma de una dirección. Seleccione el puerto en el grupo de puertos Información de estado del modelo básico.

#### **Estado de entrada de elementos**

Describe los niveles de similitud entre los elementos dirección de entrada y los datos de referencia. Seleccione el puerto en el grupo de puertos Información de estado del modelo básico.

#### **Relevancia de elementos**

Identifica los elementos de dirección que el proveedor de servicios de correo requiere para identificar un buzón para la dirección. Seleccione el puerto en el grupo de puertos Información de estado del modelo básico.

#### **Estado de resultado de elementos**

Describe cualquier actualización que la validación de direcciones realiza en la dirección de entrada. Seleccione el puerto en el grupo de puertos Información de estado del modelo básico.

#### **Estado de elementos extendido**

Indica la presencia de datos adicionales de una dirección en los datos de referencia. El puerto puede contener información detallada acerca de las actualizaciones que la validación de direcciones realiza en una dirección. Seleccione el puerto en el grupo de puertos Información de estado del modelo básico.

#### **Estado de geocodificación**

Describe el tipo de datos de geocodificación que la validación de direcciones devuelve para una dirección. Seleccione el puerto en el grupo de puertos Geocodificación del modelo básico.

#### **Puntuación de viabilidad de envío de correo**

Indica la factibilidad de entrega general de una dirección de entrada. Seleccione el puerto en el grupo de puertos Información de estado del modelo básico.

#### **Código de coincidencia**

Describe los resultados de la validación de direcciones en una dirección de entrada. Seleccione el puerto en el grupo de puertos Información de estado del modelo básico.

#### **Porcentaje de resultado**

Representa el grado de similitud entre una dirección de entrada y la dirección de salida correspondiente como un valor de porcentaje. Seleccione el puerto en el grupo de puertos Información de estado del modelo básico.

## **Definiciones de código de estado de elemento**

Los puertos de estado de entrada de elementos, relevancia de elementos, estado de resultado de elementos y estado de resultados de elementos extendidos proporcionan información de estado sobre la validez de los elementos de datos de entrada y salida. Seleccione los puertos de elementos para revisar el resultado de una operación de validación de direcciones.

Los códigos contienen la siguiente información:

- Los códigos de estado de entrada de elementos representan la calidad de la coincidencia encontrada entre los datos de dirección de entrada y los datos de referencia.
- Los códigos de relevancia de elementos identifican los elementos de dirección necesarios para la entrega en las direcciones en el país de destino.
- Los códigos de estado de resultado de elementos describen los cambios realizados en los datos de entrada durante el procesamiento.
- Los códigos de estado de elementos extendido indican que los datos de referencia de dirección contienen información adicional acerca del elemento de dirección.

Cada puerto devuelve un código de 20 caracteres en el que cada carácter hace referencia a un elemento de datos de dirección distinto. Durante la lectura de los códigos de salida de los puertos de elementos, debe

saber a qué elemento hace referencia cada carácter. Los 20 caracteres constan de 10 pares. Los dos códigos de cada par representan un tipo de información de dirección. Por ejemplo, la primera posición del código de retorno representa información básica del código postal.

**Nota:** El puerto de código de resolución de dirección devuelve una cadena de 20 caracteres basada en los mismos elementos de dirección que los puertos de estado de elemento.

En la siguiente tabla, se describen los elementos de dirección que los valores en cada posición identifican:

Posición	Elemento de dirección	Descripción	Ejemplo de elemento de dirección
1	Nivel de código postal 0	Información básica de código postal, como el código postal de cinco dígitos.	Código postal de cinco dígitos 10118
2	Nivel de código postal 1	Información de código postal adicional, como los últimos cuatro dígitos de un código ZIP+4.	0110 en el código ZIP+4 10118-0110
3	Nivel de localidad 0	Ubicación principal, como una ciudad o un municipio.	Londres, en Inglaterra
4	Nivel de localidad 1	Localidad dependiente, como un barrio residencial o un pueblo.	Islington, en Londres
5	Nivel de provincia 0	Región principal de un país, como un nombre de estado de Estados Unidos, un nombre de provincia de Canadá o un cantón de Suiza.	Estado de Nueva York
6	Nivel de provincia 1	Nombre de condado de Estados Unidos.	Condado de Queens, en el Estado de Nueva York
7	Nivel de calle 0	Información de calle principal.	South Great George's Street
8	Nivel de calle 1	Información de calle dependiente.	George's Arcade, en South Great George's Street
9	Nivel de número 0	Número de edificio o casa relacionado con la calle principal.	460, en South Great George's Street
10	Nivel de número 1	Número de edificio o casa relacionado con la calle dependiente.	81, en George's Arcade
11	Nivel de servicio de entrega 0	Descriptor y número de apartado de correos.	Apartado de correos 111
12	Nivel de servicio de entrega 1	Código de la oficina de correos responsable de la entrega.	MAIN STN

Posición	Elemento de dirección	Descripción	Ejemplo de elemento de dirección
13	Nivel de edificio 0	Nombre o número de edificio. No identifica un número de domicilio.	Alice Tully Hall
14	Nivel de edificio 1	Nombre o número de edificio adicional.	Starr Theater, en Alice Tully Hall
15	Nivel de edificio secundario 0	Nombre o número de apartamento, estudio o planta.	80, en 350 5th Avenue, planta 80
16	Nivel de edificio secundario 1	Información del apartamento, estudio o planta relacionada con la información de nivel de edificio secundario 0.	80-18, donde 18 es el número de estudio y 80 es el número de planta
17	Nivel de organización 0	Nombre de compañía.	Address Doctor GmbH
18	Nivel de organización 1	Información corporativa adicional, como la empresa matriz.	Informatica Corporation
19	Nivel de país 0	Nombre de país.	Estados Unidos de América
20	Nivel de país 1	Territorio.	Islas Vírgenes de Estados Unidos

Si un nombre de puerto incluye un sufijo de número, el nivel 0 hace referencia a los datos del número de puerto 1 y el nivel 1 hace referencia a los datos de los números de puerto de 2 a 6.

La información de nivel 0 se puede incluir antes o después de la información de nivel 1 en una dirección impresa. Por ejemplo, el nivel de código postal 1 se incluye después del nivel de código postal 0 y el nivel de localidad 1 se incluye antes del nivel de localidad 0.

## Valores del puerto de salida del código de resolución de direcciones

El código de resolución de direcciones es una cadena de 20 caracteres en el que cada carácter de la cadena representa un elemento diferente de una dirección de entrada. El valor de un carácter describe cualquier elemento de dirección no válido en la posición correspondiente en la dirección.

La siguiente tabla describe los valores del puerto del código de resolución de direcciones:

Código	Descripción
2	El elemento de dirección es obligatorio para la entrega, pero no está presente en la dirección de entrada. Los datos de referencia de direcciones contienen el elemento de dirección que falta. Una salida de 2 indica que la dirección no es válida para la entrega sin el elemento de dirección.
3	El elemento de dirección es un número de casa o un número de calle que está fuera del intervalo válido para la dirección. Por ejemplo, el elemento de dirección contiene un número de casa que no existe en la calle especificada. El modo de lista de sugerencias devuelve direcciones alternativas.
4	La validación de direcciones no puede verificar ni corregir el elemento de dirección porque la dirección de entrada contiene más de una instancia del elemento.
5	El elemento de dirección no es correcto y los datos de referencia de la dirección contienen varias alternativas. La dirección de salida copia la dirección de entrada.
6	El elemento de dirección contradice a otro elemento en la dirección. La validación de direcciones no puede determinar el elemento correcto de la dirección. La dirección de salida copia la dirección de entrada.
7	El elemento de dirección no puede ser corregido sin varios cambios en la dirección. La validación de direcciones puede corregir la dirección, pero el número de cambios indica que la dirección no es fiable.
8	Los datos no se ajustan a las reglas de validación del proveedor de correo.

## Valores del puerto de salida del estado de entrada de elementos

El estado de entrada de elementos es una cadena de 20 caracteres en la que cada carácter representa un elemento diferente de una dirección de entrada. El valor de cada carácter representa el tipo de procesamiento realizado en el elemento de dirección.

Busque el puerto en el grupo de puertos de Información de estado.

En la siguiente tabla se describen los códigos que el estado de entrada de elementos puede devolver en cada posición de la cadena de salida en los modos por lotes, certificado o de lista de sugerencias:

Código	Descripción
0	La dirección de entrada no contiene datos en esta posición.
1	Los datos en esta posición no se encuentran en los datos de referencia.
2	La posición no se puede comprobar porque faltan datos de referencia.

Código	Descripción
3	Los datos son incorrectos. Por ejemplo, un número de calle o de servicio de entrega está fuera del intervalo esperado por los datos de referencia. En los modos por lotes y certificado, los datos de entrada en esta posición se pasan de forma incorrecta como salida. En los modos de lista de sugerencias, la validación de direcciones puede ofrecer alternativas.
4	Los datos de esta posición coinciden con los datos de referencia, pero contienen algunos errores.
5	Los datos de esta posición coinciden con los datos de referencia, pero el elemento de datos se ha corregido o estandarizado.
6	Los datos de esta posición coinciden con los datos de referencia sin errores.

En la siguiente tabla se describen los códigos que el estado de entrada de elementos puede devolver en cada posición de la cadena de salida en el modo de análisis:

Código	Descripción
0	La dirección de entrada no contiene datos en esta posición.
3	Los datos son correctos en esta posición.

## Valores del puerto de salida de relevancia de elementos

Los valores de relevancia de elementos identifican los elementos de dirección que el proveedor de servicios de correo requiere para enviar correo a la dirección. Para cumplir los requisitos para el envío de correo, todos los elementos de dirección con un valor de 1 deben estar presente. Busque el puerto en el grupo de puertos de Información de estado.

Los valores de relevancia de elementos son significativos para elementos de dirección con un valor de Código de coincidencia de C o V.

La siguiente tabla describe los valores del puerto de relevancia de elementos:

Valor	Descripción
0	El proveedor de servicios de correo requiere el elemento de dirección.
1	El proveedor de servicios de correo no requieren el elemento de dirección.

## Valores del puerto de salida del estado de resultado de elementos

El estado de resultado de elementos es una cadena de 20 caracteres en la que cada carácter representa un elemento diferente de una dirección de entrada. El valor de cada carácter describe cualquier actualización que el proceso de validación hace en el elemento de dirección.

Busque el puerto en el grupo de puertos de Información de estado.

La siguiente tabla describe los valores del puerto de estado de resultado de elementos:

Código	Descripción
0	La dirección de salida no contiene datos en esta posición.
1	Los datos de esta posición no se han validado ni se han modificado. El puerto de entrada se copia en el puerto de salida.
2	Los datos de esta posición no se han validado, pero se han estandarizado.
3	Los datos de esta posición se han validado, pero no coinciden con los datos de referencia esperados. Los datos de referencia indican que los datos numéricos no están comprendidos en el intervalo válido. El puerto de entrada se copia en el puerto de salida. Este valor de estado se aplica solamente en el modo por lotes.
4	Los datos de esta posición se han validado, pero no se ha cambiado porque faltan los datos de referencia.
5	Los datos de esta posición se han validado, pero no se han modificado debido a que hay varias coincidencias en los datos de referencia. Este valor de estado se aplica solamente en el modo por lotes.
6	La validación de datos ha eliminado el valor de entrada en esta posición.
7	Los datos de esta posición se han validado y corregido de acuerdo con los datos de referencia.
8	Los datos de esta posición se han validado y modificado añadiendo un valor de los datos de referencia.
9	Los datos de esta posición se han validado pero no se han modificado, y el estado de entrega no está claro. Por ejemplo, el valor de validación de punto de entrega es incorrecto.
C	Los datos de esta posición se han validado y verificado, pero los datos de nombres están desfasados. La validación ha cambiado datos de nombres.
D	Los datos de esta posición se han validado y verificado, pero han cambiado de un exónimo a un nombre oficial.
E	Los datos de esta posición se han validado y verificado. Sin embargo, la validación de datos ha estandarizado la distinción de mayúsculas y minúsculas de caracteres o el idioma. La validación de direcciones puede cambiar el idioma si el valor coincide exactamente con una alternativa de idioma. Por ejemplo, la validación de direcciones puede cambiar "Bruselas" por "Bruxelles" en una dirección belga.
F	Los datos de esta posición se han validado y comprobado, pero no se han modificado debido a una coincidencia exacta con los datos de referencia.

Las posiciones 19 y 20 de la cadena de salida están relacionadas con los datos de país.

En la siguiente tabla se describen los valores que la validación puede devolver para las posiciones 19 y 20:

Código	Descripción
0	El puerto no contiene datos en esta posición.
4	El país se reconoce a partir del valor de país predeterminado establecido en la transformación del validador de direcciones.
E	El nombre de país se reconoce a partir de los datos de direcciones, por ejemplo a partir de un código ISO o un nombre de país.
F	El país se reconoce a partir del valor de la opción para forzar el país establecido en la transformación del validador de direcciones.

## Valores del puerto de salida del estado de elementos extendido

El estado de elementos extendido es una cadena de 20 caracteres en el que cada carácter representa un elemento diferente de una dirección de entrada. Los códigos de salida de puertos complementan los datos de estado en el puerto de estado de entrada de elementos y en el puerto de estado de resultado de elementos. El código de salida de puertos también puede indicar la presencia de información adicional acerca de un elemento de dirección en los datos de referencia.

Busque el puerto en el grupo de puertos de Información de estado.

La siguiente tabla describe los valores del puerto de estado de elementos extendido:

Código	Descripción
1	Los datos de referencia de la dirección contienen información adicional acerca del elemento de dirección. La validación de direcciones no requiere la información adicional.
2	La validación de direcciones ha actualizado el elemento de dirección para solucionar un error de datos o un error de formato. La validación de direcciones no ha comprobado el elemento de dirección.
3	La validación de direcciones ha actualizado el elemento de dirección para solucionar un error de datos o un error de formato. La validación de direcciones ha verificado los datos de número en el elemento de dirección.
4	La validación de direcciones ha movido el elemento de dirección a otro campo para solucionar un error de formato.
5	Los datos de referencia de direcciones contienen una versión alternativa del elemento de dirección, como un nombre preferido de localidad.
6	La validación de direcciones no ha comprobado todos los componentes del elemento de dirección. El elemento incluye datos que la validación de direcciones no puede validar.
7	La validación de direcciones ha encontrado un elemento de dirección en la posición incorrecta en una dirección. La validación de direcciones ha movido el elemento de dirección a la posición correcta.

Código	Descripción
8	La validación de direcciones ha encontrado un elemento de dirección válido en el campo de datos incorrecto. La validación de direcciones ha movido el elemento de dirección al campo correcto.
9	La validación de direcciones ha generado el elemento de salida según las reglas de validación del proveedor de servicios de correo.
A	La validación de direcciones ha encontrado elementos de dirección de tipos diferentes de dirección que son elegibles para la posición actual. La validación de direcciones ha seleccionado el elemento de dirección de salida que cumple con las reglas del proveedor de servicios de correo en el país de destino.
B	La validación de direcciones no puede determinar la relevancia del elemento. La validación de direcciones devuelve el valor predeterminado del país que especifique la dirección.
C	Modo de lista de sugerencias. La validación de direcciones puede devolver sugerencias de dirección adicionales para el elemento de dirección. Para devolver las sugerencias adicionales, actualice la propiedad Recuento máximo de resultados para la transformación del validador de direcciones.
D	La validación de direcciones ha interpolado los datos numéricos en el elemento de dirección.
E	La validación de direcciones no puede devolver el elemento de dirección en el idioma preferido. La validación de direcciones devuelve el elemento en el idioma predeterminado.

## Valores del puerto de salida de la puntuación de viabilidad de envío de correo

En la siguiente tabla se describen los valores del puerto de salida de la puntuación de viabilidad de envío de correo. Busque este puerto en el grupo de puertos Información de estado.

La transformación del validador de direcciones calcula la viabilidad de la entrega de una dirección y escribe una cifra que representa la estimación en el puerto de puntuación de viabilidad de envío de correo. Consulte esta cifra si el valor del código de coincidencia se encuentra en el intervalo de I1 a I4.

En la siguiente tabla se indican los códigos de salida de la puntuación de viabilidad de envío de correo:

Código	Descripción
5	Confirmación total de entrega
4	Confirmación parcial de entrega
3	Entrega probable
2	Entrega relativamente probable
1	Entrega dudosa
0	Entrega no viable

## Valores del puerto de salida del código de coincidencia

En la siguiente tabla se describen los valores del puerto de salida del código de coincidencia. Busque este puerto en el grupo de puertos Información de estado.

Código	Descripción
V4	Verificado. Los datos de entrada son correctos. La validación de direcciones ha comprobado todos los elementos relevantes para el envío postal y las entradas coinciden exactamente.
V3	Verificado. Los datos de entrada son correctos, pero algunos o todos los elementos se han estandarizado, o la entrada contiene nombres desfasados o exónimos.
V2	Verificado. Los datos de entrada son correctos, pero algunos elementos no se han podido verificar debido a que los datos de referencia son incompletos.
V1	Verificado. Los datos de entrada son correctos, pero la estandarización del usuario ha afectado negativamente a la viabilidad de la entrega. Por ejemplo, el código postal tiene una longitud demasiado corta.
C4	Corregido. Todos los elementos relevantes para el envío postal se han comprobado.
C3	Corregido. Algunos elementos no se puede comprobar.
C2	Corregido, pero el estado de entrega no está claro debido a la falta de datos de referencia.
C1	Corregidos, aunque el estado de entrega no está claro debido a los errores introducidos por la estandarización del usuario.
I4	Los datos no se pueden corregir completamente, pero hay una única coincidencia con una dirección en los datos de referencia.
I3	Los datos no se pueden corregir completamente, pero hay varias coincidencias con direcciones en los datos de referencia.
I2	Los datos no se pueden corregir. El modo por lotes devuelve direcciones sugeridas parciales.
I1	Los datos no se pueden corregir. El modo por lotes no se puede sugerir una dirección.
Q3	Modo de lista de sugerencias. La validación de direcciones puede sugerir una o más direcciones completas.
Q2	Modo de lista de sugerencias. La dirección sugerida está completa, pero incluye elementos copiados desde la dirección de entrada.
Q1	Modo de lista de sugerencias. La validación de direcciones no puede sugerir una dirección completa. La dirección necesita más datos de entrada.
Q0	Modo de lista de sugerencias. No hay suficientes datos de entrada para generar sugerencias.

Código	Descripción
RB	País reconocido por la abreviatura. Reconoce códigos de país ISO de dos caracteres e ISO de tres caracteres. Puede reconocer abreviaturas comunes como "GER" para Alemania.
RA	País reconocido en la opción para forzar el país de la transformación.
R9	País reconocido en la opción de país predeterminado de la transformación.
R8	País reconocido por otros elementos de los datos de direcciones.
R0	País no reconocido.
S4	Modo de análisis. La dirección se ha analizado correctamente.
S3	Modo de análisis. La dirección se ha analizado y se han obtenido varios resultados.
S1	Modo de análisis. Se ha producido un error de análisis debido a que un formato de entrada no coincide.
N5	Error de validación. La validación no se produjo porque la base de datos de referencia está desfasada.
N4	Error de validación. La validación no se produjo porque los datos de referencia están dañados o tienen un formato incorrecto.
N3	Error de validación. La validación no se produjo porque los datos de país no se pueden desbloquear.
N2	Error de validación. La validación no se produjo porque la base de datos de referencia requerida no está disponible.
N1	Error de validación. La validación no se produjo porque el país no se reconoce o no es compatible.

## Valores del puerto de salida del estado de geocodificación

En la siguiente tabla se describen los valores del puerto de salida del estado de geocodificación. Busque este puerto en el grupo de puertos de geocodificación.

Seleccione este puerto si ha instalado datos de referencia de geocodificación para un país de dirección de entrada.

Valor	Descripción
EGC0	No hay ninguna geocodificación disponible
EGC1-3	Reservado para uso futuro
EGC4	Datos de geocodificación con precisión de nivel de código postal parcial.
EGC5	Geocodificación con precisión de nivel de código postal

Valor	Descripción
EGC6	Geocodificación con precisión de nivel de localidad
EGC7	Geocodificación con precisión de nivel de calle
EGC8	Geocodificación con precisión de nivel de número de domicilio (dato de geocodificación interpolado)
EGC9	Geocodificación con precisión de nivel de punto de llegada
EGCA	Geocodificación con precisión de nivel de centroide de parcela
EGCU	Base de datos de geocodificación no desbloqueada.
EGCN	Base de datos de geocodificación no encontrada.
EGCC	La base de datos de geocodificación está dañada

## Estado del archivo de datos de referencia de direcciones

Utilice Developer Tool para revisar el estado de los archivos de datos de referencia de direcciones del dominio. La información de estado incluye la fecha de vencimiento de la licencia de cada archivo y el tipo de procesamiento que puede llevar a cabo con el archivo.

Utilice la ventana **Preferencias** de Developer Tool para revisar el estado del archivo de datos de referencia de direcciones. Seleccione la opción **Estado de contenido** de la ventana **Preferencias** para revisar la información de estado.

En la siguiente tabla se describen las propiedades de estado que se muestran de forma predeterminada al seleccionar **Estado de contenido**:

Propiedad	Descripción
Código ISO del país	El país al que se aplica el archivo de datos de referencia de direcciones. Esta propiedad muestra la abreviatura de tres caracteres ISO del país.
Fecha de vencimiento	La fecha en la que el archivo se debe sustituir por otro más nuevo. Se puede utilizar un archivo de datos de referencia de direcciones una vez haya vencido, pero es posible que los datos que contenga ya no sean precisos.
Tipo de país	El tipo de procesamiento de direcciones que puede llevar a cabo con los datos. Seleccione el tipo de procesamiento en la opción Modo de la ficha <b>Configuración general</b> . Si el modo que seleccione no se corresponde con un archivo de datos de direcciones del dominio, se producirá un error al realizar la asignación de validación de direcciones.

Propiedad	Descripción
Desbloquear fecha de vencimiento	La fecha de vencimiento de la licencia del archivo. Después de esta fecha no es posible utilizar el archivo.
Desbloquear fecha de inicio	La fecha en la que entra en vigor la licencia para el archivo. Antes de esta fecha no es posible utilizar el archivo.

Haga clic con el botón derecho del ratón en la tabla de propiedades para ver una lista de propiedades adicionales.

## Ajustes generales de la transformación del validador de direcciones

Configure los ajustes generales para preparar los parámetros necesarios para la validación de direcciones.

Configure las siguientes propiedades en la vista **Configuración general**:

### País predeterminado

Define el conjunto de datos de referencia que utiliza la transformación en el caso de que no pueda determinar la información del país a partir de los datos de dirección de entrada. Seleccione Ninguno si sus datos ya incluyen el país.

### Forzar país

Fuerza a la transformación a utilizar el conjunto de datos de referencia independientemente de la información específica sobre el país que pueda contener una dirección.

### Separador de línea

Especifica el símbolo delimitador que separa campos de datos en una dirección formada por una línea.

### Uso de mayúsculas y minúsculas

Define el estilo de formato de mayúsculas y minúsculas de los caracteres para los datos de salida. Seleccione Mixto para seguir el estándar de los datos de referencia para letras mayúsculas. Seleccione Conservado para restaurar el estilo de formato de mayúsculas y minúsculas utilizado en los datos de referencia de direcciones.

### Modo

Determina el tipo de validación que realizará la transformación. El modo predeterminado es Por lotes.

Seleccione una de las siguientes opciones:

Tipo de modo	Descripción
Reconocimiento del país	Determina un país de destino para la dirección postal sin realizar ninguna validación de direcciones.
Analizar	Analiza los datos de campos de dirección sin realizar ninguna validación.

Tipo de modo	Descripción
Lista de sugerencias	Lleva a cabo la validación de la dirección y genera una lista de direcciones que posiblemente coincidan con una dirección de entrada. <b>Nota:</b> Utilice el modo Lista de sugerencias con puertos del grupo de puertos de entrada discreta.
Lote	Realiza la validación de direcciones.
Certificado	Realiza una validación que cumple los estándares de certificación de un servicio postal.

## Modo de lista de sugerencias

Cuando selecciona el modo Lista de sugerencias, la transformación del validador de direcciones busca los datos de referencia de direcciones y devuelve todas las direcciones que sean posibles coincidencias con la dirección de entrada.

Seleccione este modo cuando desee verificar la precisión de una dirección.

Por ejemplo, puede usar este modo en una asignación de validador de direcciones que se ejecuta desde un servicio web en un terminal de entrada de datos. Cuando un usuario introduce una dirección en el terminal de entrada de datos, el servicio web ejecuta la asignación. La asignación devuelve todas las direcciones que coincidan o que sean similares de los datos de referencia de direcciones. El usuario puede decidir si acepta la dirección de entrada o seleccionar una dirección sugerida por la asignación.

Tenga en cuenta los siguientes factores al seleccionar el modo Lista de sugerencias:

- Utilice los puertos del grupo de entrada Discreto al configurar la transformación en el modo Lista de sugerencias.
- El modo de lista de sugerencias puede devolver varios registros para cada registro de entrada. Puede utilizar el modo de lista de sugerencias para validar un conjunto de datos de cualquier tamaño, aunque no es una alternativa a las operaciones del modo por lotes.

## Propiedades avanzadas de la transformación del validador de direcciones

Configure las propiedades avanzadas para determinar cómo el servicio de integración de datos procesa los datos para la transformación del validador de direcciones.

### Alias de calle

Determina si la validación de direcciones reemplaza un alias de calle válido con el nombre de calle oficial.

Un alias de calle es un nombre de calle alternativo que USPS reconoce como un elemento en una dirección de entrega. Puede utilizar la propiedad cuando configura la transformación del validador de direcciones para validar registros de direcciones de Estados Unidos en el modo certificado.

En la siguiente tabla se describen las opciones de Alias de calle:

Opción	Descripción
Oficial	Reemplaza cualquier nombre de calle alternativo o alias de calle con el nombre oficial de la calle. Opción predeterminada.
Conservar	Conserva un nombre de calle alternativo o un alias de calle válidos. Si el nombre de calle introducido no es válido, la validación de direcciones reemplaza el nombre con el nombre oficial.

## Uso de mayúsculas y minúsculas

Determina si la transformación usa mayúsculas o minúsculas para escribir los datos de salida.

En la siguiente tabla se describen las opciones del uso de mayúsculas y minúsculas:

Opción	Descripción
Asignar parámetro	Se asigna un parámetro que identifica el uso de mayúsculas y minúsculas.
Minúsculas	Escribe los datos de salida en minúsculas.
Mixto	Usa el estilo de mayúsculas y minúsculas que se utiliza en el país de destino siempre que sea posible.
Reservado	Aplica el estilo de mayúsculas o minúsculas que se utiliza en los datos de referencia. Opción predeterminada.
Mayúsculas	Escribe los datos de salida en mayúsculas.

También puede configurar el uso de mayúsculas o minúsculas en la ficha **Configuración general**.

### Uso de parámetro

Puede asignar un parámetro para especificar el uso de mayúsculas y minúsculas. Cuando cree el parámetro, especifique una opción de mayúsculas y minúsculas como el valor del parámetro.

Debe especificar el valor en mayúsculas. Introduzca LOWER, MIXED, PRESERVED o UPPER.

## País de origen

Identifica el país en el que se envían los registros de direcciones.

Seleccione un país de la lista. De forma predeterminada, la propiedad está vacía.

## Tipo de país

Determina el formato del nombre o la abreviatura del país en los datos de salida del puerto de línea de dirección formateada o dirección completa. La transformación escribe el nombre o abreviatura del país en el formato estándar del país que seleccione.

En la siguiente tabla se describen las opciones del tipo de país:

Opción	País
ISO 2	Código ISO del país de dos caracteres
ISO 3	Código ISO del país de tres caracteres
ISO #	Código ISO del país de tres dígitos
Abreviatura	(Reservado para su uso en el futuro)
CN	Canadá
DA	(Reservado para su uso en el futuro)
DE	Alemania
EN	Gran Bretaña (predeterminado)
ES	España
FI	Finlandia
FR	Francia
GR	Grecia
IT	Italia
JP	Japón
HU	Hungría
KR	Corea, República de
NL	Países Bajos
PL	Polonia
PT	Portugal
RU	Rusia
SA	Arabia Saudí
SE	Suecia

## País predeterminado

Especifica el conjunto de datos de referencia de direcciones que utiliza la transformación cuando un registro de direcciones no identifica un país de destino.

Seleccione un país de la lista. Utilice la opción predeterminada si los registros de direcciones incluyen información del país. El valor predeterminado es Ninguno.

También puede configurar el país predeterminado en la ficha **Configuración general**.

### Uso de parámetro

Puede usar un parámetro para especificar el país predeterminado. Cuando cree el parámetro, escriba el código ISO 3166-1 alfa-3 para el país como valor del parámetro. Cuando introduzca un valor de parámetro, utilice caracteres en mayúscula. Por ejemplo, si todos los registros de direcciones incluyen información del país, introduzca NONE.

## Prioridad de dirección dual

Determina el tipo de dirección que validar. Establezca la propiedad cuando los registros de direcciones de entrada contengan más de un tipo de datos de direcciones válidos.

Por ejemplo, use la propiedad cuando un registro de direcciones contenga elementos de apartado de correos y elementos de calle. La validación de direcciones lee los elementos de datos que contienen el tipo de datos de dirección que especifique. La validación de direcciones omite los datos incompatibles en la dirección.

La siguiente tabla describe las opciones de la propiedad Prioridad de dirección dual:

Opción	Descripción
Servicio de entrega	Valida los elementos de datos del servicio de entrega de una dirección, como los elementos de apartado de correos.
Administración postal	Valida los elementos de dirección requeridos por el proveedor de servicios de correo local. Opción predeterminada.
Calle	Valida los elementos de datos de calle de una dirección, como los elementos de número de edificio y los elementos de nombre de calle.

## Abreviatura de elemento

Determina si la transformación devuelve la forma abreviada de un elemento de dirección. Puede establecer la transformación para que devuelva la forma abreviada si los datos de referencia de direcciones contienen abreviaturas.

Por ejemplo, el servicio postal de Estados Unidos (USPS) tiene formas breves y largas de muchos nombres de calles y localidades. La forma corta de HUNTSVILLE BROWNSFERRY RD es HSV BROWNS FRY RD. Puede seleccionar la propiedad Abreviatura de elemento cuando los valores de calle o localidad excedan la longitud máxima de campo que especifica USPS.

La opción está desactivada de manera predeterminada. Active la propiedad para que devuelva los valores de dirección abreviados. La propiedad devuelve el nombre de localidad y el código de localidad abreviados cuando se utiliza la transformación en modo por lotes. La propiedad devuelve el nombre de calle, el nombre de localidad y el código de localidad abreviados cuando se utiliza la transformación en el modo de certificación.

## Instancias de ejecución

Determina el número de subprocesos que usa la transformación actual en tiempo de ejecución.

Establezca el número de instancias de ejecución para un valor menor o igual que la cantidad de CPU a las que puede acceder el servicio de integración de datos. El valor que establezca se aplicará a la transformación actual. El valor predeterminado es 1.

Cuando defina un valor de instancias de ejecución superior a uno, cambie la transformación del validador de direcciones de una transformación pasiva a una transformación activa.

**Nota:** Si se exporta una transformación del validador de direcciones a PowerCenter, el proceso de exportación crea una instancia de la transformación en el repositorio de PowerCenter para cada instancia de ejecución. Por ejemplo, si establece el número de instancias de ejecución en 1, el proceso de exportación añade una transformación del validador de direcciones única al repositorio de PowerCenter. Si establece el número de instancias de ejecución en 2, el proceso de exportación añade dos instancias de transformación al repositorio de PowerCenter.

## Expansión de rango flexible

Impone un límite práctico sobre el número de direcciones que la transformación del validador de direcciones devuelve al establecer la propiedad Rangos para expandir. Puede establecer la propiedad Rangos para expandir y la propiedad Expansión de rango flexible al configurar la transformación para que se ejecute en el modo de lista de sugerencias.

La propiedad Rangos para expandir determina cómo la transformación devuelve sugerencias de dirección si una dirección de entrada no contiene datos de número de domicilio. Si la dirección de entrada no incluye datos contextuales, como un código postal completo, la propiedad Rangos para expandir puede generar un gran número de direcciones muy similares. La propiedad Expansión de rango flexible limita el número de direcciones que la propiedad Rangos para expandir genera para una dirección individual. Habilite la propiedad Expansión de rango flexible al establecer la propiedad Rangos para expandir en Todos.

La siguiente tabla describe las opciones de la propiedad Expansión de rango flexible:

Opción	Descripción
Activada	La validación de direcciones limita el número de direcciones que la propiedad Rangos para expandir añade a la lista de sugerencias. Opción predeterminada.
Desactivada	La validación de direcciones no limita el número de direcciones que la propiedad Rangos para expandir añade a la lista de sugerencias.

**Nota:** La transformación del validador de direcciones aplica la propiedad Expansión de rango flexible de forma diferente para cada dirección que devuelve a la lista de sugerencias. La transformación no impone un límite fijo sobre el número de direcciones expandidas de la lista. La transformación también considera el valor de la propiedad Recuento máximo de resultados al calcular el número de direcciones ampliadas que incluir en la lista.

## Tipo de datos de geocodificación

Determina el modo en que la transformación del validador de direcciones calcula los datos de geocodificación para una dirección. Los datos de geocodificación son las coordenadas de latitud y longitud.

Puede seleccionar una de las siguientes opciones:

### **Punto de llegada**

Devuelve las coordenadas de latitud y longitud de la entrada al edificio o a la parcela de terreno. Opción predeterminada.

Puede seleccionar la opción de geocodificación del punto de llegada para direcciones de Estados Unidos y Canadá. Si la transformación del validador de direcciones no puede devolver datos de geocodificación del punto de llegada, devolverá datos de geocodificación interpolados.

### **Estándar**

Devuelve las coordenadas de latitud y longitud estimadas de la entrada al edificio o a la parcela de terreno. Un dato de geocodificación estimado se llama dato de geocodificación interpolado.

La transformación del validador de direcciones usa los datos de geocodificación más cercanos en los datos de referencia de direcciones para calcular los datos de geocodificación para la dirección.

### **Centroide de parcela**

Devuelve las coordenadas de latitud y longitud del centro geográfico de la parcela de terreno a nivel del terreno. Si la transformación del validador de direcciones no puede devolver datos de geocodificación del centroide de la parcela, la transformación no devolverá ningún dato de geocodificación.

## **Longitud de campo máxima global**

Determina el número máximo de caracteres en cualquier línea de la dirección. Si la transformación del validador de direcciones escribe una línea de dirección de salida que contiene más caracteres que los que especifique, la transformación abrevia los elementos de dirección de la línea.

Use la propiedad para controlar la longitud de línea de la dirección. Por ejemplo, los estándares del SNA requieren que una dirección no contenga más de 38 caracteres en una línea. Si se generan direcciones de acuerdo con el estándar del SNA, defina la longitud de campo máxima global en 38.

El valor predeterminado es 1.024.

## **Tipo de formato de entrada**

Describe el tipo de información más común que contienen los datos de entrada que no se utilizan. Utilice la propiedad Tipo de formato de entrada cuando conecte datos de entrada a los puertos Dirección completa o Línea de dirección formateada. Seleccione la opción que mejor describa la información de los datos de origen de la asignación.

Seleccione una de las siguientes opciones:

- Todos
- Dirección
- Organización
- Contacto
- Organización/Contacto  
La dirección incluye información de la organización e información de contacto.
- Organización/Departamento  
La dirección incluye información de la organización e información del departamento.

La opción predeterminada es Todos.

## Formato de entrada con país

Determina si la entrada contiene datos del país. Seleccione esta propiedad si conecta los datos de entrada con los puertos de entrada Dirección completa o Línea de dirección formateada y si los datos contienen información del país.

La opción está desactivada de manera predeterminada.

## Separador de línea

Especifica el símbolo delimitador que indica los saltos de línea en una dirección formateada.

Seleccione una de las siguientes opciones:

- Asignar un parámetro para identificar el separador de línea
- Retorno de carro
- Coma
- Avance de línea/LF
- Punto y coma
- Tabulador
- Nueva línea en Windows/CRLF

El valor predeterminado es punto y coma.

También puede configurar el separador de línea en la ficha **Configuración general**.

### Uso de parámetro

Puede asignar un parámetro para especificar el separador de línea. Cuando cree el parámetro, introduzca una opción de separador de línea como valor predeterminado. El valor no distingue entre mayúsculas y minúsculas.

## Alternativas coincidentes

Determina si la validación de direcciones reconoce nombres de lugares alternativos, como sinónimos o nombres históricos, en una dirección de entrada. La propiedad se aplica a datos de calle, localidad y provincia.

**Nota:** La propiedad Alternativas coincidentes no conserva los nombres alternativos en una dirección validada.

En la siguiente tabla se describen las opciones de las alternativas coincidentes:

Opción	Descripción
Todos	Reconoce todos los nombres de calles y nombres de lugares alternativos conocidos. Opción predeterminada.
Solo archivos	Reconoce solo los nombres históricos. Por ejemplo, la validación de direcciones valida "Constantinopla" como una versión histórica de "Estambul".
Ninguno	No reconoce nombres de calles y nombres de lugares alternativos.
Solo sinónimos	Solo reconoce sinónimos y exónimos. Por ejemplo, la validación de direcciones valida "Seville" como un exónimo de "Sevilla".

## Ámbito coincidente

Determina la cantidad de datos que la transformación coteja con los datos de referencia de direcciones durante la validación de direcciones.

En la siguiente tabla se describen las opciones del ámbito coincidente:

Opción	Descripción
Todos	Valida todos los puertos seleccionados. Opción predeterminada.
Punto de entrega	Valida los datos de la dirección de edificios y subedificios, además de los datos que valida la opción Calle.
Localidad	Valida los datos de provincia, localidad y código postal.
Calle	Valida datos de direcciones de calle además de los datos que valida la opción Localidad.

## Recuento máximo de resultados

Determina el número máximo de direcciones que la validación de direcciones puede devolver en el modo de lista de sugerencias.

Puede establecer un número máximo en el intervalo de 1 a 100. El valor predeterminado es 20.

**Nota:** El modo de lista de sugerencias comprueba las direcciones en los datos de referencia de direcciones y devuelve una lista de direcciones que son posibles coincidencias para la dirección de entrada. Cuando se comprueba una dirección en el modo de lista de sugerencias, la validación de direcciones devuelve primero la mejor coincidencia.

## Modo

Determina el tipo de análisis de direcciones que realiza la transformación. También puede configurar el modo en la ficha **Configuración general** de la transformación.

### Uso de parámetro

Puede utilizar un parámetro para especificar el modo de análisis. Cuando defina el parámetro, introduzca el modo como valor predeterminado. El valor utiliza caracteres en mayúscula y no contiene espacios.

La siguiente tabla describe las opciones del menú de modo y los valores de parámetros correspondientes que puede establecer:

Modo	Descripción	Valor del parámetro
Lote	Realiza la validación de direcciones con los datos de referencia de direcciones. Opción predeterminada.	BATCH
Certificado	Lleva a cabo la validación de direcciones frente a los datos de referencia para los estándares de certificación del servicio postal.	CERTIFIED

Modo	Descripción	Valor del parámetro
Reconocimiento del país	Identifica las direcciones de entrada que no contienen información del país. Anexa la información del país a una dirección cuando los datos de la localidad o de la provincia identifican un único país.	COUNTRYRECOGNITION
Analizar	Analiza los datos de los campos de dirección y no realiza la validación de direcciones.	PARSE
Lista de sugerencias	Realiza una comprobación de dirección en los datos de referencia de direcciones y genera una lista de direcciones que pueden ser coincidencias para una dirección de entrada. Utilice el modo Lista de sugerencias con aplicaciones de punto de entrada.	FASTCOMPLETION
Asignar parámetro	Asigna un parámetro que identifica un modo de validación de direcciones.	Introduzca un valor de parámetro de la tabla actual.

## Nivel de optimización

Determina cómo la transformación coteja datos de direcciones de entrada y datos de referencia de direcciones. La propiedad define el tipo de coincidencia que la transformación debe encontrar entre los datos de entrada y los datos de referencia antes de que puede actualizar el registro de direcciones.

En la siguiente tabla se describen las opciones del nivel de optimización:

Opción	Descripción
Estrecho	La transformación analiza los números de edificios o de domicilios con la información de la calle antes de realizar la validación. De lo contrario, la transformación valida los elementos de la dirección de entrada estrictamente según la estructura del puerto de entrada. La opción estrecha es la que realiza la validación de la dirección de forma más rápida, pero es posible que se obtengan resultados menos precisos que en otras opciones.
Estándar	La transformación analiza varios tipos de información de la dirección de los datos de entrada antes de llevar a cabo la validación. Al seleccionar la opción estándar, la transformación actualiza una dirección en caso de que pueda hacer coincidir varios valores de entrada con los datos de referencia. El valor predeterminado es Estándar.
Ancho	La transformación utiliza la configuración del análisis estándar y realiza operaciones de análisis adicionales en los datos de entrada. Al seleccionar esta opción, la transformación actualiza una dirección en caso de que pueda hacer coincidir al menos un valor de entrada con los datos de referencia. Esta opción aumenta los tiempos de ejecución de la asignación.

## Tipo de formato de salida

Describe el tipo de información más común que la transformación escribe en el puerto de salida de línea de dirección formateada o dirección completa. Seleccione la opción que mejor describa los datos que espera en el puerto de salida.

Seleccione una de las siguientes opciones:

- Todos
- Dirección
- Organización
- Contacto
- Organización/Contacto  
La dirección incluye información de la organización e información de contacto.
- Organización/Departamento  
La dirección incluye información de la organización e información del departamento.

La opción predeterminada es Todos.

## Formato de salida con país

Determina si la transformación escribe datos de identificación de país en los puertos de salida de línea de dirección formateada o dirección completa.

La opción está desactivada de manera predeterminada.

## Idioma preferido

Determina el idioma de los datos de salida.

De forma predeterminada, la transformación devuelve valores de dirección en el idioma que especifiquen los datos de referencia de direcciones. Si los datos de referencia de direcciones almacenan los elementos de dirección en más de un idioma, puede especificar el idioma.

La siguiente tabla describe las opciones de la propiedad:

Opción	Descripción
Alternativa 1, Alternativa 2, Alternativa 3	Devuelve los elementos de dirección en un idioma alternativo. Los datos de referencia de direcciones del país de la dirección determinan el idioma que puede seleccionar.
Base de datos	Devuelve la dirección en el idioma que especifiquen los datos de referencia de direcciones para la dirección. El valor predeterminado es Base de datos.
Inglés	Devuelve las versiones en inglés de los nombres de localidad y provincia, en caso de que los datos de referencia contengan versiones en inglés.
Conservar entrada	Devuelve los elementos de dirección en el idioma que usa la dirección de entrada.

La transformación del validador de direcciones puede procesar un origen de datos que contiene datos en varios idiomas y conjuntos de caracteres. Cuando la transformación lee una dirección que no está en

alfabeto latino, translitera la dirección al alfabeto latino antes de la validación. La transformación valida el formato latino de la dirección y convierte los elementos de dirección en el idioma que defina la propiedad. La propiedad utiliza el idioma del registro de entrada de forma predeterminada.

**Nota:** Si establece un idioma y un script preferidos para los datos de salida, compruebe que el idioma y la codificación de caracteres que seleccione sean compatibles.

## Codificación de caracteres preferida

Determina el conjunto de caracteres que la transformación del validador de direcciones usa para los datos de salida.

La transformación puede procesar un origen de datos que contiene datos en varios idiomas y conjuntos de caracteres. La transformación convierte todos los datos de entrada en el conjunto de caracteres Unicode UCS-2 y procesa los datos en ese formato. Después de que la transformación procese los datos, convierte los datos de cada registro de direcciones en el conjunto de caracteres que especifique en la propiedad. El proceso se llama transliteración. De forma predeterminada, la propiedad utiliza el conjunto de caracteres que utilizan los datos de referencia de direcciones locales.

La siguiente tabla describe las opciones de la propiedad:

Opción	Descripción
ASCII (Simplificado)	Devuelve datos de dirección en caracteres ASCII.
ASCII (Extendido)	Devuelve datos de dirección en caracteres ASCII y expande los caracteres especiales. Por ejemplo, Õ se translitera a OE.
Base de datos	Devuelve datos de dirección en el conjunto de caracteres de los datos de referencia de direcciones que se aplican a la dirección. El valor predeterminado es Base de datos.
Latin	Devuelve datos de dirección en caracteres Latin I.
Latin (Alt.)	Devuelve datos de dirección en caracteres Latin I con transliteración alternativa.
Administración postal	Devuelve datos de direcciones en caracteres Latin I o ASCII, según lo que prefiera el proveedor de servicios de correo del país de destino.
Administración postal (Alt.)	Devuelve datos de dirección en Latin I o caracteres ASCII. Incluye caracteres alternativos, si el proveedor de servicios de correo en el país de destino admite los caracteres.
Conservar entrada	Devuelve datos de dirección en el conjunto de caracteres que utiliza la dirección de entrada.

La transliteración puede utilizar las representaciones numéricas de cada carácter de un conjunto de caracteres cuando convierte los caracteres para su procesamiento. La transliteración también puede convertir los caracteres fonéticamente cuando no hay representación numérica equivalente de un carácter. Si la transformación del validador de direcciones no puede asignar un carácter a UCS-2, lo convierte en un espacio.

**Nota:** Si establece un idioma y un script preferidos para los datos de salida, compruebe que el idioma y la codificación de caracteres que seleccione sean compatibles.

## Rangos para expandir

Determina cómo la transformación del validador de direcciones devuelve direcciones sugeridas para una dirección de calle que no especifica un número de domicilio. Use la propiedad cuando la transformación se ejecute en el modo de lista de sugerencias.

La transformación del validador de direcciones lee una dirección de calle parcial o incompleta en el modo de lista de sugerencias. La transformación compara la dirección con los datos de referencia de direcciones y devuelve todas las direcciones similares para el usuario final. Si la dirección de entrada no contiene un número de domicilio, la transformación puede devolver una o más sugerencias de número de domicilio para la calle. La propiedad Rangos para expandir determina cómo devuelve la transformación las direcciones.

La transformación puede devolver el intervalo de números de domicilio válidos en una sola dirección, o puede devolver una dirección independiente para cada número de domicilio válido. La transformación también puede devolver una dirección para cada número del intervalo del número de domicilio más bajo al más alto de la calle.

La siguiente tabla describe las opciones de la propiedad:

Opción	Descripción
Todos	La validación de direcciones devuelve una dirección sugerida para cada número de domicilio en el rango de posibles de números de domicilio de la calle.
Ninguno	La validación de direcciones devuelve una dirección única que identifica el número de domicilio más bajo y el más alto del intervalo válido para la calle.
Solo con elementos válidos	La validación de direcciones devuelve una dirección sugerida para cada número de domicilio que los datos de referencia de direcciones reconocen como una dirección de entrega.

**Nota:** El modo de lista de sugerencias puede usar otros elementos de la dirección para especificar el intervalo válido de números de calle. Por ejemplo, un código postal puede identificar la manzana que contiene el buzón de la dirección. La transformación del validador de direcciones puede utilizar el código postal para identificar el número de domicilio más bajo y el más alto de la manzana.

Si la transformación no puede determinar un intervalo de números de domicilio dentro de los límites prácticos, el número de direcciones sugeridas puede alcanzar un tamaño inutilizable. Para restringir el número de direcciones que la propiedad Rangos para expandir genera, habilite la propiedad Expansión de rangos flexible.

## Estandarizar direcciones no válidas

Determina si el proceso de validación de direcciones estandariza los valores de datos de una dirección no válida para la entrega. La propiedad se aplica a registros de direcciones que devuelven un estado de código de coincidencia en el intervalo de I1 a I4.

Seleccione la propiedad para simplificar la terminología en los registros de direcciones. Cuando se estandarizan los datos, los procesos de nivel inferior, como el análisis de duplicados, se ejecutan más eficientemente.

La validación de direcciones puede estandarizar los siguientes elementos de dirección:

- Elementos de sufijo de calle, como "Road" y "Boulevard".
- Elementos anteriores y posteriores de la dirección, como "norte", "sur", "este" y "oeste".

- Elementos de servicio de entrega, como "Apartado de correos".
- Elementos secundarios del edificio, como "apartamento", "planta" y "suite".
- Nombres de estado o provincia. La estandarización devuelve la forma abreviada de los nombres.

La siguiente tabla describe las opciones de la propiedad:

Opción	Descripción
Desactivada	La validación de direcciones no corrige errores de datos. Opción predeterminada.
Activada	La validación de direcciones corrige errores de datos.

## Nivel de seguimiento

Define la cantidad de detalles que se incluyen en el registro.

Puede configurar niveles de seguimiento de registros.

Configure la siguiente propiedad en la ficha **Avanzadas**:

### Nivel de seguimiento

Cantidad de detalles que aparecen en el registro de esta transformación. Puede elegir simplificado, normal, inicialización detallada o datos detallados. El valor predeterminado es Normal.

## Informes de certificación

Puede generar un informe para un proveedor de servicios de correo que certifique las operaciones de validación de direcciones de la transformación del validador de direcciones.

El informe certifica que las operaciones de validación de direcciones cumplen los estándares de la agencia de correos. Por ejemplo, el servicio postal de los Estados Unidos (USPS) puede certificar un motor de software que valida las direcciones de Estados Unidos a un nivel de código postal+4. La transformación del validador de direcciones valida direcciones de los Estados Unidos a un nivel de código postal+4 y puede proporcionar datos adicionales que ayudan a la clasificación y a la entrega.

La transformación del validador de direcciones genera informes para los siguientes estándares:

### Address Machine Approval System (AMAS)

Australia Post define el estándar de certificación AMAS.

### Coding Accuracy Support System (CASS)

USPS define el estándar de certificación CASS.

### SendRight

New Zealand Post define el estándar de certificación SendRight.

### Software Evaluation and Recognition Program (SERP)

Canada Post define el estándar de certificación SERP.

Un proveedor de servicios de correo puede ofrecer descuentos de precio si usa un software certificado para validar datos de direcciones.

Cuando se entregan artículos de correo al proveedor de servicios de correo, se envía el informe con la lista de direcciones de correo. El informe contiene datos sobre la organización. Los datos se introducen cuando se configura la transformación del validador de direcciones. La transformación escribe el archivo de informe en la ruta de acceso que se especifica.

**Nota:** Puede utilizar la transformación del validador de direcciones para certificar registros de direcciones francesas de acuerdo con el estándar de certificación del Servicio de administración de direcciones nacional (SNA). No genere un informe para la certificación de direcciones del SNA.

## Campos de informe AMAS

Cuando se configura un informe AMAS, se proporciona información acerca de la organización que envía el conjunto de registros de direcciones certificadas a Australia Post. Guarde o imprima el informe, e incluya el informe con los registros de direcciones que envíe a Australia Post.

Utilice la vista **Informes** para introducir la información.

La siguiente tabla describe la información que se especifica:

Campo	Descripción
Nombre del archivo de informe	Nombre y ubicación del informe creado por la validación de direcciones. De forma predeterminada, el validador de direcciones crea el informe en el directorio <code>bin</code> del equipo del servicio de integración de datos.  Para grabar el archivo de informe en otra ubicación del equipo del servicio de integración de datos, especifique la ruta de acceso al archivo y el nombre del archivo. Puede especificar una ruta de acceso completa o una ruta de acceso relativa. La ruta de acceso relativa utiliza el directorio <code>bin</code> como directorio raíz. El directorio especificado debe existir antes de ejecutar la asignación de validación de direcciones.
Nombre de lista de direcciones	Nombre del conjunto de registros de direcciones que envíe a Australia Post.
Nombre de procesador de lista	Nombre de la organización que envía el conjunto de registros de direcciones.
Nombre del administrador/propietario de la lista	Nombre del administrador o propietario de los datos de direcciones de la organización.
Número de teléfono	Número de teléfono de contacto de la organización que envía el conjunto de registros de direcciones.
Dirección	Dirección de la organización que envía el conjunto de registros de direcciones.

## TEMAS RELACIONADOS

- [“Cómo definir un informe de certificación” en la página 51](#)

## Campos de informe CASS

Cuando se configura un informe CASS, se proporciona información acerca de la organización que envía el conjunto de registros de direcciones certificadas a USPS. Guarde o imprima el informe, e incluya el informe con los registros de direcciones que envíe a USPS.

Utilice la vista **Informes** para introducir la información.

La siguiente tabla describe la información que se especifica:

Campo	Descripción
Nombre del archivo de informe	<p>Nombre y ubicación del informe creado por la validación de direcciones. De forma predeterminada, el validador de direcciones crea el informe en el directorio <code>bin</code> del equipo del servicio de integración de datos.</p> <p>Para grabar el archivo de informe en otra ubicación del equipo del servicio de integración de datos, especifique la ruta de acceso al archivo y el nombre del archivo. Puede especificar una ruta de acceso completa o una ruta de acceso relativa. La ruta de acceso relativa utiliza el directorio <code>bin</code> como directorio raíz.</p> <p>El directorio especificado debe existir antes de ejecutar la asignación de validación de direcciones.</p>
Nombre/ID de lista	Nombre o número de identificación de la lista de direcciones que se envía al proveedor de servicios de correo.
Nombre de procesador de lista	Nombre de la organización que realiza la validación de direcciones.
Nombre/Dirección	Nombre y dirección postal de la organización que realiza la validación de direcciones.

## TEMAS RELACIONADOS

- [“Cómo definir un informe de certificación” en la página 51](#)

## Informe SendRight

Cuando se configura un informe SendRight, se proporciona información acerca de la organización que envía el conjunto de registros de direcciones certificadas a New Zealand Post. Guarde o imprima el informe, e incluya el informe con los registros de direcciones que envíe a New Zealand Post.

Utilice la vista **Informes** para introducir la información.

La siguiente tabla describe la información que se especifica:

Campo	Descripción
Nombre del cliente	Nombre de la organización que envía el conjunto de registros de direcciones.
Número de NZP del cliente	Número de la cuenta de New Zealand Post de la organización que envía el conjunto de registros de direcciones. Si un servicio de correo envía los registros en nombre de la organización, especifique el número de identificación de transporte (TPID) del servicio de correo.
Base de datos del cliente	Nombre del archivo que contiene el conjunto de registros de direcciones. La transformación del validador de direcciones crea el informe en la ubicación que se especifique en el servicio de administración de contenido. Use Administrator Tool para definir la ubicación.
Dirección de cliente	Dirección de la organización que envía el conjunto de registros de direcciones.

## Campos de informe SERP

Cuando se configura un informe SERP, se proporciona información acerca de la organización que envía el conjunto de registros de direcciones certificadas a Canada Post. Guarde o imprima el informe, e incluya el informe con los registros de direcciones que envíe a USPS.

Utilice la vista **Informes** para introducir la información.

La siguiente tabla describe la información que se especifica:

Campo	Descripción
Nombre del archivo de informe	Nombre y ubicación del informe creado por la validación de direcciones. De forma predeterminada, el validador de direcciones crea el informe en el directorio <code>bin</code> del equipo del servicio de integración de datos.  Para grabar el archivo de informe en otra ubicación del equipo del servicio de integración de datos, especifique la ruta de acceso al archivo y el nombre del archivo. Puede especificar una ruta de acceso completa o una ruta de acceso relativa. La ruta de acceso relativa utiliza el directorio <code>bin</code> como directorio raíz. El directorio especificado debe existir antes de ejecutar la asignación de validación de direcciones.
Número de cliente de CPC	Número de cliente emitido por Canada Post Corporation a la organización que realiza la validación de direcciones.
Nombre/dirección del cliente	Nombre y dirección de la organización que realiza la validación de direcciones.

#### TEMAS RELACIONADOS

- [“Cómo definir un informe de certificación” en la página 51](#)

## Cómo configurar una transformación del validador de direcciones.

Utilice una transformación del validador de direcciones para validar y mejorar la calidad de los datos de direcciones postales.

La transformación del validador de direcciones lee datos de referencia de direcciones. Compruebe que Developer Tool puede acceder a los archivos de datos de referencia necesarios.

1. Abra la transformación.
2. Haga clic en la vista **Configuración general** y configure las propiedades generales.
3. Haga clic en la vista **Plantillas** para añadir puertos de entrada y de salida.
4. Haga clic en la vista **Informes** para generar informes para una certificación de dirección de servicio postal.
5. Haga clic en la vista **Avanzadas** para configurar las propiedades avanzadas de validación de direcciones.
6. Conecte los puertos de entrada y de salida.

**Nota:** Conecte los puertos de entrada que no desee que valide la transformación del validador de direcciones en el grupo de puertos de entrada **Transferencia**.

# Cómo añadir puertos a la transformación del validador de direcciones

Utilice la vista **Plantillas** para añadir puertos a la transformación del validador de direcciones.

1. Haga clic en la vista **Plantillas**.
2. Expanda una plantilla.
  - Elija la plantilla **Modelo básico** para añadir campos de dirección comunes.
  - Elija la plantilla **Modelo avanzado** para añadir campos de dirección especializados.
3. Expanda el grupo de puertos de entrada que se corresponde con el formato de los datos de entrada. Los grupos de puertos de entrada son **Discreto**, **Varias líneas** e **Híbrido**.
4. Seleccione puertos de entrada.  
**Sugerencia:** Haga clic en la tecla **CTRL** para seleccionar varios puertos.
5. Haga clic con el botón derecho en los puertos y seleccione **Añadir puerto a transformación**.
6. Expanda el grupo de puertos de salida que contiene los campos que necesita.
7. Haga clic con el botón derecho en los puertos y seleccione **Añadir puerto a transformación**.
8. Para añadir puertos de transferencia a columnas que no se desean validar, haga clic en la ficha **Puertos**, seleccione el grupo de puertos de entrada **Transferencia** y haga clic en **Nuevo**.

## Cómo crear plantillas definidas por el usuario

Cree plantillas para agrupar los puertos de dirección que prevé reutilizar.

Puede crear plantillas personalizadas seleccionando puertos desde las plantillas Básica y Avanzada. Puede seleccionar las plantillas personalizadas cuando se crean transformaciones del validador de direcciones subsiguientes.

**Nota:** Las plantillas no son objetos del repositorio. Las plantillas residen en el equipo que se utiliza para crearlas.

1. Seleccione la vista **Plantillas**.
2. Haga clic en **Nueva**.
3. Especifique un nombre para la plantilla..
4. Expanda la plantilla **Modelo básico** o **Modelo avanzado** y seleccione los puertos que necesite.
5. Haga clic en **Aceptar**.

# Cómo definir modelos del validador de direcciones

Los modelos del validador de direcciones definen los puertos de entrada y de salida para las transformaciones del validador de direcciones.

Las transformaciones del validador de direcciones no contienen puertos de entrada y de salida predeterminados. Sin embargo, puede definir un modelo para especificar los puertos de entrada y de salida que utilizan las transformaciones del validador de direcciones.

**Nota:** Los modelos no son objetos del repositorio. Los modelos residen en el equipo que se utiliza para crearlos.

Para definir un modelo del validador de direcciones, realice los siguientes pasos:

1. Seleccione la vista **Plantillas**.
2. Expanda la plantilla **Modelo básico** o **Modelo avanzado** y seleccione los puertos que necesite.
3. Seleccione **Crear modelo de validador de direcciones predeterminado mediante los puertos seleccionados**.
4. Para restablecer el modelo y quitar todos los puertos, seleccione **Borrar modelo de validador de direcciones predeterminado**.

## Cómo definir un informe de certificación

Cuando se define un informe de certificación en la transformación del validador de direcciones, se configuran opciones en las vistas **Configuración general** e **Informes**.

1. En la vista **Configuración general**, establezca la opción **Modo enCertificado**.
2. En la vista **Informes**, seleccione el tipo de informes que generar. Se pueden seleccionar los siguientes tipos de informe:

Opción	Descripción
Informe AMAS	Contiene información que Australia Post requiere para procesar el conjunto de registros.
Informe CASS	Contiene información que USPS requiere para procesar el conjunto de registros.
Informe SendRight	Contiene información que New Zealand Post requiere para procesar el conjunto de registros.
Informe SERP	Contiene información que Canada Post requiere para procesar el conjunto de registros.

3. Especifique los detalles del informe para cada tipo de informe seleccionado.

Envíe el archivo de informe al proveedor de servicios de correo con la lista de registros de dirección que haya validado con la transformación del validador de direcciones.

### TEMAS RELACIONADOS

- [“Campos de informe AMAS” en la página 46](#)
- [“Campos de informe CASS” en la página 47](#)

- [“Campos de informe SERP” en la página 48](#)

## CAPÍTULO 3

# Transformación de excepción de registros incorrectos

Este capítulo incluye los siguientes temas:

- [Resumen de la transformación de excepción de registros incorrectos, 53](#)
- [Tipos de registros de salida de excepción de registros incorrectos, 54](#)
- [Flujo de proceso de administración de excepciones de registros incorrectos, 55](#)
- [Asignaciones de excepción de registros incorrectos, 56](#)
- [Puertos de excepción de registros incorrectos , 58](#)
- [Vista de configuración de excepción de registros incorrectos, 59](#)
- [Asignación problemática de excepción de registros incorrectos , 61](#)
- [Propiedades avanzadas de Transformación de excepción, 62](#)
- [Cómo configurar una transformación de excepción de registros incorrectos, 63](#)
- [Ejemplo de asignación de excepción de registros incorrectos, 64](#)

## Resumen de la transformación de excepción de registros incorrectos

La transformación de excepción de registros incorrectos es una transformación activa que lee la salida de los procesos de calidad de datos e identifica registros que requieren la revisión manual.

Configure una transformación de excepción de registros incorrectos para analizar la salida de un proceso que identifica los problemas de calidad de datos en los registros. A un registro que tiene un problema de calidad de datos que necesita más revisión se le llama una excepción.

La transformación de excepción de registros incorrectos recibe la entrada de otra transformación o de un objeto de datos en otra asignación. La entrada de la transformación de registros incorrectos debe contener uno o varios puertos de problemas de calidad que reciban las descripciones en texto de los problemas de calidad de datos. La entrada de la transformación de excepción también puede contener una puntuación de registro numérica que la transformación puede utilizar para determinar la calidad de datos de cada registro. Establezca un umbral de puntuación superior e inferior en la transformación de excepción para clasificar los registros como calidad correcta e incorrecta según la puntuación del registro. La transformación de excepción de registros incorrectos escribe las excepciones y la descripción de los problemas de calidad asociados en una tabla de registros incorrectos.

Por ejemplo, una organización necesita validar las direcciones del cliente antes de enviarle el correo. Un desarrollador crea una asignación que valida la ciudad, el estado y el código postal del cliente frente a las tablas de referencia con una transformación de etiquetador. La transformación de etiquetador valida los campos y añade una puntuación de registro para cada fila según los resultados. La transformación de etiquetador también añade el texto que describe los problemas de calidad para cada registro que tenga un error. La transformación de etiquetador añade el texto del problema de calidad como `ciudad no válida` o `código postal en blanco` para cada excepción. La transformación de excepción de registros incorrectos escribe los registros de los clientes que deberían revisarse de forma manual en la tabla de registros incorrectos. Los analistas de datos revisan y corrigen los registros incorrectos en Informatica Data Director para Data Quality.

## Tipos de registros de salida de excepción de registros incorrectos

La excepción de registros incorrectos examina las puntuaciones de registros de entrada para determinar la calidad de los registros. Devuelve los registros a grupos de salida diferentes

La transformación de excepción identifica los siguientes tipos de registros en función de cada puntuación de registros:

### Registros correctos

Registros con puntuaciones mayores o iguales al umbral superior. Los registros correctos son válidos y no necesitan revisarse. Por ejemplo, si configura el umbral superior como 90, cualquier registro con una puntuación de 90 o más no necesitará revisarse.

### Registros incorrectos

Los registros con puntuaciones inferiores al umbral superior y puntuaciones mayores o iguales que el umbral inferior. Los registros incorrectos son las excepciones que hay que revisar en Analyst Tool. Por ejemplo, cuando el umbral inferior es 40, cualquier registro con una puntuación de 40 a 90 necesita revisión manual.

### Registros rechazados

Los registros con puntuaciones inferiores al umbral inferior. Los registros rechazados no son válidos. De forma predeterminada, la transformación de excepción suelta registros rechazados del flujo de datos. Para este ejemplo, cualquier registro con una puntuación de 40 o inferior es un registro rechazado.

**Nota:** Si los campos de problemas de calidad tienen valores nulos, el registro no es una excepción. Cuando un problema de calidad contiene texto o contiene una cadena vacía, el registro es una excepción. Compruebe que un puerto de problema de calidad contenga valores nulos cuando un campo no tenga ningún error. Si los puertos de problema de calidad contienen espacios en blanco en lugar de valores nulos, la transformación de excepción marca cada registro como una excepción. Cuando un usuario necesita corregir los problemas en Analyst Tool, el usuario no puede filtrar las excepciones por problema de calidad de datos.

Cuando un registro tiene una puntuación inferior a cero o mayor que 100, la fila no es válida. El servicio de integración de datos registra un mensaje de error de que la fila no es válida y omite el procesamiento del registro.

Si no se conecta una puntuación de registros como entrada de la transformación de excepción, la transformación escribe todos los registros que contienen los problemas de calidad a la tabla de registros incorrectos.

Cuando se incluye la transformación de excepción de los registros incorrectos a una tarea de asignación, puede configurar una tarea humana en el mismo flujo de trabajo para incluir una revisión manual de las excepciones. La tarea humana se inicia cuando una tarea de asignación del flujo de trabajo finaliza. La tarea humana requiere que los usuarios accedan a Analyst Tool para solucionar los problemas de calidad. Un usuario puede actualizar los datos y cambiar el estado de la calidad de cada registro en la tabla de registros incorrectos.

## Flujo de proceso de administración de excepciones de registros incorrectos

La transformación de excepción recibe puntuaciones de registros de las transformaciones de calidad de datos y crea las tablas que contienen los registros con distintos niveles de calidad de datos. Debe configurar las transformaciones de calidad de los datos para encontrar problemas de calidad y especificar una puntuación de registros para cada fila.

Puede configurar las transformaciones de calidad de datos en una asignación individual, o puede crear asignaciones en distintas etapas de los procesos de calidad de datos.

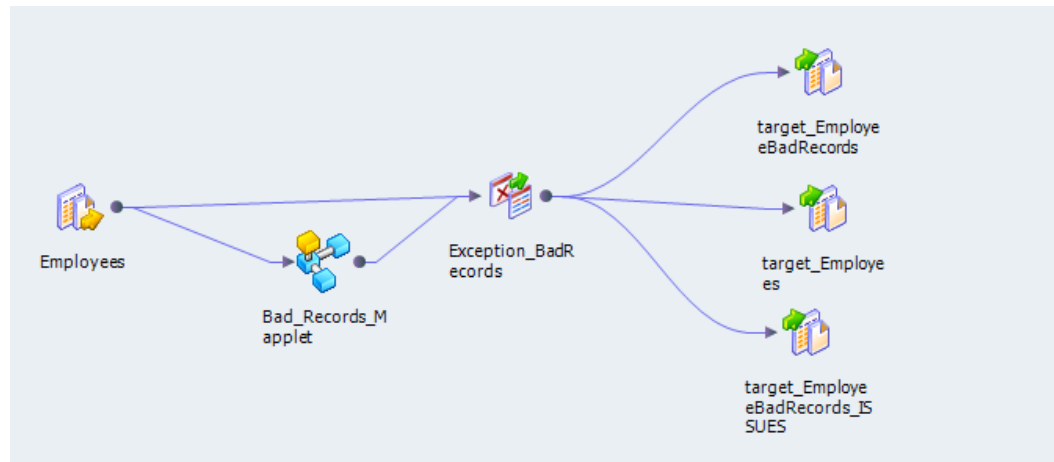
Realice las siguientes tareas de administración de excepciones de registros incorrectos:

1. En la herramienta Developer, defina las transformaciones que generan los valores de puntuación para datos de origen según los problemas de calidad de datos que se definan. Defina las transformaciones que devuelven el texto para describir la calidad de los datos de origen. Por ejemplo, puede configurar una transformación de etiquetador para comprobar los datos de origen en las tablas de referencia y luego escribir una etiqueta descriptiva para cada comparación. Puede definir una regla IF/THEN en una transformación de decisión para examinar un campo de datos. Puede definir varias transformaciones y maplets que realicen diferentes operaciones de calidad de los datos.
2. Configure una transformación de excepción para analizar las puntuaciones del registro que recibe de las operaciones de calidad de datos. Configure la transformación para escribir los registros en las tablas de base de datos según los valores de puntuación de los registros. Puede crear tablas independientes para los registros correctos, registros incorrectos, problemas de calidad y registros rechazados.
3. Asigne un puerto de problema de calidad a cada puerto de entrada que pueda contener datos incorrectos.
4. Opcionalmente, configure los objetos de datos de destino para los registros correctos e incorrectos. Conecte los puertos de salida de la transformación de excepción a los objetos de datos de destino de la asignación.
5. Cree el objeto de datos de destino para los registros incorrectos. Elija generar una tabla de registros incorrectos y añádala a la asignación. Cuando se genere una tabla de registros incorrectos, Developer Tool también generará una tabla de problemas de calidad. Añada la tabla de problemas de calidad a la asignación.
6. Añada la asignación a un flujo de trabajo.
7. Configure una tarea humana para asignar la revisión manual de registros incorrectos a los usuarios. Los usuarios pueden revisar y actualizar los registros incorrectos en Data Director para Data Quality.

# Asignaciones de excepción de registros incorrectos

Cuando cree una asignación que identifique excepciones de registros incorrectos, configure la asignación para escribir en uno o más destinos de la base de datos según la calidad de los datos en los registros.

La siguiente figura muestra un ejemplo de asignación de excepción de registros incorrectos:



La asignación contiene los siguientes objetos:

## Origen de datos

Un origen de datos de empleados que contiene los registros para analizar la calidad de datos.

## Bad\_Records\_Maplet

Bad\_Records\_Maplet contiene transformaciones que buscan los problemas de calidad y las puntuaciones de registros, y los añaden a los registros de origen. Las reglas son transformaciones que analizan los datos y buscan los problemas de calidad. Por ejemplo, puede incluir una transformación de etiquetador para comparar datos de entrada con tablas de referencia. Según los resultados, puede configurar la transformación de etiquetador para devolver los problemas de calidad como columnas adicionales en las filas. Puede configurar una transformación de decisión que utiliza las instrucciones IF, THEN, ELSE para examinar los datos y aplicar problemas de calidad y puntuaciones de registros a los datos de entrada.

## Transformación de excepción

La transformación de excepción determina qué registros se deben escribir en los destinos de datos, incluidas la tabla de registros incorrectos y la tabla de problemas.

## Destino de datos

De forma predeterminada, la transformación de excepción escribe todos los registros de calidad satisfactoria en este destino.

## Tabla de registros incorrectos

De forma predeterminada, la transformación de excepción escribe todos los registros de calidad incorrecta en este destino. Los registros incorrectos requieren revisión manual.

## Tabla de problemas

La transformación de excepción escribe problemas de calidad en otra tabla. Al visualizar los registros incorrectos en Data Director para Data Quality, la interfaz de usuario vincula los problemas de calidad a los registros incorrectos.

Opcionalmente, la transformación de excepción puede escribir los registros rechazados en una tabla de registros rechazados. Debe elegir crear un grupo de salida independiente para los registros rechazados en la vista **Configuración**.

## Problemas de calidad de excepción de registros incorrectos

Los problemas de calidad son cadenas de texto que describen el tipo de problema de calidad de datos que ha generado una puntuación de registros baja. La transformación de excepción de registros incorrectos recibe problemas de calidad asociados con cada fila del origen que contiene una puntuación de registros baja. Puede configurar distintos tipos de transformaciones que determinen problemas de calidad y puntuaciones de registros.

Por ejemplo, puede crear una transformación de decisión que examine el número de teléfono. La transformación de decisión genera la puntuación de registros y los problemas de calidad para el número de teléfono.

La siguiente estrategia de decisión identifica los números de teléfono de longitud incorrecta en una transformación de decisión:

```
IF LENGTH(Phone_Number) > 10 THEN
    Score:=50
    Phone_Quality_Issue:='Phone num too long'
ELSEIF LENGTH(Phone_Number) < 10 THEN
    Score:=50
    Phone_Quality_Issue:=' Phone num too short'
ELSE
    Score:=90
ENDIF
```

Cuando configure la transformación de excepción, debe asociar Phone\_Quality\_Issue con el puerto Phone\_Number. Los puertos son de diferentes grupos de entrada.

La transformación de excepción lee las puntuaciones generadas por la transformación de decisión y asigna registros con una puntuación de "50" al grupo de registros incorrectos de puertos de salida. Escribe Phone\_Quality\_Issue en el grupo de problemas de puertos de salida.

## Tareas humanas

Cuando configura un flujo de trabajo que contiene una transformación de excepción, incluya la asignación en una tarea de asignación. Puede añadir una tarea humana al mismo flujo de trabajo. La tarea humana requiere que uno o varios usuarios corrijan manualmente las excepciones en Informatica Data Director para Data Quality.

La tarea de asignación identifica registros en los datos de origen que contienen problemas de calidad de datos sin resolver. Los usuarios que realizan una tarea humana, utilizan la herramienta Informatica Data Director para Data Quality para solucionar los problemas y actualizar el estado de la calidad de datos de cada registro.

Cuando configure una tarea humana, puede crear una o más instancias de tarea y uno o más pasos de tarea. Una instancia de tarea representa el conjunto de datos en los que un usuario debe trabajar. Un paso de tarea representa el tipo de trabajo que un usuario realiza en los registros de su instancia de tarea. Puede crear varias instancias de tareas de modo que distintos usuarios trabajen en diferentes partes de los datos de Informatica Data Director para Data Quality.

Un usuario puede actualizar el estado de los registros incorrectos en Informatica Data Director para Data Quality en una de las siguientes maneras:

- Si un registro es válido, el usuario actualiza los metadatos de la tabla para confirmar el registro por el almacenamiento persistente en la base de datos.

- Si un registro no es válido, el usuario actualiza los metadatos de la tabla para eliminar dicho registro de la base de datos en una fase posterior del flujo de trabajo.
- Si el estado del registro no se confirma, el usuario actualiza los metadatos de la tabla de modo que el registro vuelve al flujo de trabajo para seguir procesándolo en una tarea de asignación.

Para obtener más información sobre las tareas humanas, consulte la *Guía de flujo de trabajo de Informatica Developer*.

## Puertos de excepción de registros incorrectos

Configure los puertos de entrada y salida en la ficha **Puertos** de la transformación de excepción de registros incorrectos.

La transformación de excepción de registros incorrectos contiene grupos de puertos de entrada y salida.

La siguiente figura muestra la ficha **Puertos**:

Puertos									
	Nombre	Tipo	Precisión	Escala	Entrada	Salida	Predeterminado	Descripción	
Entradas									
Datos (3)									
1	EmployeeID	string	10	0	<input checked="" type="checkbox"/>	<input type="checkbox"/>			
2	Employee_Name	string	10	0	<input checked="" type="checkbox"/>	<input type="checkbox"/>			
3	Phone_Number	string	10	0	<input checked="" type="checkbox"/>	<input type="checkbox"/>			
Problemas de calidad (3)									
1	EmployeeID_Quality	string	10	0	<input checked="" type="checkbox"/>	<input type="checkbox"/>			
2	Name_Quality_Issue	string	10	0	<input checked="" type="checkbox"/>	<input type="checkbox"/>			
3	Phone_Quality_Issue	string	10	0	<input checked="" type="checkbox"/>	<input type="checkbox"/>			
Control (1)									
1	Puntuación	double	15	0	<input checked="" type="checkbox"/>	<input type="checkbox"/>			
Salidas									
Salida estándar (4)									
1	Puntuación	double	15	0	<input type="checkbox"/>	<input checked="" type="checkbox"/>			
2	EmployeeID	string	10	0	<input type="checkbox"/>	<input checked="" type="checkbox"/>			
3	Employee_Name	string	10	0	<input type="checkbox"/>	<input checked="" type="checkbox"/>			
4	Phone_Number	string	10	0	<input type="checkbox"/>	<input checked="" type="checkbox"/>			
Registros incorrectos (6)									
1	Workflow_ID	string	64	0	<input type="checkbox"/>	<input checked="" type="checkbox"/>			
2	Row_Identifier	bigint	19	0	<input type="checkbox"/>	<input checked="" type="checkbox"/>			
3	Record_Status	string	20	0	<input type="checkbox"/>	<input checked="" type="checkbox"/>			
4	EmployeeID	string	10	0	<input type="checkbox"/>	<input checked="" type="checkbox"/>			
5	Employee_Name	string	10	0	<input type="checkbox"/>	<input checked="" type="checkbox"/>			
6	Phone_Number	string	10	0	<input type="checkbox"/>	<input checked="" type="checkbox"/>			
Problemas (2)									
1	Workflow_ID	string	64	0	<input type="checkbox"/>	<input checked="" type="checkbox"/>			
2	Row_Identifier	bigint	19	0	<input type="checkbox"/>	<input checked="" type="checkbox"/>			

## Puertos de entrada de transformación de excepción de registros incorrectos

Una transformación de excepción de registros incorrectos tiene distintos grupos de entrada para los datos, los problemas de calidad y la puntuación de registros.

La transformación de excepción de registros incorrectos incluye los siguientes grupos de entrada:

### Datos

Los campos de datos de origen.

### Problemas de calidad

Contiene puertos que describen los problemas de calidad de los registros. Los puertos de problemas de calidad contienen cadenas como "Excess\_Characters" o "Bad\_Data\_Format". Puede tener varios problemas de calidad en cada registro. La transformación no asociará los puertos del grupo de problemas de calidad a los campos de datos de origen hasta que se asignen los problemas a los puertos de datos en la vista **Asignación problemática**.

### Control

La puntuación de registros. La transformación de excepción analiza la puntuación de registros para determinar si las filas de entrada son las excepciones. Si no se conecta el puerto de puntuación, la transformación de excepción identifica una fila como una excepción si un puerto de problemas de calidad contiene datos.

## Salida de transformación de excepción de registros incorrectos

Una transformación de excepción de registros incorrectos tiene varios grupos de salida.

La transformación de excepción de registros incorrectos tiene los siguientes grupos de salida:

### Salida estándar

Los registros de calidad satisfactoria que no es necesario examinar para detectar problemas de calidad de datos.

Cada registro del grupo de salida estándar contiene un puerto de puntuación que representa la calidad de datos del registro.

### Registros incorrectos

Las excepciones que necesita examinar para detectar problemas de calidad de datos.

Cada registro del grupo de registros incorrectos contiene un ID de flujo de trabajo, un identificador de fila y un puerto de estado del registro.

### Problemas

Los problemas de calidad para los registros del grupo de registros incorrectos. Los problemas de calidad son los metadatos que Informatica Data Director para Data Quality muestra cuando se revisan manualmente los registros incorrectos.

Cada registro del grupo de problemas contiene un ID de flujo de trabajo y un puerto identificador de filas que identifica a qué fila de registros incorrectos pertenecen los problemas.

### Registros rechazados

Grupo opcional que contiene los registros que pueden eliminarse de la base de datos. Cada registro del grupo de registros rechazados contiene una puntuación de registros baja en el puerto de puntuación.

## Vista de configuración de excepción de registros incorrectos

La vista **Configuración** define los umbrales de puntuación superior e inferior del registro en la transformación de excepción. También define dónde escribe la transformación de excepción los registros correctos, registros incorrectos y registros rechazados.

La siguiente figura muestra la vista **Configuración** de la transformación de excepción:

**Umbral inferior :**  10,00

**Umbral superior :**  90,00

---

Opciones de distribución de datos

Tipo	Salida estándar	Tabla de registros incorrectos
Registros correctos (sobre el umbral superior)	<input checked="" type="checkbox"/>	<input type="checkbox"/>
Registros incorrectos (entre umbrales)	<input type="checkbox"/>	<input checked="" type="checkbox"/>
Registros rechazados (por debajo del umbral inferior)	<input type="checkbox"/>	<input type="checkbox"/>

☐ Crear un grupo de salida aparte para los registros rechazados

Generar tablas de registros incorrectos

[illegible]☐ Crear un grupo de salida aparte para los registros rechazados

Generar tablas de registros incorrectos

Puede configurar las siguientes propiedades en la vista **Configuración**:

**Umbral inferior**

Límite inferior para el intervalo de puntuaciones de registros incorrectos. La transformación procesa registros con puntuaciones inferiores a este valor como registros rechazados.

**Umbral superior**

Límite superior para el intervalo de puntuaciones de registros incorrectos. La transformación procesa registros con puntuaciones superiores o iguales a este valor como registros correctos.

### Opciones de distribución de datos

Los tipos de registros de salida. La configuración predeterminada distribuye los registros correctos a la salida estándar y los registros incorrectos a la tabla de registros incorrectos, y descarta los registros rechazados. Puede cambiar el grupo que recibe cada tipo de registro.

### Salida estándar

Los tipos de registro que la transformación escribe en los puertos de salida estándar.

El valor predeterminado es Registros correctos.

### Tabla de registros incorrectos

Los tipos de registro que la transformación escribe en los puertos de salida de registros incorrectos. El valor predeterminado es Registros incorrectos.

### Crear un grupo de salida aparte para los registros rechazados

Crea un grupo de salida aparte para los registros rechazados. Si el valor predeterminado está deshabilitado, la transformación de excepción no escribe registros rechazados en una tabla de base de datos.

### Generar tabla de registros incorrectos

Crea un objeto de base de datos que contiene datos de registros incorrectos. Cuando se selecciona esta opción, la transformación de excepción crea el objeto de la base de datos, añade el objeto al repositorio de modelos y añade una instancia del objeto al lienzo de la asignación. Puede generar la

tabla de registros incorrectos para una instancia de transformación de excepción en una asignación. Al generar la tabla de registros incorrectos, Developer Tool también crea la tabla de problemas de registros incorrectos.

**Nota:** Developer Tool añade un sufijo de 12 caracteres a cada nombre de columna de las tablas. Si usa una base de datos de Oracle, el nombre de la columna de origen debe ser de 18 caracteres o menos. Las tablas de base de datos deben estar en un esquema predeterminado.

## Cómo generar la tabla de registros incorrectos y la tabla de problemas

Cuando la transformación está en una asignación, puede generar la tabla de registros incorrectos y la tabla de problemas. Cuando el usuario genera las tablas, Developer Tool añade las tablas del repositorio de modelos.

1. Haga clic en **Generar tabla de registros incorrectos** para generar la tabla.  
Los objetos también se añaden a MRS, de modo que estará visible en los objetos de datos físicos en el proyecto.  
A continuación, aparece el cuadro de diálogo **Crear objeto de datos relacionales**.
2. Busque y seleccione una conexión a la base de datos donde guardar la tabla.
3. Especifique un nombre para la tabla de registros incorrectos de la base de datos.  
La tabla de problemas recibe el mismo nombre que la tabla de registros incorrectos con "\_Issue" añadido al nombre.
4. Especifique un nombre para el objeto de la tabla de registros incorrectos del repositorio de modelos.
5. Haga clic en **Finalizar**.  
Developer Tool añade las tablas al lienzo de la asignación y al repositorio de modelos. Puede ver las tablas como objetos de datos físicos en el proyecto. La asignación vincula las tablas a la transformación de excepción de registros incorrectos.

## Asignación problemática de excepción de registros incorrectos

Debe asignar puertos y prioridades a problemas de calidad de datos.

La siguiente figura muestra la vista **Asignación problemática**:

Asignación de problemas			
Asignar puertos y prioridades a problemas de calidad			
Problema de calidad	Entrada		
EmployeeID_Quality_Issue	EmployeeID	1	
Name_Quality_Issue	EmployeeName	1	
Phone_Quality_Issue		1	

La vista **Asignación problemática** contiene los siguientes campos:

### Problema de calidad

Cada puerto de problema de calidad que se define en el grupo de entrada de problemas de calidad aparece en la columna **Problema de calidad**.

### Entrada

La columna **Entrada** contiene los puertos de datos que se asignan a los problemas de calidad en la vista **Asignación problemática**. Asocie un puerto de entrada con cada puerto de problema de calidad. Cada puerto de entrada que contiene los datos de calidad incorrectos debe tener al menos un puerto de problema de calidad que indica un tipo de problema. Puede seleccionar el puerto Phone\_Number para Phone\_Quality\_Issue, por ejemplo. Puede asignar un puerto a más de un problema de calidad.

### Prioridad de problema

Las prioridades de problema determinan qué problemas tienen la mayor importancia a la hora de asignar el mismo puerto de entrada a varios problemas de calidad. Si se produce más de un problema para un campo de entrada, el servicio de integración de datos aplica el problema de calidad que tiene la prioridad más alta. En el caso de que exista más de un problema de calidad para un puerto de entrada y los problemas tengan todos la misma prioridad, el servicio de integración de datos aplica el problema de calidad que aparece primero en la lista. Especifique una prioridad entre 1 y 99, donde 1 representa la prioridad más alta.

Puede asignar prioridades de problema para filtrar los registros en Data Director para Data Quality.

## Asignar puertos a problemas de calidad

Asigne un puerto para asociarlo con cada problema de calidad. Developer Tool crea puertos en el grupo de salida de problemas para cada asociación que se añada a la vista **Asignación problemática**.

1. Para cada problema de calidad, haga clic en el campo **Entrada** para mostrar una lista de los puertos de entrada.
2. Seleccione un puerto de entrada para asociarlo con el problema de calidad.  
Puede elegir el mismo puerto para más de un problema.
3. Haga clic en la columna **Problema** y seleccione una prioridad para el problema de calidad.

## Propiedades avanzadas de Transformación de excepción

Configure las propiedades que determinan cómo el servicio de integración de datos procesa los datos para la transformación de excepción.

Puede configurar niveles de seguimiento de registros.

Configure la siguiente propiedad en la ficha **Avanzadas**:

### Nivel de seguimiento

Cantidad de detalles que aparecen en el registro de esta transformación. Puede elegir simplificado, normal, inicialización detallada o datos detallados. El valor predeterminado es Normal.

# Cómo configurar una transformación de excepción de registros incorrectos

Cuando configure una transformación de excepción de registros incorrectos, configure los puertos de entrada y los problemas de calidad que pueden aparecer en cada puerto. Defina los umbrales superior e inferior para determinar la calidad de los datos. Configure dónde escribir las excepciones y los registros rechazados.

1. Cree una transformación de excepción de registros incorrectos reutilizable o no reutilizable.

- Para crear una transformación reutilizable, seleccione **Archivo > Nuevo > Transformación**.
- Para crear una transformación no reutilizable, abra una asignación y añada la transformación en el lienzo de la asignación.

Seleccione una transformación de excepción de registros incorrectos.

2. Configure los puertos de entrada.

- Si crea una transformación reutilizable, seleccione la ficha **Puertos** y añada puertos para los datos que desea conectar a la transformación.
- Si crea una transformación no reutilizable, añada otros objetos al lienzo de la asignación y arrastre los puertos de entrada para la transformación.

3. Seleccione la vista **Configuración**.

4. Configure los umbrales de puntuación superior e inferior.

5. En la sección **Opciones de distribución de datos**, configure las propiedades estándar de la salida y de la tabla de excepciones para establecer dónde la transformación escribirá cada tipo de registro.

Configure dónde escribir los registros correctos, registros incorrectos y registros rechazados. Puede escribirlos en la salida estándar o en la tabla de registros incorrectos.

6. Abra la vista **Asignación problemática**. Asigne los problemas de calidad de datos a los puertos de datos.

Asignar una prioridad a cada problema. Si un puerto contiene valores con varios problemas, la transformación muestra el problema con mayor prioridad.

7. Seleccione la opción para generar una tabla de registros incorrectos. Introduzca la información de conexión de la base de datos y el nombre de la tabla. La tabla debe proceder del esquema predeterminado.

- Si genera una tabla de registros incorrectos, genere una tabla para los registros y una tabla adicional para los problemas de calidad de datos relacionados con los registros. La transformación crea un objeto de base de datos en el repositorio de modelos.

8. Conecte los puertos de salida de la transformación a uno o varios destinos de datos. Conecte los puertos de salida a los objetos de datos que correspondan a las opciones de salida que ha establecido en la vista **Configuración**.

- Si crea una transformación reutilizable, añada la transformación a una asignación y conecte los puertos de salida.
- Si crea una transformación no reutilizable, la transformación conecta los puertos a la tabla de registros incorrectos. Conecte los puertos de salida a cualquier otro destino de datos.

# Ejemplo de asignación de excepción de registros incorrectos

Una organización dirige un proyecto de datos para revisar nuevos datos de clientes. La organización necesita comprobar que los datos de contacto son válidos. El siguiente ejemplo muestra cómo definir una transformación de excepción de registros incorrectos que recibe los registros de un mapplet que realiza un análisis de calidad de datos de registros de los clientes.

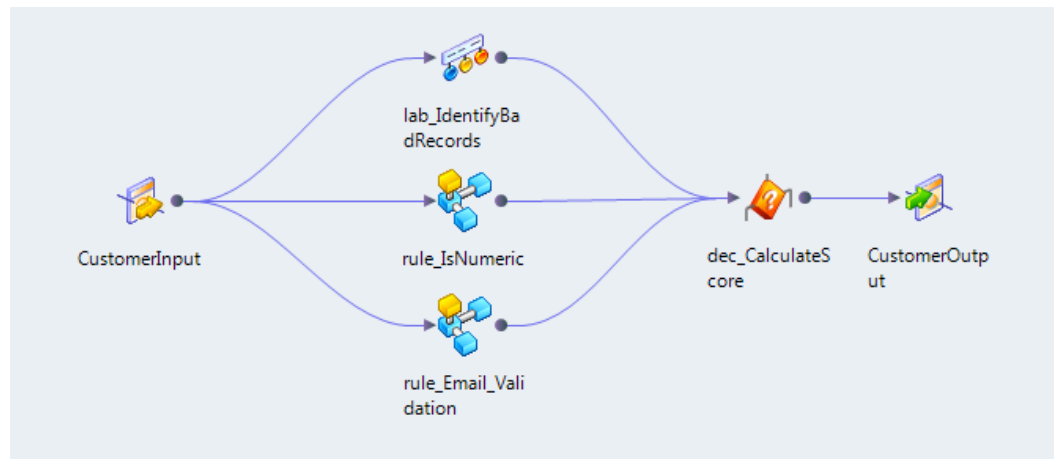
Cree un mapplet con transformaciones de calidad de datos que evalúen el formato y la exactitud de los datos de los clientes. El mapplet incluye transformaciones que generan una puntuación de registros según los resultados del análisis de calidad de datos. Las transformaciones también definen los problemas de calidad de los datos según los resultados del análisis.

## Mapplet de excepción de registros incorrectos

Cree un mapplet que contenga las transformaciones de calidad de datos para comprobar los valores de determinados campos. Las transformaciones comprueban las tablas de referencia y los conjuntos de contenido para determinar si los campos de los registros son válidos. Las transformaciones aplican una puntuación de registros para cada registro según los resultados. La transformación de excepción recibe los registros del mapplet y distribuye cada registro a la salida correspondiente según la puntuación de registros.

El mapplet se compone de transformaciones de etiquetador, transformaciones de decisión y transformaciones de expresión.

La siguiente figura muestra los objetos en el mapplet:



El mapplet realiza las siguientes tareas:

- Una transformación de etiquetador comprueba la localidad, el estado, el código de país, el código postal, y el código postal de los datos que recibe en los puertos de entrada. La transformación contiene una estrategia para cada puerto. Las estrategias comparan los datos de origen con las tablas de referencia e identifican los valores que no son válidos.
- Un mapplet de transformación de expresión comprueba que el número de teléfono es numérico y que contiene 10 dígitos.
- Una transformación de etiquetador y un mapplet de transformación de expresión comprueba que la dirección de correo electrónico es válida. La transformación de expresión comprueba la estructura de la cadena de correo electrónico. La transformación de etiquetador comprueba la dirección IP en una tabla de referencia de sufijos de dirección IP internacionales.

- Una transformación de decisión recibe la salida de la transformación y de los mapplets. Se calcula una puntuación de registros general para el registro de contactos del cliente.

Cree una asignación de excepción de registros incorrectos que incluya el mapplet. La asignación de excepción de registros incorrectos incluye una transformación de excepción que escribe las excepciones en una tabla de base de datos de registros incorrectos. Un analista de datos investiga y actualiza los registros de excepción en la tabla de registros incorrectos con Informatica Data Director para Data Quality.

## Grupos de entrada de ejemplo de excepción de registros incorrectos

La transformación de excepción tiene tres grupos de entrada. La transformación tiene un grupo de datos que recibe los datos de origen. Tiene el grupo de problemas de calidad, que recibe los problemas de calidad de los datos que las transformaciones de calidad de los datos encuentran. También tiene un grupo de control que contiene la puntuación de registros para la fila.

La siguiente figura muestra los grupos de entrada de la transformación de excepción:

	Nombre	Tipo	Precisión	Escala	Entrada	Salida
Entradas						
Datos (11)						
1	CUST_ID	decimal	20	0	<input checked="" type="checkbox"/>	<input type="checkbox"/>
2	COMPANY	string	100	0	<input checked="" type="checkbox"/>	<input type="checkbox"/>
3	CONTACT	string	100	0	<input checked="" type="checkbox"/>	<input type="checkbox"/>
4	TITLE	string	100	0	<input checked="" type="checkbox"/>	<input type="checkbox"/>
5	ADDR1	string	100	0	<input checked="" type="checkbox"/>	<input type="checkbox"/>
6	ADDR2	string	100	0	<input checked="" type="checkbox"/>	<input type="checkbox"/>
7	ADDR3	string	100	0	<input checked="" type="checkbox"/>	<input type="checkbox"/>
8	ADDR4	string	30	0	<input checked="" type="checkbox"/>	<input type="checkbox"/>
9	COUNTRY	string	20	0	<input checked="" type="checkbox"/>	<input type="checkbox"/>
10	PHONE	string	50	0	<input checked="" type="checkbox"/>	<input type="checkbox"/>
11	EMAIL	string	100	0	<input checked="" type="checkbox"/>	<input type="checkbox"/>
Problemas de calidad (6)						
1	CompanyStat...	string	30	0	<input checked="" type="checkbox"/>	<input type="checkbox"/>
2	LocalityStatus	string	30	0	<input checked="" type="checkbox"/>	<input type="checkbox"/>
3	ProvinceStatus	string	30	0	<input checked="" type="checkbox"/>	<input type="checkbox"/>
4	CountryStatus	string	30	0	<input checked="" type="checkbox"/>	<input type="checkbox"/>
5	ZipStatus	string	30	0	<input checked="" type="checkbox"/>	<input type="checkbox"/>
6	EmailStatus	string	30	0	<input checked="" type="checkbox"/>	<input type="checkbox"/>
Control (1)						
Salidas						

## Configuración de ejemplo de excepción de registros incorrectos

Defina los umbrales superior e inferior en la vista **Configuración**. Identifique dónde escribe la transformación los registros correctos, los registros incorrectos y los registros rechazados.

Acepte la configuración predeterminada para la distribución de los registros correctos, registros incorrectos y problemas.

La siguiente figura muestra la vista **Configuración** de la transformación de excepción:



Salida

Nombre: [Excepción.Bad\\_Output\\_DI](#)

	Workflow_ID	Row_Identifier	Record_Status	CUST_ID	COMPANY	CONTACT	TITLE	ADDR1	ADDR2
1	DummyWorkflowID	0	INVALID	1001590	E-AGENCY	OAKLAND	Federal Express ...	291 3RD STREET	2500 WESTERF...
2	DummyWorkflowID	1	INVALID	1001604	KPMG PEAT...	WOODCLIFF...	US Postal 2-day	530 CHESNUT...	"50 BROAD ST...
3	DummyWorkflowID	2	INVALID	1001622	INVESTEX	NEW YORK	United Parcel S...	"50 BROAD ST...	20TH FL"
4	DummyWorkflowID	3	INVALID	7121564	"ARTHURVA...	INC."	WHITEHOUSE...	Federal Express ...	310 BRIER ROAD
5	DummyWorkflowID	4	INVALID	7121569	RICE VOELKER	MANDEVILLE	Federal Express ...	3840 HIGHWAY...	"2ND & D STRE...
6	DummyWorkflowID	5	INVALID	1001658	HOUSEINFO...	WASHINGTON	Federal Express ...	"2ND & D STRE...	SW HOUSEA...
7	DummyWorkflowID	6	INVALID	1001694	BANK OF MO...	BOSTON	US Postal 2-day	"125 BROAD ST...	38TH FLOOR"
8	DummyWorkflowID	7	INVALID	1001738	"INTERFACE...	INC."	ANN ARBOR	United Parcel S...	5855 INTERFAC...
9	DummyWorkflowID	8	INVALID	1001798	NORTHWEST...	MILWAUKEE	US Postal 2-day	EAST MASON S...	EAST MASON S...
10	DummyWorkflowID	9	INVALID	1001735	TOPSPIN CR...	NEW YORK	Federal Express ...	"276 FIFTH AVE.	#902"

La tabla de registros incorrectos incluye todos los campos del registro de origen. Un registro incorrecto también incluye los siguientes campos:

#### Workflow\_ID

El nombre del flujo de trabajo que incluyó la transformación de excepción. El flujo de trabajo contiene la tarea de asignación de transformación de excepción y la tarea humana para revisar los problemas. El ID de flujo de trabajo contiene `DummyWorkflowID` si la transformación de excepción no está en un flujo de trabajo.

#### Row\_Identifier

Un número único que identifica cada fila.

#### Record\_Status

Un estado del registro del Informatica Data Director para Data Quality. Cada registro de la tabla de registros incorrectos recibe un estado `No válido`. Puede conservar el estado del registro al actualizar registros en Informatica Data Director para Data Quality.

## Tabla de problemas

La tabla de problemas contiene una fila para cada fila de la tabla de registros incorrectos. Cada fila contiene los problemas que el análisis de calidad de datos ha encontrado para el registro de origen.

La siguiente figura muestra las columnas de la tabla de problemas:

Salida				
Nombre: <a href="#">exc_BadRecord.Issues_DI</a>				
	Workflow_ID	Row_Identifier	CUST_ID	DQAPRIORITY_CUST_ID
1	DummyWorkflowID	0	6803 S. TUCSON WAY	1
2	DummyWorkflowID	1	6803 S. TUCSON WAY	1
3	DummyWorkflowID	2	20TH FL"	1
4	DummyWorkflowID	3	310 BRIER ROAD	1
5	DummyWorkflowID	4	767 FIFTH AVENUE	1
6	DummyWorkflowID	5	6803 S. TUCSON WAY	1
7	DummyWorkflowID	6	6803 S. TUCSON WAY	1
8	DummyWorkflowID	7	6803 S. TUCSON WAY	1
9	DummyWorkflowID	8	6803 S. TUCSON WAY	1
10	DummyWorkflowID	9	SW HOUSE ANNEX 2"	1
11	DummyWorkflowID	10	6803 S. TUCSON WAY	1

La tabla de problemas contiene las siguientes columnas:

#### Workflow\_ID

Identifica el flujo de trabajo que ha creado el registro. El flujo de trabajo contiene la tarea de asignación y la tarea humana de la transformación de excepción para revisar el problema.

#### Row\_Identifier

Identifica la fila del registro en la tabla de base de datos. El identificador de filas identifica qué fila en la tabla de registros incorrectos se corresponde con la fila en la tabla de problemas.

### Nombre de campo del problema

El nombre de campo es el nombre del campo que podría tener problemas de calidad. Cuando el campo contiene un error el valor de la columna es el texto del problema de calidad. En la figura anterior, el nombre de campo ADDR2 contiene el problema de calidad `invalid_locality`.

### DQAPriority

La prioridad del problema. Cuando se producen varios problemas para el mismo campo, el problema con la prioridad más alta aparece en el problema el nombre de campo del problema.

## Tabla de registros correctos

Cada registro de la tabla de registros correctos tiene una puntuación de registro mayor que el umbral superior. En este ejemplo, el umbral superior es 90.

La siguiente figura muestra los registros correctos que la transformación de excepción devuelve:

Salida							
Nombre: <a href="#">exc_BadRecord.Output_DI</a>							
	Score	CUST_ID	COMPANY	CONTACT	TITLE	ADDR1	ADDR2
1	<null>	<null>	<null>	1001658	HOUSEINFOR...	WASHINGTON	Federal Express 2-day
2	<null>	<null>	<null>	15951451	D E FREY & CO...	DENVER	Federal Express Overnight
3	<null>	<null>	<null>	15951453	SONANGOL CO...	HOUSTON	United Parcel Service Red
4	<null>	<null>	<null>	15951457	CITADEL INVES...	CHICAGO	United Parcel Service Intern
5	<null>	<null>	<null>	15951458	IOMEGA	ROY	US Postal International
6	<null>	<null>	<null>	15951461	VICTORY SBSF...	NEW YORK	US Postal International
7	<null>	<null>	<null>	15951465	WILLIAMSENE...	TULSA	United Parcel Service Blue
8	<null>	<null>	<null>	15951469	PAINE WEBBER	DALLAS	US Postal 2-day
9	<null>	<null>	<null>	15951475	PINKSHEETS.C...	NEW YORK	United Parcel Service Red
10	<null>	<null>	<null>	15951482	JAMES A LUSTIG	DENVER	United Parcel Service Blue

Los registros de la tabla de registros correctos contienen los campos de la puntuación de registro y los datos de origen.

## CAPÍTULO 4

# Transformación de asociación

Este capítulo incluye los siguientes temas:

- [Resumen de la transformación de asociación, 69](#)
- [Asignación de memoria, 70](#)
- [Propiedades avanzadas de transformación de asociación, 71](#)

## Resumen de la transformación de asociación

La transformación de asociación procesa datos de salida de una transformación de coincidencia. Crea vínculos entre registros duplicados que se asignan a distintos clústeres de coincidencia, de modo que estos registros pueden asociarse juntos en operaciones de administración de datos principales y de consolidación de datos.

La transformación de asociación genera un valor **AssociationID** para cada fila en un grupo de registros asociados y escribe los valores de ID en un puerto de salida.

La transformación de consolidación lee la salida de la transformación de asociación. Utilice una transformación de consolidación para crear un registro principal basado en registros con valores de ID de asociación comunes.

La transformación de asociación acepta los valores de cadena y numéricos en los puertos de entrada. Si añade un puerto de entrada, la transformación convierte los valores de datos del puerto en cadenas.

El puerto de salida AssociationID escribe datos de enteros. La transformación puede escribir datos de cadenas en un puerto AssociationID si se ha configurado la transformación en una versión anterior de Informatica Data Quality.

### Ejemplo: Cómo asociar salidas de transformación de coincidencia

La siguiente tabla contiene tres registros que pueden identificar a un mismo individuo:

ID	Nombre	Dirección	Ciudad	Estado	Código postal	SSN
1	David Jones	100 Admiral Ave.	New York	NY	10547	987-65-4321
2	Dennis Jones	1000 Alberta Ave.	New Jersey	NY	-	987-65-4321
3	D. Jones	Admiral Ave.	New York	NY	10547-1521	-

Una operación de análisis de duplicados definida en una transformación de coincidencia no identifica los tres registros como duplicados por las siguientes razones:

- Si define una búsqueda de duplicados en datos de nombres y de direcciones, los registros 1 y 3 se identifican como duplicados, pero se omite el registro 2.
- Si define una búsqueda de duplicados en datos de nombres y números de seguridad social, los registros 1 y 2 se identifican como duplicados, pero se omite el registro 3.
- Si se define una búsqueda de duplicados en los tres atributos (nombre, dirección y número de seguridad social), puede que la transformación de coincidencia no identifique ninguno de los registros como una coincidencia.

La transformación de asociación enlaza datos de distintos clústeres de coincidencia, de modo que los registros que comparten un ID de clúster reciben un valor AssociationID común. En este ejemplo, los tres registros reciben el mismo AssociationID, tal como se muestra en la siguiente tabla:

ID	Nombre	Dirección	Ciudad	Estado	Código postal	SSN	ID de clúster de nombre y dirección	ID de clúster de nombre y SSN	ID de asociación
1	David Jones	100 Admiral Ave.	New York	NY	10547	987-65-4320	1	1	1
2	Dennis Jones	1000 Alberta Ave.	New Jersey	NY	-	987-65-4320	2	1	1
3	D. Jones	Alberta Ave.	New York	NY	10547-1521	-	1	2	1

Puede consolidar los datos de registros duplicados en la transformación de consolidación.

## Asignación de memoria

Puede definir la cantidad mínima de memoria caché que utiliza la transformación de asociación. El ajuste predeterminado es 400.000 bytes.

Defina el valor en la propiedad **Tamaño de la memoria caché** de la ficha **Avanzadas**.

El valor predeterminado representa la cantidad mínima de memoria que utiliza la transformación. La transformación de asociación intenta obtener un múltiplo del valor predeterminado en función del número de puertos que se asocian. La transformación utiliza esta fórmula para obtener la memoria caché:

$(\text{Número de puertos de asociación} + 1) \times \text{memoria caché predeterminada}$

Por ejemplo, si se configuran cuatro puertos de asociación, la transformación intenta asignar 3,2 millones de bytes, o 3,05 MB, a la memoria caché.

Si se cambia el ajuste predeterminado, la transformación no intenta obtener memoria adicional.

**Nota:** Si se especifica un valor de memoria caché menor que 65536, la transformación de asociación lee el valor en megabytes.

## Propiedades avanzadas de transformación de asociación

Configure propiedades para ayudar a determinar cómo el servicio de integración de datos procesa los datos para la transformación de asociación.

Puede configurar niveles de seguimiento de registros.

Configure la siguiente propiedad en la ficha **Avanzadas**:

### Nivel de seguimiento

Cantidad de detalles que aparecen en el registro de esta transformación. Puede elegir simplificado, normal, inicialización detallada o datos detallados. El valor predeterminado es Normal.

## CAPÍTULO 5

# Transformación de agregación

Este capítulo incluye los siguientes temas:

- [Resumen de la transformación de agregación, 72](#)
- [Cómo desarrollar una transformación de agregación, 72](#)
- [Puertos de transformación de agregación, 73](#)
- [Propiedades avanzadas de transformación de agregación, 73](#)
- [Memorias cachés de agregado, 74](#)
- [Expresiones de agregado, 74](#)
- [Puertos Agrupar por, 76](#)
- [Entrada ordenada para una transformación de agregación, 77](#)
- [Cómo crear una transformación de agregación no reutilizable, 79](#)
- [Cómo crear una transformación de agregación no reutilizable, 79](#)
- [Consejos para transformaciones de agregación, 80](#)
- [Solución de problemas de transformaciones de agregación, 80](#)

## Resumen de la transformación de agregación

Utilice la transformación de agregación para realizar cálculos de agregado, como promedios y sumas. El servicio de integración de datos realiza cálculos de agregado mientras lee y almacena grupos de datos y datos de fila en una memoria caché de agregado. La transformación de agregación es una transformación activa.

La transformación de agregación difiere de la transformación de expresión en que la primera se usa para realizar cálculos en grupos. La transformación de expresión solamente permite realizar cálculos fila por fila.

Cuando se usa el lenguaje de transformación para crear expresiones de agregado, se pueden usar cláusulas condicionales para filtrar filas, lo que proporciona una mayor flexibilidad que el lenguaje de SQL.

## Cómo desarrollar una transformación de agregación

Cuando se desarrolla una transformación de agregación se deben tener en cuenta factores, como el tipo de cálculos que se desea realizar, así como el rendimiento de la transformación.

Tenga en cuenta los siguientes factores cuando desarrolle una transformación de agregación:

- Las expresiones que se desea utilizar en puertos de salida para calcular datos.
- Si se desea utilizar o no memorias cachés de agregado para procesar los datos de transformaciones.
- Si se desea utilizar o no agrupar por puertos para definir grupos para agregaciones, en lugar de realizar la agregación en todos los datos de entrada. La agrupación por puertos puede aumentar el rendimiento.
- Si se desea utilizar o no la entrada ordenada, de modo que el servicio de integración de datos realice cálculos de agregado mientras lee las filas para un grupo.

## Puertos de transformación de agregación

Una transformación de agregación tiene distintos tipos de puerto que permiten realizar diversas tareas de transformación, como agregar datos y agrupar datos.

Una transformación de agregación tiene los siguientes tipos de puerto:

### **Entrada**

Recibe datos desde transformaciones de un nivel superior.

### **Salida**

Proporciona el valor de devolución de una expresión. La expresión puede incluir expresiones de no agregado y cláusulas condicionales. Puede crear varios puertos de salida agregados.

### **Transferencia**

Pasa los datos inalterados.

### **Variable**

Se usan para variables locales.

### **Agrupar por**

Indica cómo crear grupos. El puerto puede ser cualquier puerto de entrada, de entrada/salida, de salida o variable. Al agrupar datos, la transformación de agregación produce como salida la última fila de cada grupo salvo que se indique lo contrario.

## Propiedades avanzadas de transformación de agregación

Configure propiedades para ayudar a determinar cómo el servicio de integración de datos procesa los datos para la transformación de agregación.

En la tabla siguiente se describen las propiedades avanzadas para las transformaciones de agregación:

### **Directorio de la memoria caché**

Directorio local donde el servicio de integración de datos crea los archivos de memoria caché del índice y los archivos de memoria caché de datos. Compruebe que el directorio existe y que tiene suficiente espacio de disco para las memorias caché de agregado.

Si ha habilitado la agregación incremental, el servicio de integración de datos crea una copia de seguridad de los archivos cada vez que se ejecuta la asignación. El directorio de la memoria caché debe tener suficiente espacio de disco para dos conjuntos de archivos.

El valor predeterminado es el parámetro del sistema CacheDir.

#### **Tamaño de la memoria caché de datos**

Tamaño de la memoria caché de datos para la transformación. El valor predeterminado es Auto.

#### **Tamaño de la memoria caché de índice**

Tamaño de la memoria caché de índice para la transformación. El valor predeterminado es Auto.

#### **Entrada ordenada**

Indica que los datos de entrada se ordenan previamente por grupos. Seleccione esta opción solamente si la asignación pasa datos ordenados a la transformación de agregación.

#### **Nivel de seguimiento**

Cantidad de detalles que aparecen en el registro de esta transformación. Puede elegir simplificado, normal, inicialización detallada o datos detallados. El valor predeterminado es Normal.

## Memorias cachés de agregado

Cuando se ejecuta una asignación que utiliza una transformación de agregación, el servicio de integración de datos crea una memoria caché del índice y una memoria caché de datos en la memoria para procesar la transformación. Si el servicio de integración de datos requiere más espacio, almacena los valores de desbordamiento en los archivos de la memoria caché.

Puede configurar la memoria caché del índice y la memoria caché de datos en la transformación de agregación.

El servicio de integración de datos utiliza memoria para procesar una transformación de agregación con puertos ordenados. No utiliza la memoria caché. No necesita configurar la memoria caché para las transformaciones de agregación que utilizan puertos ordenados.

## Expresiones de agregado

Developer Tool solamente permite expresiones de agregado en la transformación de agregación.

Una expresión de agregado puede incluir cláusulas condicionales y funciones no agregadas. También puede incluir una función de agregado anidada dentro de otra función de agregado, como:

```
MAX( COUNT( ITEM ) )
```

El resultado de una expresión de agregado varía según los puertos agrupar por de la transformación. Por ejemplo, cuando el servicio de integración de datos calcula la siguiente expresión de agregado sin ningún puerto agrupar por definido, busca la cantidad total de artículos vendidos:

```
SUM( QUANTITY )
```

Sin embargo, si se utiliza la misma expresión y se agrupa el puerto ITEM, el servicio de integración de datos devuelve la cantidad total de artículos vendidos por artículo.

Se puede crear una expresión de agregado en cualquier puerto de salida y usar varios puertos de agregado en una transformación.

## Funciones de agregado

Utilice funciones de agregado dentro de una transformación de agregación. Puede anidar una función de agregado dentro de otra función de agregado.

El lenguaje de la transformación incluye las siguientes funciones de agregado:

- AVG
- COUNT
- FIRST
- LAST
- MAX
- MEDIAN
- MIN
- PERCENTILE
- STDDEV
- SUM
- VARIANCE

Debe utilizar estas funciones en una expresión dentro de una transformación de agregación.

## Funciones de agregado anidadas

Se pueden incluir varias funciones de un nivel o varias funciones anidadas en distintos puertos de la transformación de agregación

No se pueden incluir tanto funciones de un nivel y funciones anidadas en una función de agregación. Por lo tanto, si una transformación de agregación contiene una función de un nivel en cualquier puerto de salida, no se puede utilizar una función anidada en ningún otro puerto de esa transformación. Cuando se incluyen funciones de un nivel y funciones anidadas en la misma transformación de agregación, Developer Tool marca la asignación o el maplet como no válidos. Si necesita crear funciones de un nivel y funciones anidadas al mismo tiempo, cree transformaciones de agregación individuales.

## Cláusulas condicionales en expresiones de agregado

Utilice cláusulas condicionales en la expresión de agregado para reducir el número de filas utilizadas en la agregación. La cláusula condicional puede ser cualquier cláusula cuya evaluación da como resultado TRUE o FALSE.

Por ejemplo, utilice la siguiente expresión para calcular el total de las comisiones de los empleados que hayan sobrepasado su cuota trimestral:

```
SUM( COMMISSION, COMMISSION > QUOTA )
```

## Puertos Agrupar por

La transformación de agregación permite definir grupos para agregaciones, en lugar de realizar la agregación de todos los datos de entrada. Por ejemplo, en lugar de buscar el total de ventas de la empresa, puede buscar el total de ventas agrupado por región.

Para definir un grupo para la expresión de agregado, seleccione los puertos de entrada, entrada/salida, salida y variables apropiados en la transformación de agregación. Puede seleccionar varias agrupaciones por puerto para crear un grupo nuevo para cada combinación única. Luego, el servicio de integración de datos realiza la agregación definida para cada grupo.

Cuando se agrupan valores, el servicio de integración de datos produce una fila para cada grupo. Si no se agrupan valores, el servicio de integración de datos devuelve una fila para todas las filas de entrada. Normalmente, el servicio de integración de datos devuelve la última fila de cada grupo (o la última fila recibida) con el resultado de la agregación. No obstante, si especifica que se devuelva una fila en concreto (por ejemplo, utilizando la función FIRST), el servicio de integración de datos devuelve la fila especificada.

Cuando se seleccionan varias agrupaciones por puerto en la transformación de agregación, el servicio de integración de datos utiliza el orden de los puertos para determinar el orden en que realizará la agrupación. Puesto que el orden de los grupos puede afectar a los resultados, ordene las agrupaciones por puertos para asegurar que se realiza la agrupación apropiada. Por ejemplo, los resultados de agrupar por ITEM\_ID y luego por QUANTITY pueden variar respecto a la agrupación por QUANTITY y luego por ITEM\_ID, ya que los valores numéricos para la cantidad no son necesariamente únicos.

Si envía los siguientes datos mediante esta transformación de agregación:

STORE_ID	ITEM	QTY	PRICE
101	'battery'	3	2.99
101	'battery'	1	3.19
101	'battery'	2	2.59
101	'AAA'	2	2.45
201	'battery'	1	1.99
201	'battery'	4	1.59
301	'battery'	1	2.45

El servicio de integración de datos lleva a cabo el cálculo del agregado en los siguientes grupos únicos:

STORE_ID	ITEM
101	'battery'
101	'AAA'
201	'battery'
301	'battery'

Luego, el servicio de integración de datos pasa la última fila recibida junto con los resultados de la agregación, tal como sigue:

STORE_ID	ITEM	QTY	PRICE	SALES_PER_STORE
101	'battery'	2	2.59	17.34
101	'AAA'	2	2.45	4.90
201	'battery'	4	1.59	8.35
301	'battery'	1	2.45	2.45

## Expresiones de no agregado

Utilice expresiones de no agregado en agrupar por puertos para modificar o reemplazar grupos.

Por ejemplo, si desea reemplazar 'pila AAA' antes de la agrupación, puede crear un puerto de salida agrupar por llamado CORRECTED\_ITEM utilizando la siguiente expresión:

```
IIF( ITEM = 'AAA battery', battery, ITEM )
```

## Valores predeterminados de puertos Agrupar por

Defina un valor predeterminado para cada puerto en el grupo para reemplazar valores de entrada nulos. Esto permite que el servicio de integración de datos incluya grupos de elementos nulos en la agregación.

# Entrada ordenada para una transformación de agregación

Puede mejorar el rendimiento de una transformación de agregación mediante la opción de entrada ordenada.

Cuando se utiliza la entrada ordenada, el servicio de integración de datos da por sentado que todos los datos están ordenados por grupo y realiza cálculos de agregado a medida que lee las filas de un grupo. En caso necesario, el servicio de integración de datos almacena la información sobre el grupo en la memoria. Para utilizar la opción Entrada ordenada, debe pasar datos ordenados a la transformación de agregación.

Si no se utiliza la entrada ordenada, el servicio de integración de datos realiza los cálculos de agregados a medida que lee las filas. Puesto que los datos no están ordenados, el servicio de integración de datos almacena datos para cada grupo hasta que lee el origen por completo para asegurarse de que todos los cálculos de agregado se realizan de forma precisa.

Por ejemplo, una transformación de agregación tiene los puertos agrupar por STORE\_ID e ITEM con la opción de entrada ordenada seleccionada. Cuando se pasan los siguientes datos a través de la agregación, el servicio de integración de datos realiza una agregación por cada tres filas en el grupo 101/batería cuando encuentra el grupo 201/batería:

STORE_ID	ITEM	QTY	PRICE
101	'battery'	3	2.99
101	'battery'	1	3.19

STORE_ID	ITEM	QTY	PRICE
101	'battery'	2	2.59
201	'battery'	4	1.59
201	'battery'	1	1.99

Si se utiliza la entrada ordenada y no se ordenan previamente los datos correctamente, se obtienen resultados inesperados.

## Condiciones de entrada ordenada

Hay ciertas condiciones que impiden que se utilice la entrada ordenada.

No utilice la entrada ordenada si se cumple una de las dos condiciones siguientes:

- La expresión de agregado utiliza funciones de agregado anidadas.
- La transformación utiliza agregación incremental.

Si utiliza la entrada ordenada y no ordena los datos correctamente, el servicio de integración de datos genera un error en la ejecución de la asignación.

## Cómo ordenar datos en una transformación de agregación

Para utilizar una entrada ordenada, pase datos ordenados por una transformación de agregación.

Debe ordenar los datos por los puertos agrupar por del agregador en el orden que aparecen en la transformación de agregación.

En el caso de entradas relacionales y de archivo sin formato, utilice la transformación de ordenación para ordenar los datos de la asignación antes de pasarlos a la transformación de agregación. Puede poner la transformación de ordenación en cualquier lugar de la asignación antes del agregador en el caso de que ninguna transformación cambie el orden de los datos ordenados. La agrupación por columnas en la transformación de agregación debe estar en el mismo orden que aparece en la transformación de ordenación.

La transformación de ordenación ordena los datos del siguiente modo:

ITEM_NO	ITEM_NAME	QTY	PRICE
345	Soup	4	2.95
345	Soup	1	2.95
345	Soup	2	3.25
546	Cereal	1	4.49
546	Cereal	2	5.25

Con la entrada ordenada, la transformación de agregación devuelve los siguientes resultados:

ITEM_NAME	QTY	PRICE	INCOME_PER_ITEM
Cereal	2	5.25	14.99

ITEM_NAME	QTY	PRICE	INCOME_PER_ITEM
Soup	2	3.25	21.25

## Cómo crear una transformación de agregación no reutilizable

Cree una transformación de unión no reutilizable para utilizarla en varias asignaciones o mapplets.

1. Seleccione un proyecto o una carpeta en la vista **Explorador de objetos**.
2. Haga clic en **Archivo > Nueva > Transformación**.  
A continuación, aparece el cuadro de diálogo **Nueva**.
3. Seleccione la transformación de agregación.
4. Haga clic en **Siguiente**.
5. Especifique un nombre para la transformación.
6. Haga clic en **Finalizar**.  
La transformación aparece en el editor.
7. Haga clic en el botón **Nueva** para añadir un puerto a la transformación.
8. Edite el puerto para definir el nombre, el tipo de datos y la precisión.
9. Determine el tipo de cada puerto: entrada, salida, transferencia o variable.
10. Configure expresiones de agregado para puertos de salida.
11. Haga clic en la vista **Avanzadas** y edite las propiedades de la transformación.

## Cómo crear una transformación de agregación no reutilizable

Cree una transformación de agregación no reutilizable como parte de una asignación o de un mapplet.

1. En una asignación o en un mapplet, arrastre una transformación de agregación desde la paleta Transformación hasta el editor.  
A continuación, la transformación aparece en el editor.
2. En la vista **Propiedades**, edite el nombre y la descripción de la transformación.
3. En la ficha **Puertos**, haga clic en el botón **Nuevo** para añadir puertos a la transformación.
4. Edite los puertos para definir el nombre, el tipo de datos y la precisión.
5. Determine el tipo de cada puerto: entrada, salida, transferencia o variable.
6. Configure expresiones de agregado para puertos de salida.
7. En la vista **Avanzadas**, edite las propiedades de la transformación.

# Consejos para transformaciones de agregación

Puede seguir estos consejos para utilizar las transformaciones de agregación de forma más eficaz.

## **Utilizar la entrada ordenada para reducir el uso de memorias cachés de agregado.**

La entrada ordenada reduce la cantidad de datos que se guardan en la memoria caché durante la ejecución de una asignación y aumenta el rendimiento. Utilice esta opción con la transformación de ordenación para pasar datos ordenados a la transformación de agregación.

## **Limitar el número de puertos de entrada/salida o de salida conectados.**

Limite el número de puertos de entrada/salida o de salida conectados para reducir la cantidad de datos que la transformación de agregación almacena en la memoria caché de datos.

## **Filtrar los datos antes de agregarlos.**

Si se utiliza una transformación de filtro en la asignación, colóquela antes de la transformación de agregación para reducir toda agregación innecesaria.

# Solución de problemas de transformaciones de agregación

Se pueden solucionar los problemas de transformaciones de agregación.

**He seleccionado la entrada ordenada, pero el flujo de trabajo tarda lo mismo que antes.**

No se puede usar la entrada ordenada si se cumple una de las siguientes condiciones:

- La expresión de agregado contiene funciones de agregado anidadas.
- Se utiliza la agregación incremental.
- Los datos de origen dependen de los datos.

Cuando se cumple alguna de estas condiciones, el servicio de integración de datos procesa la transformación del mismo modo que si no se utiliza la entrada ordenada.

**Una asignación con una transformación de agregación ralentiza el rendimiento.**

Es posible que el servicio de integración de datos esté paginando en el disco. En las propiedades de la transformación, puede aumentar el rendimiento incrementando los tamaños de la memoria caché de índices y datos.

## CAPÍTULO 6

# Transformación de conversión de mayúsculas y minúsculas

Este capítulo incluye los siguientes temas:

- [Resumen de la transformación de conversión de mayúsculas y minúsculas, 81](#)
- [Propiedades de la estrategia de distinción de mayúsculas y minúsculas, 82](#)
- [Cómo configurar una estrategia de conversión de mayúsculas y minúsculas, 82](#)
- [Propiedades avanzadas de Transformación de conversión de mayúsculas y minúsculas, 83](#)

## Resumen de la transformación de conversión de mayúsculas y minúsculas

La transformación de conversión de mayúsculas y minúsculas es una transformación pasiva que uniformiza los datos mediante la estandarización del formato de mayúsculas y minúsculas de las cadenas de los datos de entrada.

Para mejorar la calidad de los datos, utilice la transformación de conversión de mayúsculas y minúsculas para formatear los datos con formatos de carácter similares. Puede seleccionar tipos de conversión del formato de mayúsculas y minúsculas predefinidos como mayúsculas, minúsculas, formato alternativo, formato de título y formato de frase.

También puede usar tablas de referencia para controlar la conversión del formato de mayúsculas y minúsculas. Utilice tablas de referencia para cambiar el formato de mayúsculas y minúsculas de cadenas de entrada por los valores de la columna "Valid" de la tabla de referencia. Puede usar tablas de referencia cuando el tipo de conversión de formato de mayúsculas y minúsculas sea **Formato de mayúsculas y minúsculas de título** o **Formato de mayúsculas y minúsculas de frase**.

Dentro de una transformación de conversión de mayúsculas y minúsculas puede crear varias estrategias de conversión de mayúsculas y minúsculas. Cada estrategia utiliza un tipo de conversión individual. La transformación de conversión de mayúsculas y minúsculas proporciona un asistente que puede usarse para crear estrategias.

# Propiedades de la estrategia de distinción de mayúsculas y minúsculas

Se pueden configurar propiedades para estrategias de conversión de mayúsculas y minúsculas.

En la vista **Estrategias** puede configurar las siguientes propiedades para la conversión de mayúsculas y minúsculas:

## Tipo de conversión

Define el método de conversión de mayúsculas y minúsculas que utiliza una estrategia. Se pueden aplicar los siguientes tipos de conversión de mayúsculas y minúsculas.

- **Mayúsculas.** Convierte todas las letras a mayúsculas.
- **Tipo de frase.** Capitaliza la primera letra de la cadena de datos del campo.
- **Alternar mayúsculas y minúsculas.** Convierte las letras minúsculas en mayúsculas y viceversa.
- **Tipo de título.** Capitaliza la primera letra de cada subcadena.
- **Minúsculas.** Convierte todas las letras a minúsculas.

El método de conversión de mayúsculas y minúsculas predeterminado es mayúsculas.

## No modificar palabras en mayúscula

Reemplaza la capitalización elegida para las cadenas en mayúscula.

## Delimitadores

Define cómo funciona la capitalización en la conversión de mayúsculas y minúsculas. Por ejemplo, elija un guión como delimitador para transformar "smith-jones" en "Smith-Jones". El delimitador predeterminado es el carácter de espacio.

## Tabla de referencia

Aplica el formato de capitalización especificado por una tabla de referencia. Se aplica únicamente si la opción de conversión de mayúsculas y minúsculas es **Formato de mayúsculas y minúsculas de título** o **Formato de mayúsculas y minúsculas de frase**. Haga clic en **Nuevo** para añadir una tabla de referencia a la estrategia.

**Nota:** Si se produce una coincidencia de tabla de referencia al principio de un token, el siguiente carácter de ese token se cambia a mayúscula. Por ejemplo, si la cadena de entrada es mcdonald y la tabla de referencia tiene una entrada para Mc, la cadena de salida será McDonald.

# Cómo configurar una estrategia de conversión de mayúsculas y minúsculas

Para cambiar el formato de mayúsculas y minúsculas de las cadenas de entrada, configure los ajustes en la vista **Estrategias** de una transformación de conversión de mayúsculas y minúsculas.

1. Seleccione la vista **Estrategias**.
2. Haga clic en **Nueva**.  
Se abre el asistente **Nueva estrategia**.
3. Si lo desea, edite el nombre y la descripción de la estrategia.

4. Haga clic en los campos **Entradas y Salidas** para seleccionar los puertos de la estrategia.
5. Configure las propiedades de la estrategia. La estrategia de conversión predeterminada es **Mayúsculas**.
6. Haga clic en **Siguiente**.
7. Si lo desea, añada tablas de referencia para personalizar las opciones de mayúsculas y minúsculas de los datos de entrada que coinciden con las entradas de la tabla de referencia. La personalización del formato de mayúsculas y minúsculas de la tabla de referencia solamente se aplica a las estrategias de formato de mayúsculas y minúsculas de títulos y de frases.
8. Haga clic en **Finalizar**.

## Propiedades avanzadas de Transformación de conversión de mayúsculas y minúsculas

Configure las propiedades que ayudan a determinar cómo el servicio de integración de datos procesa los datos para la Transformación de conversión de mayúsculas y minúsculas.

Puede configurar niveles de seguimiento de registros.

Configure la siguiente propiedad en la ficha **Avanzadas**:

### Nivel de seguimiento

Cantidad de detalles que aparecen en el registro de esta transformación. Puede elegir simplificado, normal, inicialización detallada o datos detallados. El valor predeterminado es Normal.

## CAPÍTULO 7

# Transformación de clasificador

Este capítulo incluye los siguientes temas:

- [Resumen de la transformación de clasificador, 84](#)
- [Modelos de clasificador, 85](#)
- [Algoritmos clasificadores, 85](#)
- [Opciones de la transformación de clasificador, 85](#)
- [Estrategias de clasificador, 86](#)
- [Propiedades avanzadas de Transformación de clasificador, 86](#)
- [Cómo configurar una estrategia de clasificador, 87](#)
- [Ejemplo de análisis de clasificador, 87](#)

## Resumen de la transformación de clasificador

La transformación de clasificador es una transformación pasiva que analiza los campos de entrada e identifica el tipo de información de cada campo. Utilice una transformación de clasificador cuando los campos de entrada contengan varios valores de texto.

Cuando configure la transformación de clasificador, seleccione un modelo de clasificador y un algoritmo de clasificador. Un modelo de clasificador es un tipo de objeto de datos de referencia. Un algoritmo de clasificador es un conjunto de reglas que calcula el número de palabras similares en una cadena y la posición relativa de las palabras. La transformación compara el análisis del algoritmo con el contenido del modelo de clasificador. La transformación devuelve la clasificación del modelo que identifica el tipo de información dominante en la cadena.

La transformación de clasificador puede analizar cadenas de longitud considerable. Por ejemplo, puede utilizar la transformación para clasificar el contenido de mensajes de correo electrónico, mensajes de medios sociales y texto de documentos. Puede pasar el contenido de cada documento o mensaje a un campo en una columna del origen de datos, y conectar la columna a una transformación de clasificador. En cada caso, prepare el origen de datos de modo que cada campo incluya el contenido completo de un documento o de cadena que desee analizar.

# Modelos de clasificador

La transformación de clasificador utiliza un objeto de datos de referencia llamado modelo de clasificador para analizar datos de entrada. Debe seleccionar el modelo de clasificador cuando configure una transformación de clasificador. La transformación compara los datos de entrada con los datos del modelo de clasificador e identifica el tipo de información en cada campo de entrada.

El modelo de clasificador contiene las siguientes columnas:

## Columna de datos

Una columna que contiene las palabras y las frases que están probable que podrían existir en los datos de entrada. La transformación compara los datos de entrada con los datos de esta columna.

## Columna Etiqueta

Una columna que contiene etiquetas descriptivas que pueden definir la información de los datos. La transformación devuelve una etiqueta de esta columna como salida.

El modelo de clasificador también contiene datos de compilación que la transformación de clasificador utiliza para calcular el tipo de información correcta para los datos de entrada.

Puede crear un modelo de clasificador en Developer Tool. El repositorio de modelos almacena los metadatos para el objeto de modelo de clasificador. Los datos de columna y los datos de compilación se almacenan en un archivo en la estructura del directorio de Informatica.

# Algoritmos clasificadores

Cuando se añade un modelo clasificador a una estrategia de transformación, también debe seleccionar un algoritmo clasificador. El algoritmo determina cómo la transformación compara los datos del modelo de clasificador con los datos de entrada.

Puede seleccionar el algoritmo **Bayesiano Ingenuo** o el algoritmo **Entropía máxima**.

Tenga en cuenta los siguientes factores cuando seleccione un algoritmo:

- El algoritmo Entropía máxima realiza un análisis más exhaustivo que el algoritmo Bayesiano Ingenuo.
- Una asignación que utiliza el algoritmo Bayesiano Ingenuo se ejecuta más rápidamente que una asignación que utiliza el algoritmo Entropía máxima en los mismos datos.
- Seleccione el algoritmo Entropía máxima con el modelo de clasificador que Informatica incluye en el acelerador del núcleo.

# Opciones de la transformación de clasificador

La transformación de clasificador muestra opciones configurables en una serie de fichas o vistas en Developer Tool.

Cuando abre una transformación reutilizable, las opciones aparecen en una serie de fichas en el editor de transformaciones. Cuando abre una transformación no reutilizable en una asignación, las opciones aparecen en una serie de vistas en el editor de asignaciones. Seleccione la ficha de propiedades de asignación para ver las vistas en una transformación no reutilizable.

Puede seleccionar las siguientes vistas:

#### General

Ver y actualizar el nombre y la descripción de la transformación.

#### Puertos

Ver los puertos de entrada y salida en la transformación.

**Nota:** En una transformación de clasificador reutilizable, las vistas General y Puertos se combinan en la ficha **Resumen**.

#### Estrategias

Añadir, quitar o editar una estrategia.

#### Dependencias

Ver los puertos de entrada y salida en cada estrategia.

#### Avanzadas

Establecer el nivel de detalle que la transformación escribe en los archivos de registro.

## Estrategias de clasificador

Una estrategia es un conjunto de operaciones de análisis de datos que una transformación realiza en los datos de entrada. Debe crear como mínimo una estrategia en la transformación de clasificador. Una estrategia de clasificador lee un solo puerto de entrada.

Puede definir una o más operaciones en una estrategia. Una operación de clasificador identifica un modelo de clasificador y un algoritmo de clasificador para aplicar a los datos de los puertos de entrada. Cada operación escribe para un puerto de salida diferente. Puede crear varias operaciones en una estrategia cuando desee analizar un puerto de entrada de diferentes maneras.

**Nota:** Cuando desee identificar los idiomas utilizados en los datos de origen, seleccione el algoritmo Entropía máxima en la operación de clasificador.

## Propiedades avanzadas de Transformación de clasificador

Configure propiedades que ayudan a determinar cómo el servicio de integración de datos procesa los datos para la Transformación de clasificador.

Puede configurar niveles de seguimiento de registros.

Configure la siguiente propiedad en la ficha **Avanzadas**:

#### Nivel de seguimiento

Cantidad de detalles que aparecen en el registro de esta transformación. Puede elegir simplificado, normal, inicialización detallada o datos detallados. El valor predeterminado es Normal.

# Cómo configurar una estrategia de clasificador

Puede configurar una estrategia para identificar los tipos de información en los datos. Cada estrategia analiza un puerto de entrada.

En una transformación no reutilizable, conecte los puertos de entrada a la transformación antes de configurar la estrategia.

1. Abra la transformación y seleccione la vista **Estrategias**.
2. Haga clic en **Nueva estrategia**.  
Se abrirá el asistente para la creación de estrategias.
3. Escriba un nombre y una descripción opcional para la estrategia.
4. Seleccione un puerto de entrada del campo Entradas.
5. Compruebe que el valor de precisión del puerto de entrada es lo suficientemente alto para leer todos los campos del puerto de entrada. El puerto trunca los datos de entrada cuando alcanzan el límite de precisión.
6. Seleccione o borre la opción para añadir datos de puntuación a la salida de estrategia.
7. Haga clic en **Siguiente**.
8. Confirme el tipo de operación de clasificador y haga clic en **Siguiente**.
9. Seleccione un algoritmo de clasificador. Puede seleccionar los siguientes algoritmos:
  - Bayesiano Ingenuo
  - Entropía máxima

**Nota:** Para identificar el idioma utilizado en los datos de origen, seleccione el algoritmo de entropía máxima.
10. Compruebe el puerto de salida.  
La transformación crea un solo puerto de salida para cada operación en una estrategia. Puede editar el nombre de puerto y la precisión.
11. Seleccione un modelo de clasificador.  
El asistente enumera los objetos de modelo de clasificador en el repositorio de modelos.
12. Haga clic en **Siguiente** para añadir otra operación a la estrategia. De lo contrario, haga clic en **Finalizar**.

## Ejemplo de análisis de clasificador

Trabaja como gestor de datos en una empresa de software que ha lanzado una nueva aplicación para smartphones. La empresa quiere entender la respuesta del público a la aplicación y la cobertura que recibe en los medios de comunicación. La compañía le pide a usted y a su equipo que analicen los comentarios de las redes sociales sobre la aplicación.

Decide capturar datos de fuentes de Twitter que analizan smartphones. Utiliza la interfaz de programación de aplicaciones de Twitter para filtrar el flujo de datos de Twitter. Crea un origen de datos que contiene los datos de Twitter que desea analizar.

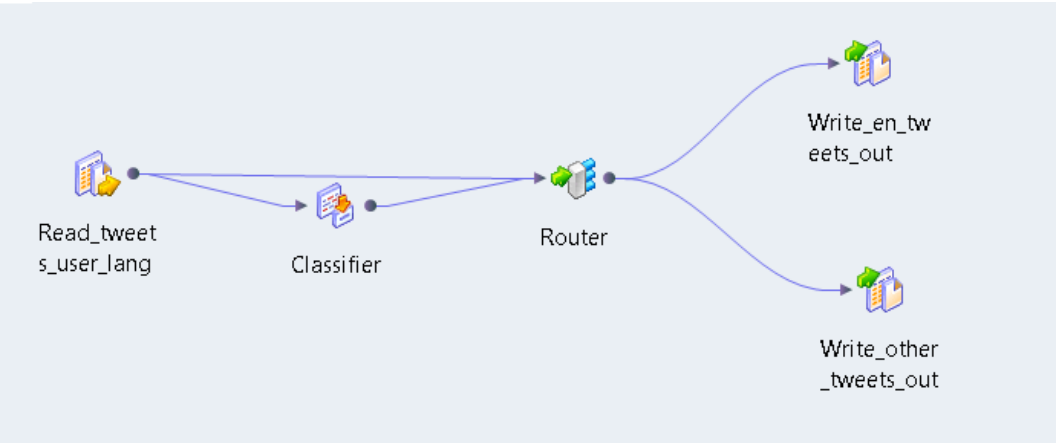
Dado que las fuentes de Twitter contienen mensajes en varios idiomas, debe identificar el idioma utilizado en cada mensaje. Decide usar una transformación de clasificador para analizar los idiomas. Crea una

asignación que identifica los idiomas en los datos de origen y escribe los mensajes de Twitter en destinos de datos en inglés e idiomas distintos al inglés.

## Crear la asignación de clasificador

Puede crear una asignación que lea un origen de datos, clasifique los idiomas en los datos y escriba los datos en destinos según los idiomas que contengan.

La siguiente imagen muestra la asignación en Developer Tool:



La asignación que cree contiene los siguientes objetos:

Nombre de objeto	Descripción
Read_tweet_user_lang	Origen de datos. Contiene los mensajes de Twitter
Clasificador	Transformación de clasificador. Identifica los idiomas utilizados en los mensajes de Twitter.
Enrutador	Transformación de enrutador. Distribuye los mensajes de Twitter a los objetos de destino de datos según los idiomas contienen.
Write_en_tweets_out	Destino de datos. Contiene mensajes de Twitter en inglés.
Write_other_tweets_out	Destino de datos. Contiene mensajes de Twitter en idiomas distintos al inglés.

## Muestra de datos de entrada

El siguiente fragmento de datos muestra un ejemplo de los datos de Twitter que analiza en la asignación:

### Twitter Message

```
RT @GanaphoneS3: Faltan 10 minutos para la gran rifa de un iPhone 5...
RT @Clarified: How to Downgrade Your iPhone 4 From iOS 6.x to iOS 5.x (Mac)...
RT @jerseyjazz: The razor was the iPhone of the early 2000s
RT @KrissiDevine: Apple Pie that I made for Thanksgiving. http://t.com/s9ImzFxO
RT @sophieHz: Dan yang punya 2 kupon undian. Masuk dalam kotak undian yang
berhadiah Samsung
RT @IsabelFreitas: o galaxy tem isso isso isso e a bateria ã melhor que do iPhone
RT @PremiusIpad: Faltan 15 minutos para la gran rifa de un iPhone 5...
RT @payyton3: I want apple cider
RT @wiesteronder: Retweet als je iets van Apple, Nike, Adidas of microsoft hebt!
```

## Configuración de origen de datos

El origen de datos contiene un solo puerto. Cada fila en el puerto contiene un único mensaje de Twitter.

La siguiente tabla describe la configuración del origen de datos:

Nombre de puerto	Tipo de puerto	Precisión
text	n/a	200

## Configuración de transformación de clasificador

La transformación de clasificador utiliza un solo puerto de entrada y salida. El puerto de entrada de la transformación lee el campo de texto del origen de datos. El puerto de salida contiene el idioma identificado para cada mensaje de Twitter en el campo de texto. La transformación de clasificador utiliza códigos de países ISO para identificar el idioma.

La siguiente tabla describe la configuración de la transformación de clasificador:

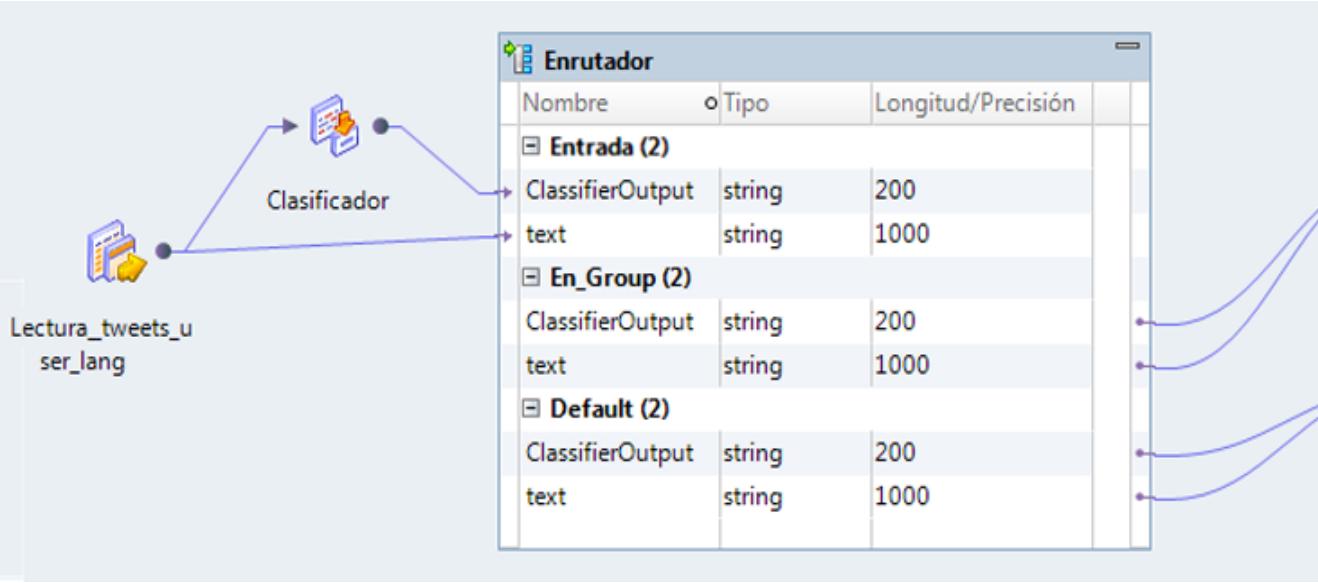
Nombre de puerto	Tipo de puerto	Precisión	Estrategia
text_input	Entrada	200	Classifier1
Classifier_Output	Salida	2	Classifier1

## Configuración de transformación de enrutador

La transformación de enrutador usa dos puertos de entrada. Lee los mensajes de Twitter desde el origen de datos y los códigos ISO de países desde el clasificador de transformación. La transformación de enrutador

envía los datos en los puertos de entrada a distintos puertos de salida en función de una condición que especifique.

La siguiente imagen muestra los grupos de puertos y las conexiones de puertos de la transformación de enrutador:



La siguiente tabla describe la configuración de la transformación de enrutador:

Nombre de puerto	Tipo de puerto	Grupo de puertos	Precisión
Classifier_Output	Entrada	Entrada	2
text	Entrada	Entrada	200
Classifier_Output	Entrada	Predeterminado	2
text	Entrada	Predeterminado	200
Classifier_Output	Entrada	En_Group	2
text	Entrada	En_Group	200

Configure la transformación para crear flujos de datos para mensajes en inglés y para mensajes en otros idiomas. Para crear un flujo de datos, añada un grupo de puertos de salida a la transformación. Utilice las opciones de **Grupos** en la transformación para añadir el grupo de puertos.

Para determinar cómo la transformación enruta los datos a cada flujo de datos, debe definir una condición en un grupo de puertos. La condición identifica un puerto y especifica un valor posible en el puerto. Cuando la transformación encuentra un valor de puerto de entrada que coincide con la condición, enruta los datos de entrada al grupo de puertos que cumple la condición.

Defina la siguiente condición en En\_Group:

```
ClassifierOutput='en'
```

**Nota:** La transformación de enrutador lee datos de dos objetos en la asignación. La transformación puede combinar los datos en cada grupo de salida porque no modifica la secuencia de filas definida en los objetos de datos.

## Configuración de destino de datos

La asignación contiene un destino de datos para los mensajes de Twitter en inglés y un destino para los mensajes en otros idiomas. Debe conectar los puertos de un grupo de salida de transformación de enrutador a un destino de datos.

La siguiente tabla describe la configuración de los destinos de datos:

Nombre de puerto	Tipo de puerto	Precisión
text	n/a	200
Classifier_Output	n/a	2

## Resultados de la asignación de clasificador

Cuando se ejecuta la asignación, la transformación de clasificador identifica el idioma de cada mensaje de Twitter. La transformación de enrutador escribe el texto del mensaje para los destinos de datos en función de las clasificaciones de idioma.

El siguiente fragmento de datos muestra un ejemplo de datos de destino del inglés:

ISO Country Code	Twitter Message
en	RT @Clarified: How to Downgrade Your iPhone 4 From iOS 6.x to iOS 5.x (Mac)...
en	RT @jerseyjazz: The razor was the iPhone of the early 2000s
en	RT @KrissiDevine: Apple Pie that I made for Thanksgiving. <a href="http://t.com/s9ImzFxO">http://t.com/s9ImzFxO</a>
en	RT @payyton3: I want apple cider

El siguiente fragmento de datos muestra un ejemplo de los datos de destino identificados para otros idiomas:

ISO Country Code	Twitter Message
es	RT @Ganaphones3: Faltan 10 minutos para la gran rifa de un iPhone 5...
id	RT @sophieHz: Dan yang punya 2 kupon undian. Masuk dalam kotak undian yang berhadiah Samsung Champ.
pt	RT @IsabelFreitas: o galaxy tem isso isso isso e a bateria ã melhor que do iPhone

ISO Country Code	Twitter Message
es	RT @PremiusIpad: Faltan 15 minutos para la gran rifa de un iPhone 5...
nl	RT @wiesteronder: Retweet als je iets van Apple, Nike, Adidas of microsoft hebt! <a href="http://t.co/Je6Ts00H">http://t.co/Je6Ts00H</a>

## CAPÍTULO 8

# Transformación de comparación

Este capítulo incluye los siguientes temas:

- [Resumen de la transformación de comparación, 93](#)
- [Estrategias para la coincidencia de campos, 94](#)
- [Estrategias de coincidencia de identidades, 96](#)
- [Cómo configurar una estrategia de comparación, 97](#)
- [Propiedades avanzadas de Transformación de comparación, 97](#)

## Resumen de la transformación de comparación

La transformación de comparación es una transformación pasiva que evalúa las similitudes entre pares de cadenas de entrada y calcula el grado de similitud para cada par en forma de puntuación numérica.

Cuando configure la transformación, seleccione un par de columnas y asígneles una estrategia de coincidencia.

La transformación de comparación produce puntuaciones de coincidencia en un intervalo de 0 a 1, donde 1 indica una coincidencia perfecta.

**Nota:** Las estrategias disponibles en la transformación de comparación también están disponibles en la transformación de coincidencia. Utilice la transformación de comparación para definir operaciones de comparación de coincidencia que se añadirán a un mapplet de coincidencia. Puede añadir varias estrategias de comparación al mapplet. Utilice la transformación de coincidencia para definir comparaciones de coincidencia en una transformación individual. Puede incrustar un mapplet de coincidencia en una transformación de coincidencia.

# Estrategias para la coincidencia de campos

La transformación de comparación incluye estrategias para la coincidencia de campos predefinidas que comparan pares de campos de datos de entrada.

## Bigram

Utilice el algoritmo Bigram para comparar cadenas de texto largas, como las direcciones postales introducidas en un único campo.

El algoritmo Bigram calcula una puntuación de coincidencia para dos cadenas de datos según la ocurrencia de caracteres consecutivos en ambas cadenas. El algoritmo busca pares de caracteres consecutivos comunes a ambas cadenas y divide el número de pares de caracteres coincidentes por el número total de pares de caracteres.

### Ejemplo de Bigram

Considere las siguientes cadenas:

- larder
- lerder

Estas cadenas producen los siguientes grupos Bigram:

```
l a, a r, r d, d e, e r  
l e, e r, r d, d e, e r
```

Observe que la segunda ocurrencia de la cadena "e r" dentro de la cadena "lerder" no tiene coincidencia, puesto que no hay una segunda ocurrencia correspondiente de "e r" en la cadena "larder".

Para calcular la puntuación de coincidencia Bigram, la transformación divide el número de pares coincidentes (6) por el número total de pares en ambas cadenas (10). En este ejemplo, las cadenas son similares en un 60% y la puntuación de coincidencia es de 0,60.

## Distancia de Hamming

Utilice el algoritmo Distancia de Hamming cuando la posición de los caracteres de datos sea un factor crítico; por ejemplo, en campos numéricos o de códigos como números de teléfono, códigos postales o códigos de producto.

El algoritmo de la distancia de Hamming calcula una puntuación de coincidencia para dos cadenas de datos calculando el número de posiciones en las que los caracteres difieren de una cadena de datos a otra. En el caso de que las cadenas tengan una longitud diferente, cada carácter adicional de la cadena más larga se cuenta como una diferencia de una cadena a otra.

### Ejemplo de distancia de Hamming

Considere las siguientes cadenas:

- Morlow
- Marlowes

Los caracteres resaltados indican las posiciones que el algoritmo de Hamming identifica como diferentes.

Para calcular la puntuación de coincidencia Hamming, la transformación divide el número de caracteres coincidentes (5) por la longitud de la cadena más larga (8). En este ejemplo, las cadenas son similares en un 62,5% y la puntuación de coincidencia es de 0,625.

## Distancia de edición

Utilice el algoritmo Distancia de edición para comparar palabras o cadenas de texto cortas, como un nombre.

El algoritmo Distancia de edición calcula el "coste" mínimo de transformar una cadena en otra por medio de la inserción, eliminación o reemplazo de caracteres.

### Ejemplo de Distancia de edición

Considere las siguientes cadenas:

- Levenston
- Levenshtein

Los caracteres resaltados indican las operaciones necesarias para transformar una cadena en otra.

El algoritmo Distancia de edición divide el número de caracteres no modificados (8) entre la longitud de la cadena más larga (11). En este ejemplo, las cadenas son similares en un 72,7% y la puntuación de coincidencia es de 0,727.

## Distancia de Jaro

Utilice el algoritmo Distancia de Jaro para comparar dos cadenas cuando la similitud entre los caracteres iniciales de las cadenas tenga prioridad.

La puntuación de coincidencia de la distancia de Jaro refleja el grado de similitud entre los primeros cuatro caracteres de ambas cadenas y el número de transposiciones de carácter identificadas. La transformación pondera la importancia de la coincidencia entre los primeros cuatro caracteres utilizando el valor que se introduce en la propiedad Penalización.

### Propiedades de la distancia de Jaro

Cuando se configura un algoritmo Distancia de Jaro, puede configurar las siguientes propiedades:

#### Penalización

Determina la penalización para la puntuación de coincidencia si los primeros cuatro caracteres en dos cadenas comparadas no son idénticos. La transformación resta el valor completo de la penalización cuando no hay ninguna coincidencia con el primer carácter. La transformación resta fracciones de la penalización según la posición que ocupan los otros caracteres sin coincidencia. El valor predeterminado de la penalización es 0,20.

#### Distinguir mayúsculas de minúsculas

Determina si el algoritmo Distancia de Jaro tiene en cuenta el formato de mayúsculas y minúsculas de los caracteres cuando realiza operaciones de coincidencia.

### Ejemplo de la distancia de Jaro

Considere las siguientes cadenas:

- 391859
- 813995

Si utiliza el valor de Penalización predeterminado de 0,20 para analizar estas cadenas, el algoritmo Distancia de Jaro devuelve una puntuación de coincidencia de 0,513. Esta puntuación de coincidencia indica que las cadenas tienen una similitud del 51,3%.

## Invertir la distancia de Hamming

Utilice el algoritmo Invertir la distancia de Hamming para calcular el porcentaje de posiciones de carácter que difiere entre dos cadenas, leídas de derecha a izquierda.

El algoritmo de la distancia de Hamming calcula una puntuación de coincidencia para dos cadenas de datos calculando el número de posiciones en las que los caracteres difieren de una cadena de datos a otra. En el caso de que las cadenas tengan una longitud diferente, el algoritmo cuenta cada carácter adicional de la cadena más larga se cuenta como una diferencia de una cadena a otra.

### Ejemplo de Invertir la distancia de Hamming

Considere las siguientes cadenas, que utilizan la alineación de derecha a izquierda para imitar el algoritmo Invertir la distancia de Hamming:

- 1-999-9999
- **011-01**-999-9991

Los caracteres resaltados indican las posiciones que el algoritmo Invertir la distancia de Hamming identifica como diferentes.

Para calcular la puntuación de coincidencia de Invertir la distancia de Hamming, la transformación divide el número de caracteres coincidentes (9) entre la longitud de la cadena más larga (15). En este ejemplo, la puntuación de coincidencia es 0,6, lo que indica que las cadenas tienen una similitud del 60%.

## Estrategias de coincidencia de identidades

La transformación de comparación incluye estrategias de coincidencia de identidades que puede usar para encontrar coincidencias en individuos, direcciones o entidades empresariales.

En la siguiente tabla se describe la operación de coincidencia que realiza cada estrategia de coincidencia de identidades:

Estrategias de coincidencia de identidades	Operación de coincidencia
Dirección	Identifica una coincidencia de dirección.
Contacto	Identifica un contacto de una organización en una ubicación individual.
Entidad corporativa	Identifica una organización por su nombre corporativo legal.
División	Identifica una organización en una dirección.
Familia	Identifica una familia por nombre de familia y dirección o número de teléfono.
Campos	Identifica los campos personalizados seleccionados.
Doméstico	Identifica los miembros de una familia que residen en la misma vivienda.
Individuo	Identifica un individuo por su nombre y por su ID o fecha de nacimiento.
Organización	Identifica una organización por su nombre.

Estrategias de coincidencia de identidades	Operación de coincidencia
Nombre de persona	Identifica una persona por su nombre.
Residente	Identifica una persona en una dirección.
Contacto en general	Identifica un contacto de una organización independientemente de cual sea su ubicación.
Doméstico en general	Identifica los miembros de una familia independientemente de su ubicación.

**Nota:** Las estrategias de coincidencia de identidades leen archivos de datos de referencia llamados **poblaciones**. Póngase en contacto con el usuario de Informatica Administrator para obtener información sobre los archivos de poblaciones instalados en el sistema.

## Cómo configurar una estrategia de comparación

Para configurar una estrategia de comparación, edite los ajustes en la vista **Estrategias** de una transformación de comparación.

1. Seleccione la vista **Estrategias**.
2. Seleccione una estrategia de comparación de la sección **Estrategias**.
3. En la sección **Campos**, haga doble clic en una celda de la columna **Campos disponibles** para seleccionar una entrada.

**Nota:** Debe seleccionar una entrada para cada fila que muestra el nombre de entrada en negrita en la columna **Campos de entrada**.

## Propiedades avanzadas de Transformación de comparación

Configure propiedades que ayudan a determinar cómo el servicio de integración de datos procesa los datos para la Transformación de comparación.

Puede configurar niveles de seguimiento de registros.

Configure la siguiente propiedad en la ficha **Avanzadas**:

### Nivel de seguimiento

Cantidad de detalles que aparecen en el registro de esta transformación. Puede elegir simplificado, normal, inicialización detallada o datos detallados. El valor predeterminado es Normal.

## CAPÍTULO 9

# Transformación de consolidación

Este capítulo incluye los siguientes temas:

- [Resumen de la transformación de consolidación, 98](#)
- [Asignaciones de consolidación, 99](#)
- [Puertos de transformación de consolidación, 99](#)
- [Vistas de transformación de consolidación, 99](#)
- [Estrategias simples, 102](#)
- [Estrategias basadas en filas, 103](#)
- [Estrategias avanzadas, 103](#)
- [Funciones de consolidación simple, 104](#)
- [Funciones de consolidación basada en filas, 108](#)
- [Ejemplo de asignación de consolidación, 112](#)
- [Cómo configurar una transformación de consolidación, 113](#)

## Resumen de la transformación de consolidación

La transformación de consolidación es una transformación activa que analiza los grupos de registros relacionados y crea un registro consolidado para cada grupo. Utilice la transformación de consolidación para consolidar los grupos de registros generados por transformaciones como las transformaciones de generador de claves, de coincidencia y de asociación.

La transformación de consolidación genera registros consolidados aplicando estrategias a grupos de registros relacionados. La transformación contiene un puerto de salida que indica qué registro es el consolidado. Puede optar por limitar la salida de transformación para incluir solamente los registros consolidados.

Por ejemplo, puede consolidar los grupos de registros duplicados de empleados que la transformación de coincidencia genera. La transformación de consolidación puede crear un registro consolidado a que contenga los datos fusionados de todos los registros del grupo.

Puede configurar la transformación de consolidación para utilizar diferentes tipos de estrategias según los requisitos de consolidación. Utilice estrategias simples para crear un registro consolidado a partir de varios registros. Cuando se utilizan estrategias simples, se debe especificar una estrategia para cada puerto. Utilice una estrategia basada en filas para analizar las filas en el grupo de registros y crear un registro consolidado con los valores de una de las filas. Utilice una estrategia avanzada para crear un registro consolidado aplicando una expresión que cree.

## Asignaciones de consolidación

Para consolidar registros, cree una asignación que cree grupos de registros relacionados. Añada una transformación de consolidación y configúrela para consolidar cada grupo de registros en un único registro principal.

Cuando cree una asignación de consolidación, seleccione transformaciones de acuerdo con el tipo y la complejidad de los requisitos. Por ejemplo, si utiliza una transformación de generador de claves para agrupar registros, puede conectar una transformación de consolidación directamente a la transformación de generador de claves. La transformación de consolidación crea un registro consolidado a para cada grupo creado por la transformación de generador de claves.

Para consolidar los registros coincidentes, se puede conectar la transformación de consolidación a una transformación de coincidencia. Si necesita consolidar los registros como parte de la administración de registros de excepción, conecte la transformación de consolidación a una transformación de excepción.

## Puertos de transformación de consolidación

Developer Tool crea un puerto de salida para cada puerto de entrada que se añade. No se pueden añadir puertos de salida a la transformación manualmente. La transformación de consolidación también incluye un puerto de salida **IsSurvivor** que indica los registros consolidados.

Uno de los puertos de entrada que añada a la transformación de consolidación debe contener claves de grupo. La transformación de consolidación requiere información de la clave de grupo porque las estrategias de consolidación procesan grupos de registros en lugar de conjuntos de datos completos.

Cuando se añade un puerto de entrada, Developer Tool crea un nombre de puerto de salida añadiendo el sufijo "1" al nombre del puerto de entrada. La transformación también incluye el puerto de salida **IsSurvivor**, que indica si un registro es el consolidado. En el caso de los registros consolidados, la transformación de consolidación escribe la cadena "Y" en el puerto **IsSurvivor**. En el caso de los registros de entrada, la transformación de consolidación escribe la cadena "N" en el puerto **IsSurvivor**.

## Vistas de transformación de consolidación

La transformación de consolidación contiene vistas para los puertos, las estrategias y las propiedades avanzadas.

### Vista Estrategias de la transformación de consolidación

La vista **Estrategias** contiene propiedades para las estrategias simples, basadas en filas y avanzadas.

La siguiente lista describe los tipos de estrategias de consolidación:

#### Estrategia simple

Una estrategia simple analiza todos los valores de un puerto de un grupo de registros y selecciona uno. Se debe especificar una estrategia simple para cada puerto. La transformación de consolidación utiliza los valores de puerto seleccionados por todas las estrategias simples para crear un registro

consolidado. Algunos ejemplos de estrategias simples son el valor más frecuente de un puerto, el valor más largo de un puerto o el valor más frecuente que no está en blanco de un puerto.

#### **Estrategia basada en filas**

Una estrategia basada en filas analiza las filas del grupo de registros y selecciona una. La transformación de consolidación utiliza los valores de puerto de la fila para crear un registro consolidado. Algunos ejemplos de estrategias basadas en filas son el mayor número de caracteres, el menor número de campos en blanco o el mayor número de campos más frecuentes.

#### **Estrategia avanzada**

Una estrategia de avanzada analiza un grupo de registros mediante estrategias definidas por el usuario. Las estrategias avanzadas se construyen utilizando funciones de consolidación en una expresión. La transformación de consolidación crea un registro consolidado con base en la salida de la expresión. La expresión creada por el usuario también puede usar todas las funciones disponibles en la transformación de decisión.

## **Propiedades avanzadas de Transformación de consolidación**

La transformación de consolidación contiene propiedades avanzadas para la ordenación, la salida, la configuración de la memoria caché y el nivel de seguimiento.

La siguiente lista describe las propiedades de consolidación avanzadas:

#### **Ordenar**

Ordena las filas de entrada según el grupo por el puerto que configure en la vista **Estrategias**.

Seleccione esta opción si las filas de entrada no están ordenadas. Esta propiedad está habilitada de manera predeterminada.

#### **Ordenar con distinción de mayúsculas y minúsculas**

Utiliza la ordenación con distinción de mayúsculas y minúsculas al ordenar las filas de entrada. Esta propiedad está habilitada de manera predeterminada.

#### **Modo de salida**

Determina si la salida de transformación incluye todos los registros o solamente los consolidados. La opción predeterminada es Todos.

#### **Directorio de archivo caché**

El servicio de integración de datos utiliza el directorio del archivo de memoria caché para crear archivos temporales cuando ordena datos. Después de que el servicio de integración de datos ordena los datos, elimina los archivos temporales. Puede especificar cualquier directorio del equipo donde se ejecuta el servicio de integración de datos para usarlo como directorio del archivo de memoria caché. El valor predeterminado es el parámetro del sistema CacheDir.

#### **Tamaño de archivo caché**

Determina la cantidad máxima de memoria necesaria cuando la transformación de consolidación realiza operaciones de ordenación. El servicio de integración de datos pasa todos los datos entrantes a la transformación de consolidación antes de realizar una operación de ordenación. El valor predeterminado es 400.000 bytes.

#### **Nivel de seguimiento**

Cantidad de detalles que aparecen en el registro de esta transformación. Puede elegir simplificado, normal, inicialización detallada o datos detallados. El valor predeterminado es Normal.

## Tamaño de archivo caché

La propiedad de tamaño del archivo de memoria caché determina la cantidad máxima de memoria necesaria cuando la transformación de consolidación realiza operaciones de clasificación. El servicio de integración de datos pasa todos los datos entrantes a la transformación de consolidación antes de realizar una operación de ordenación.

Antes de iniciar la operación de ordenación, el servicio de integración de datos asigna la cantidad de memoria configurada para el tamaño del archivo de memoria caché. Si el servicio de integración de datos no puede asignar suficiente memoria, la asignación genera un error.

Para conseguir un rendimiento óptimo, configure el tamaño del archivo de memoria caché con un valor menor o igual que la cantidad de memoria física RAM del equipo donde reside el servicio de integración de datos. Asigne como mínimo 16 MB (16.777.216 bytes) de memoria física para ordenar datos con una transformación de consolidación.

Si la cantidad de datos entrantes es mayor que el tamaño del archivo de memoria caché, el servicio de integración de datos almacena temporalmente los datos en el directorio del archivo de memoria caché. El servicio de integración de datos necesita un espacio de disco de como mínimo el doble de la cantidad de datos entrantes cuando almacena datos en el directorio del archivo de memoria caché. Si la cantidad de datos entrantes es considerablemente mayor que el tamaño del archivo de memoria de caché, el servicio de integración de datos puede requerir más del doble del espacio de disco disponible para el directorio del archivo de memoria caché.

Utilice la siguiente fórmula para determinar el tamaño de los datos entrantes:

$$[\text{number\_of\_input\_rows} \times (\text{Sum}(\text{column\_size}) + 16)]$$

La siguiente tabla describe los valores de tamaño de columna, por tipo de datos, para los cálculos de datos de archivo de memoria caché:

Tipo de datos	Tamaño de columna
Binary	precisión + 8 Redondeo al múltiplo de 8 más cercano
Date/Time	29
Decimal, alta precisión desactivada (todas las precisiones)	16
Decimal, alta precisión activada (precisión <=18)	24
Decimal, alta precisión activada (precisión >18, <=28)	32
Decimal, alta precisión activada (precisión >28)	16
Decimal, alta precisión activada (escala negativa)	16
Double	16
Real	16
Integer	16
String, Text	Modo Unicode: 2*(precisión + 5) Modo ASCII: precisión + 9

# Estrategias simples

Una estrategia simple analiza un puerto de un grupo de registros y devuelve un valor. Se debe especificar una estrategia simple para cada puerto. La transformación de consolidación utiliza los valores de puerto seleccionados por todas las estrategias simples para crear un registro consolidado.

Cuando configura una estrategia en la vista **Estrategias** de la transformación, la estrategia muestra el siguiente texto como el método de consolidación:

*Utilizar predeterminado.*

La estrategia predeterminada es "ID de fila máximo".

Puede elegir de entre las siguientes estrategias simples:

## **Media**

Analiza un puerto del grupo de registros y devuelve el promedio de todos los valores.

En el caso de los tipos de datos String y Date/time, la estrategia devuelve el valor que se repite con más frecuencia.

## **El más largo**

Analiza un puerto del grupo de registros y devuelve el valor con el mayor número de caracteres. Si hay dos o más valores con el mayor número de caracteres, la estrategia devuelve el primer valor apto.

## **Máximo**

Analiza un puerto del grupo de registros y devuelve el valor más alto.

Para el tipo de datos String, la estrategia devuelve la cadena más larga. Para el tipo de datos Date/time, la estrategia devuelve la fecha más reciente.

## **Mínimo**

Analiza un puerto del grupo de registros y devuelve el valor más bajo.

Para el tipo de datos String, la estrategia devuelve la cadena más corta. Para el tipo de datos Date/time, la estrategia devuelve la fecha más temprana.

## **Más frecuente**

Analiza un puerto del grupo de registros y devuelve el valor más frecuente, incluidos los valores en blanco o nulos. Si dos o más valores son los más frecuentes, la estrategia devuelve el primer valor apto.

## **Más frecuente que no está en blanco**

Analiza un puerto del grupo de registros y devuelve el valor más frecuente, excluyendo los valores en blanco o nulos. Si hay dos valores con el número más alto de elementos que no están en blanco, la estrategia devuelve el primer valor apto.

## **Más corto**

Analiza un puerto del grupo de registros y devuelve el valor con el menor número de caracteres. Si hay dos o más valores que tienen el menor número de caracteres, la estrategia devuelve el primer valor apto.

## **Id de fila máximo**

Analiza un puerto del grupo de registros y devuelve el valor con el ID de fila más alto.

## Estrategias basadas en filas

Una estrategia basada en filas analiza las filas del grupo de registros y selecciona una. La transformación de consolidación utiliza los valores de puerto de la fila para crear un registro consolidado. La estrategia predeterminada es "mayor parte de los datos".

Elija una de las siguientes estrategias basadas en filas:

### Mayor parte de los datos

Selecciona la fila con el mayor número de caracteres. Si hay dos o más filas con el mayor número de caracteres, la estrategia devuelve el último valor apto.

### Más completos

Selecciona la fila con el número más alto de columnas que no están en blanco. Si hay dos o más filas con el mayor número de columnas que no están en blanco, la estrategia devuelve el último valor apto.

### Exacto modal

Selecciona la fila con el mayor número de valores más frecuentes que no están en blanco. Por ejemplo, imaginemos una fila que tiene tres puertos que contienen a su vez los valores más frecuentes del grupo de registros. El número de valores más frecuentes de esa fila es "3".

Si hay dos o más filas con el mayor número de valores más frecuentes que no están en blanco, la estrategia devuelve el último valor apto.

### Ejemplo de estrategia basada en filas

La siguiente tabla muestra un grupo de registros de ejemplo. La última columna describe los motivos por los que cada estrategia basada en filas selecciona filas diferentes en este grupo de registros.

ID de producto	Nombre	Apellido	Código postal	Selección de estrategia
2106	Bartholomew		28516	La estrategia Mayor parte de los datos selecciona esta fila porque contiene más caracteres que el resto de filas.
2236	Bart	Smith	28579	La estrategia Más completos selecciona esta fila porque tiene más columnas que no están en blanco que el resto de filas.
2236	<En blanco>	Smith	28516	La estrategia Exacto modal selecciona esta fila porque contiene el mayor número de valores más frecuentes.

## Estrategias avanzadas

Puede utilizar las estrategias avanzadas para crear estrategias de consolidación a partir de funciones predefinidas. Puede utilizar funciones de consolidación y otras funciones de Informática.

Puede crear expresiones que contengan funciones de consolidación simple o basada en filas. Las funciones de consolidación simple se utilizan para construir un registro consolidado según los valores de puerto del grupo de registros. Las funciones de consolidación basada en filas se utilizan para seleccionar una fila del grupo de registros.

Las expresiones de consolidación deben llenar todos los puertos de salida en la transformación de consolidación. Si las expresiones de consolidación no utilizan todos los puertos de salida, la transformación hace que las asignaciones fallen.

Puede utilizar una estrategia simple o basada en filas como plantilla para una estrategia de avanzada. Configure una estrategia simple o basada en filas y, a continuación, seleccione Avanzada. La transformación de consolidación genera una expresión con las funciones que realizan la estrategia. Puede añadir más funciones para implementar requisitos adicionales.

## Funciones de consolidación simple

Las funciones de consolidación simple seleccionan un valor de todos los valores de puerto de un grupo de registros. Cuando se utiliza una función de consolidación simple, se debe proporcionar la función con un puerto y el puerto agrupar por.

### CONSOL\_AVG

Analiza un puerto del grupo de registros y devuelve el promedio de todos los valores.

#### Sintaxis

```
CONSOL_AVG(string, group by)
```

En la siguiente tabla se describen los argumentos de este comando:

Argumento	Obligatorio/opcional	Descripción
<i>string</i>	Obligatorio	Nombre del puerto de entrada.
<i>agrupar por</i>	Obligatorio	Nombre del puerto de entrada que contiene el identificador de grupo.

#### Valor de retorno

El promedio de todos los valores de un puerto.

Para los tipos de datos String y Date/time, la función devuelve el valor que más se repite.

#### Ejemplo

La siguiente expresión usa la función `CONSOL_AVG` para buscar el valor promedio del puerto de entrada SalesTotal:

```
SalesTotal1:= CONSOL_AVG(SalesTotal, GroupKey)
```

En esta expresión, la función `CONSOL_AVG` utiliza el puerto `GroupKey` para identificar un grupo de registros. Dentro de ese grupo de registros, la función analiza el puerto `SalesTotal` y devuelve el valor promedio. La expresión escribe el valor promedio del puerto de salida `SalesTotal1`.

### CONSOL\_LONGEST

Analiza un puerto del grupo de registros y devuelve el valor con el mayor número de caracteres.

#### Sintaxis

```
CONSOL_LONGEST(string, group by)
```

En la siguiente tabla se describen los argumentos de este comando:

Argumento	Obligatorio/opcional	Descripción
<i>string</i>	Obligatorio	Nombre del puerto de entrada.
<i>agrupar por</i>	Obligatorio	Nombre del puerto de entrada que contiene el identificador de grupo.

### Valor de retorno

El valor de puerto con el mayor número de caracteres.

Si hay dos o más valores con el mayor número de caracteres, la estrategia devuelve el primer valor apto.

### Ejemplo

La siguiente expresión usa la función `CONSOL_LONGEST` para analizar el puerto de entrada `FirstName` y buscar el valor con el mayor número de caracteres:

```
FirstName1:= CONSOL_LONGEST(FirstName, GroupKey)
```

En esta expresión, la función `CONSOL_LONGEST` utiliza el puerto `GroupKey` para identificar un grupo de registros. Dentro de ese grupo de registros, la función analiza el puerto `FirstName` y devuelve el valor más largo. La expresión escribe este valor en el puerto de salida `FirstName1`.

## CONSOL\_MAX

Analiza un puerto del grupo de registros y devuelve el valor más alto.

### Sintaxis

```
CONSOL_MAX(string, group by)
```

En la siguiente tabla se describen los argumentos de este comando:

Argumento	Obligatorio/opcional	Descripción
<i>string</i>	Obligatorio	Nombre del puerto de entrada.
<i>agrupar por</i>	Obligatorio	Nombre del puerto de entrada que contiene el identificador de grupo.

### Valor de retorno

El valor de puerto más alto.

Para el tipo de datos `String`, la función devuelve la cadena más larga. Para el tipo de datos `Date/time`, la función devuelve la fecha más reciente.

### Ejemplo

La siguiente expresión usa la función `CONSOL_MAX` para analizar el puerto de entrada `SalesTotal` y buscar el valor máximo:

```
SalesTotal1:= CONSOL_MAX(SalesTotal, GroupKey)
```

En esta expresión, la función `CONSOL_MAX` utiliza el puerto `GroupKey` para identificar un grupo de registros. Dentro del grupo de registros, la función analiza el puerto `SalesTotal` y devuelve el valor más alto. La expresión escribe este valor en el puerto de salida `SalesTotal1`.

## CONSOL\_MIN

Analiza un puerto del grupo de registros y devuelve el valor más bajo.

### Sintaxis

```
CONSOL_MIN(string, group by)
```

En la siguiente tabla se describen los argumentos de este comando:

Argumento	Obligatorio/opcional	Descripción
<i>string</i>	Obligatorio	Nombre del puerto de entrada.
<i>agrupar por</i>	Obligatorio	Nombre del puerto de entrada que contiene el identificador de grupo.

### Valor de retorno

El valor de puerto más bajo.

Para el tipo de datos String, la función devuelve la cadena más corta. Para el tipo de datos Date/time, la función devuelve la fecha más temprana.

### Ejemplo

La siguiente expresión usa la función `CONSOL_MIN` para analizar el puerto de entrada `SalesTotal` y buscar el valor mínimo:

```
SalesTotal1:= CONSOL_MIN(SalesTotal, GroupKey)
```

En esta expresión, la función `CONSOL_MIN` utiliza el puerto `GroupKey` para identificar un grupo de registros. Dentro del grupo de registros, la función analiza el puerto `SalesTotal` y devuelve el valor más bajo. La expresión escribe este valor en el puerto de salida `SalesTotal1`.

## CONSOL\_MOSTFREQ

Analiza un puerto del grupo de registros y devuelve el valor más frecuente, incluidos los valores en blanco o nulos.

### Sintaxis

```
CONSOL_MOSTFREQ(string, group by)
```

En la siguiente tabla se describen los argumentos de este comando:

Argumento	Obligatorio/opcional	Descripción
<i>string</i>	Obligatorio	Nombre del puerto de entrada.
<i>agrupar por</i>	Obligatorio	Nombre del puerto de entrada que contiene el identificador de grupo.

### Valor de retorno

El valor más frecuente, incluidos los valores en blanco o nulos.

Si dos o más valores son los más frecuentes, la estrategia devuelve el primer valor apto.

## Ejemplo

La siguiente expresión usa la función `CONSOL_MOSTFREQ` para analizar el puerto de entrada `Company` y buscar el valor más frecuente:

```
Company1:= CONSOL_MOSTFREQ(Company, GroupKey)
```

En esta expresión, la función `CONSOL_MOSTFREQ` utiliza el puerto `GroupKey` para identificar un grupo de registros. Dentro del grupo de registros, la función analiza el puerto `Company` y devuelve el valor más frecuente. La expresión escribe este valor en el puerto de salida `Company1`.

## CONSOL\_MOSTFREQ\_NB

Analiza un puerto del grupo de registros y devuelve el valor más frecuente, excluyendo los valores en blanco o nulos.

### Sintaxis

```
CONSOL_MOSTFREQ_NB(string, group by)
```

En la siguiente tabla se describen los argumentos de este comando:

Argumento	Obligatorio/opcional	Descripción
<i>string</i>	Obligatorio	Nombre del puerto de entrada.
<i>agrupar por</i>	Obligatorio	Nombre del puerto de entrada que contiene el identificador de grupo.

### Valor de retorno

El valor más frecuente, excluyendo los valores en blanco o nulos.

Si dos o más valores son los más frecuentes, la estrategia devuelve el primer valor apto.

## Ejemplo

La siguiente expresión usa la función `CONSOL_MOSTFREQ_NB` para analizar el puerto de entrada `Company` y buscar el valor más frecuente:

```
Company1:= CONSOL_MOSTFREQ_NB(Company, GroupKey)
```

En esta expresión, la función `CONSOL_MOSTFREQ_NB` utiliza el puerto `GroupKey` para identificar un grupo de registros. Dentro del grupo de registros, la función analiza el puerto `Company` y devuelve el valor más frecuente. La expresión escribe este valor en el puerto de salida `Company1`.

## CONSOL\_SHORTEST

Analiza un puerto del grupo de registros y devuelve el valor con el menor número de caracteres.

### Sintaxis

```
CONSOL_SHORTEST(string, group by)
```

En la siguiente tabla se describen los argumentos de este comando:

Argumento	Obligatorio/opcional	Descripción
<i>string</i>	Obligatorio	Nombre del puerto de entrada.
<i>agrupar por</i>	Obligatorio	Nombre del puerto de entrada que contiene el identificador de grupo.

### Valor de retorno

El valor de puerto con el menor número de caracteres.

Si hay dos o más valores que tienen el menor número de caracteres, la estrategia devuelve el primer valor apto.

### Ejemplo

La siguiente expresión usa la función `CONSOL_SHORTEST` para analizar el puerto de entrada `FirstName` y buscar el valor con el menor número de caracteres:

```
FirstName1:= CONSOL_SHORTEST(FirstName, GroupKey)
```

En esta expresión, la función `CONSOL_SHORTEST` utiliza el puerto `GroupKey` para identificar un grupo de registros. Dentro del grupo de registros, la función analiza el puerto `FirstName` y devuelve el valor más corto. La expresión escribe este valor en el puerto de salida `FirstName1`.

## Funciones de consolidación basada en filas

Utilice las funciones de consolidación basada en filas para seleccionar un registro en un grupo de registros. Debe utilizar funciones de consolidación basada en filas en las declaraciones `IF-THEN-ELSE`.

### CONSOL\_GETROWFIELD

Lee la fila identificada por una función de consolidación basada en filas y devuelve el valor del puerto que especifique. Para especificar un puerto se utiliza un argumento numérico.

Debe utilizar la función `CONSOL_GETROWFIELD` junto con una de las siguientes funciones de consolidación basada en filas:

- `CONSOL_MODEXACT`
- `CONSOL_MOSTDATA`
- `CONSOL_MOSTFILLED`

Para cada puerto de entrada de una función de consolidación basada en filas debe utilizar una instancia de la función `CONSOL_GETROWFIELD`.

### Sintaxis

```
CONSOL_GETROWFIELD(value)
```

En la siguiente tabla se describen los argumentos de este comando:

Argumento	Obligatorio/opcional	Descripción
<i>value</i>	Obligatorio	El número que indica un puerto de entrada en la función de consolidación basada en filas. Utilice "0" para especificar el puerto situado en el extremo izquierdo de la función. Utilice los números posteriores para indicar otros puertos.

## Valor de retorno

El valor del puerto que especifique. La función lee este valor desde una fila identificada por una función de consolidación basada en filas.

## Ejemplo

La siguiente expresión usa la función `CONSOL_GETROWFIELD` junto con la función `CONSOL_MOSTDATA`:

```
IF (CONSOL_MOSTDATA(First_Name,Last_Name,GroupKey,GroupKey))
THEN
First_Name1 := CONSOL_GETROWFIELD(0)
Last_Name1 := CONSOL_GETROWFIELD(1)
GroupKey1 := CONSOL_GETROWFIELD(2)
ELSE
First_Name1 := First_Name
Last_Name1 := Last_Name
GroupKey1 := GroupKey
ENDIF
```

En esta expresión, la función `CONSOL_MOSTDATA` analiza las filas de un grupo de registros e identifica una única fila. Las funciones `CONSOL_GETROWFIELD` utilizan números consecutivos para leer los valores de puerto de esa fila y escribir los valores en los puertos de salida.

# CONSOL\_MODELEXACT

Identifica la fila con el mayor número de valores más frecuentes.

Por ejemplo, imaginemos una fila que tiene tres puertos que contienen a su vez los valores más frecuentes del grupo de registros. El número de valores más frecuentes de esa fila es "3".

Debe usar esta función junto con la función `CONSOL_GETROWFIELD`. `CONSOL_GETROWFIELD` devuelve los valores de la fila que identifica la función `CONSOL_MODELEXACT`.

## Sintaxis

```
CONSOL_MODELEXACT(string1, [string2, ..., stringN,]
group by)
```

En la siguiente tabla se describen los argumentos de este comando:

Argumento	Obligatorio/opcional	Descripción
<i>string</i>	Obligatorio	Nombre del puerto de entrada.
<i>agrupar por</i>	Obligatorio	Nombre del puerto de entrada que contiene el identificador de grupo.

## Valor de retorno

TRUE para la fila que presenta el mayor número de campos más frecuentes, FALSE para todas las demás filas.

## Ejemplo

La siguiente expresión usa la función `CONSOL_MODELEXACT` para buscar la fila que contiene el mayor número de campos más frecuentes:

```
IF (CONSOL_MODELEXACT(First_Name, Last_Name, GroupKey, GroupKey))
THEN
First_Name1 := CONSOL_GETROWFIELD(0)
Last_Name1 := CONSOL_GETROWFIELD(1)
GroupKey1 := CONSOL_GETROWFIELD(2)
ELSE
First_Name1 := First_Name
Last_Name1 := Last_Name
GroupKey1 := GroupKey
ENDIF
```

En esta expresión, la función `CONSOL_MODELEXACT` analiza las filas de un grupo de registros e identifica una única fila. Las funciones `CONSOL_GETROWFIELD` utilizan números consecutivos para leer los valores de puerto de esa fila y escribir los valores en los puertos de salida.

# CONSOL\_MOSTDATA

Identifica la fila que contiene el mayor número de caracteres de todos los puertos

Debe usar esta función junto con la función `CONSOL_GETROWFIELD`. La función `CONSOL_GETROWFIELD` devuelve los valores de la fila que identifica la función `CONSOL_MOSTDATA`.

## Sintaxis

```
CONSOL_MOSTDATA(string1, [string2, ..., stringN,]
group by)
```

En la siguiente tabla se describen los argumentos de este comando:

Argumento	Obligatorio/opcional	Descripción
<i>string</i>	Obligatorio	Nombre del puerto de entrada.
<i>agrupar por</i>	Obligatorio	Nombre del puerto de entrada que contiene el identificador de grupo.

## Valor de retorno

TRUE para la fila que contiene el mayor número de caracteres de todos los puertos, FALSE para todas las demás filas.

## Ejemplo

La siguiente expresión usa la función `CONSOL_MOSTDATA` para encontrar la fila que contiene el mayor número de caracteres:

```
IF (CONSOL_MOSTDATA(First_Name, Last_Name, GroupKey, GroupKey))
THEN
First_Name1 := CONSOL_GETROWFIELD(0)
Last_Name1 := CONSOL_GETROWFIELD(1)
GroupKey1 := CONSOL_GETROWFIELD(2)
ELSE
First_Name1 := First_Name
```

```

Last_Name1 := Last_Name
GroupKey1 := GroupKey
ENDIF

```

En esta expresión, la función `CONSOL_MOSTDATA` analiza las filas de un grupo de registros e identifica una única fila. Las funciones `CONSOL_GETROWFIELD` utilizan números consecutivos para leer los valores de puerto de esa fila y escribir los valores en los puertos de salida.

## CONSOL\_MOSTFILLED

Identifica la fila que contiene el número más alto de campos que no están en blanco.

Debe usar esta función junto con la función `CONSOL_GETROWFIELD`. `CONSOL_GETROWFIELD` devuelve los valores de la fila que identifica la función `CONSOL_MOSTFILLED`.

### Sintaxis

```

CONSOL_MOSTFILLED(string1, [string2, ..., stringN,]
group by)

```

En la siguiente tabla se describen los argumentos de este comando:

Argumento	Obligatorio/opcional	Descripción
<i>string</i>	Obligatorio	Nombre del puerto de entrada.
<i>agrupar por</i>	Obligatorio	Nombre del puerto de entrada que contiene el identificador de grupo.

### Valor de retorno

TRUE para la fila que contiene el número más alto de campos que no están en blanco, FALSE para todas las demás filas.

### Ejemplo

La siguiente expresión usa la función `CONSOL_MOSTFILLED` para buscar la fila que contiene el mayor número de caracteres:

```

IF (CONSOL_MOSTFILLED(First_Name, Last_Name, GroupKey, GroupKey))
THEN
First_Name1 := CONSOL_GETROWFIELD(0)
Last_Name1 := CONSOL_GETROWFIELD(1)
GroupKey1 := CONSOL_GETROWFIELD(2)
ELSE
First_Name1 := First_Name
Last_Name1 := Last_Name
GroupKey1 := GroupKey
ENDIF

```

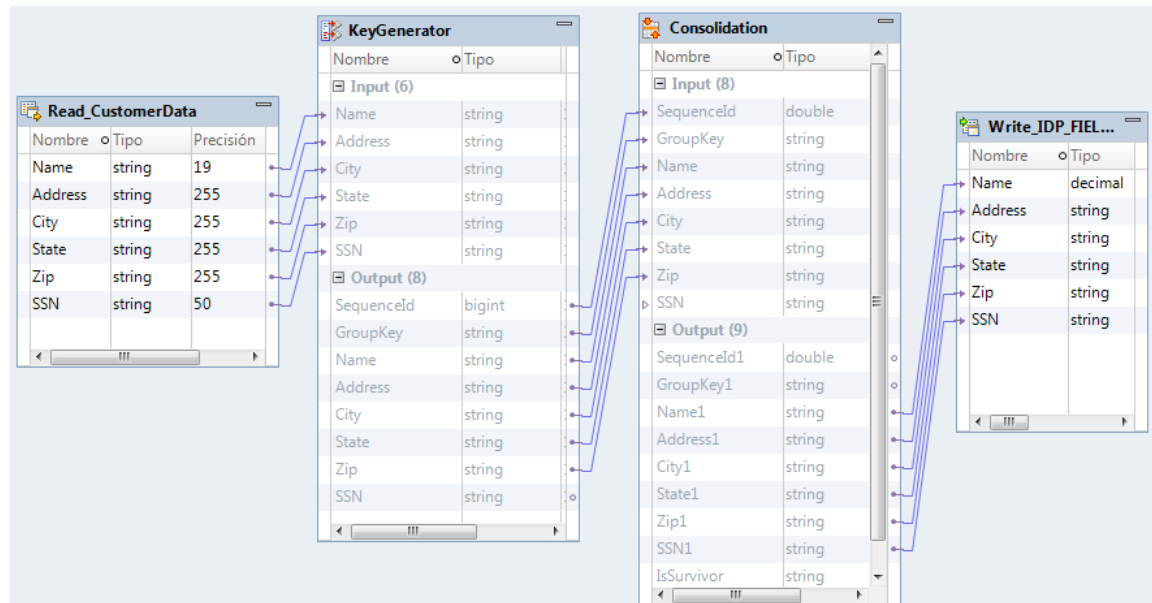
En esta expresión, la función `CONSOL_MOSTFILLED` analiza las filas de un grupo de registros e identifica una única fila. Las funciones `CONSOL_GETROWFIELD` utilizan números consecutivos para leer los valores de puerto de esa fila y escribir los valores en los puertos de salida.

## Ejemplo de asignación de consolidación

Su organización necesita consolidar registros de clientes duplicados. Para consolidar los registros de clientes, debe agrupar los datos con una transformación de generador de claves y utilizar la transformación de consolidación para consolidar los registros.

Una asignación se crea con un origen de datos que contenga los registros de clientes, una transformación de generador de claves, una transformación de consolidación y un destino de datos. Esta asignación agrupa los registros de clientes, consolida los grupos y escribe un único registro consolidado.

La siguiente figura muestra la asignación:



## Datos de entrada

Los datos de entrada que desea analizar contienen información de clientes.

La siguiente tabla contiene los datos de entrada para este ejemplo:

Nombre	Dirección	Ciudad	Estado	Código postal	SSN
Dennis Jones	100 All Saints Ave	New York	NY	10547	987-65-4320
Dennis Jones	1000 Alberta Rd	New York	NY	10547	987-65-4320
D Jones	100 All Saints Ave	New York	NY	10547-1521	

## Transformación de generador de claves

Utilice la transformación de generador de claves para agrupar los datos de entrada con base en el puerto de código postal.

La transformación devuelve los siguientes datos:

SequenceId	GroupKey	Nombre	Dirección	Ciudad	Estado	Código postal	SSN
1	10547	Dennis Jones	100 All Saints Ave	New York	NY	10547	987-65-4320

SequenceId	GroupKey	Nombre	Dirección	Ciudad	Estado	Código postal	SSN
2	10547	Dennis Jones	1000 Alberta Rd	New York	NY	10547	
3	10547	D Jones	100 All Saints Ave	New York	NY	10547-1521	987-65-4320

## Transformación de consolidación

Utilice la transformación de consolidación para generar el registro consolidado.

Configure la transformación de consolidación para utilizar el tipo de estrategia basada en filas. Seleccione la estrategia Exacto modal para elegir la fila con el mayor número de valores más frecuentes. La estrategia Exacto modal utiliza los valores de la fila para generar un registro consolidado. El registro consolidado es el registro con el valor "Y" en el puerto IsSurvivor.

La transformación devuelve los siguientes datos:

GroupKey	Nombre	Dirección	Ciudad	Estado	Código postal	SSN	IsSurvivor
10547	Dennis Jones	100 All Saints Ave	New York	NY	10547	987-65-4320	N
10547	Dennis Jones	1000 Alberta Rd	New York	NY	10547		N
10547	D Jones	100 All Saints Ave	New York	NY	10547-1521	987-65-4320	N
10547	D Jones	100 All Saints Ave	New York	NY	10547-1521	987-65-4320	Y

## Salida de la asignación de consolidación

Configure la transformación de consolidación de modo que la salida de asignación contenga solamente registros consolidados.

En este ejemplo, tiene certeza suficiente de que los valores más frecuentes seleccionados por la estrategia Exacto modal son los valores de puerto correctos. Para escribir solo registros consolidados en el destino de la asignación, seleccione la vista **Avanzadas** y establezca el modo de salida en "Solo superviviente".

Cuando se ejecuta la asignación, la salida de la asignación solo contiene registros consolidados.

## Cómo configurar una transformación de consolidación

Cuando configura la transformación de consolidación, debe elegir los tipos de estrategia, elegir las estrategias o escribir expresiones, seleccionar un puerto de agrupación y configurar opciones avanzadas.

1. Seleccione la vista **Consolidación**.
2. Elija un tipo de estrategia.
3. Configure la estrategia.
  - Para el tipo de estrategia de simple, seleccione una estrategia para cada puerto.
  - Para el tipo de estrategia basada en filas, seleccione una estrategia.

- Para el tipo de estrategia de avanzada, cree una expresión que utilice las funciones de consolidación.
4. En el campo Agrupar por, seleccione el puerto que contiene el identificador de grupo.
  5. Habilite la ordenación en la vista **Avanzada** si los datos de entrada no están ordenados.
  6. Configure la salida para que contenga los registros consolidados o todos los registros.

## CAPÍTULO 10

# Transformación de enmascaramiento de datos

Este capítulo incluye los siguientes temas:

- [Resumen de transformación de enmascaramiento de datos, 115](#)
- [Técnicas de enmascaramiento, 116](#)
- [Reglas de enmascaramiento, 127](#)
- [Formatos de máscara especiales, 132](#)
- [Archivo de valores predeterminados, 136](#)
- [Configuración de la transformación de enmascaramiento de datos, 137](#)
- [Propiedades de la sesión de transformación de enmascaramiento de datos, 139](#)
- [Ejemplo de enmascaramiento de datos, 140](#)
- [Propiedades avanzadas de Transformación de enmascaramiento de datos, 142](#)

## Resumen de transformación de enmascaramiento de datos

La transformación de enmascaramiento de datos cambia datos de producción confidenciales a datos de prueba realistas para entornos que no son de producción. La transformación de enmascaramiento de datos modifica los datos de origen en función de las técnicas de enmascaramiento configuradas para cada columna.

Cree datos enmascarados para el desarrollo, la prueba, la formación y la minería de datos de software. Puede conservar las relaciones de datos en los datos enmascarados y conservar la integridad referencial entre tablas de bases de datos.

La transformación de enmascaramiento de datos proporciona reglas de enmascaramiento basadas en los tipos de datos de origen y la técnica de enmascaramiento que configure para una columna. Para las cadenas, puede restringir los caracteres que se reemplazarán en una cadena. Puede restringir los caracteres que se aplicarán en una máscara. Para los números y fechas, puede proporcionar un intervalo de números para los datos enmascarados. Puede configurar un intervalo que sea fijo o una variación de porcentaje del número original. El servicio de integración de datos reemplaza los caracteres en función de la configuración regional que establezca para la transformación.

# Técnicas de enmascaramiento

La técnica de enmascaramiento es el tipo de enmascaramiento de datos que se aplica a la columna seleccionada.

Puede seleccionar una de las siguientes técnicas de enmascaramiento para una columna de entrada:

## Aleatorio

Produce resultados aleatorios no repetibles para los mismos datos de origen y reglas de enmascaramiento. Puede enmascarar los tipos de datos de fecha, numéricos y de cadena. El enmascaramiento aleatorio no requiere un valor de inicialización. Los resultados del enmascaramiento aleatorio no son deterministas.

## Expresión

Aplica una expresión a una columna de origen para crear o enmascarar datos. Puede enmascarar todos los tipos de datos.

## De claves

Reemplaza los datos de origen con valores repetibles. La transformación de enmascaramiento de datos proporciona resultados deterministas para los mismos datos de origen, las mismas reglas de enmascaramiento y el mismo valor de inicialización. Puede enmascarar los tipos de datos de fecha, numéricos y de cadena.

## Sustitución

Reemplaza una columna de datos con datos similares pero no relacionados desde un diccionario. Puede enmascarar el tipo de datos de cadena.

## Dependiente

Reemplaza los valores de una columna de origen según los valores de otra columna de origen. Puede enmascarar el tipo de datos de cadena.

## Formatos de máscara especiales

Número de tarjeta de crédito, dirección de correo electrónico, dirección IP, número de teléfono, SSN, SIN o dirección URL. La transformación de enmascaramiento de datos aplica reglas integradas para enmascarar de manera inteligente estos tipos comunes de datos confidenciales.

## Sin máscara

La transformación de enmascaramiento de datos no cambia los datos de origen.

La opción predeterminada es Sin máscara.

## Enmascaramiento aleatorio

El enmascaramiento aleatorio genera datos enmascarados aleatorios no deterministas. La transformación de enmascaramiento de datos devuelve valores diferentes cuando el mismo valor de origen aparece en filas diferentes. Puede definir reglas de enmascaramiento que afecten al formato de los datos que devuelve la transformación de enmascaramiento de datos. Enmascare valores numéricos, de cadena y de datos con el enmascaramiento aleatorio.

## Enmascaramiento de valores de cadena

Configure el enmascaramiento aleatorio para generar una salida aleatoria para columnas de cadena. Para configurar las limitaciones para cada carácter de la cadena de salida, configure un formato de máscara.

Configure los caracteres de filtro para definir qué caracteres de origen se van a enmascarar y los caracteres con los que se van a enmascarar.

Puede aplicar las siguientes reglas de enmascaramiento para un puerto de cadena:

#### **Intervalo**

Configure la longitud de cadena mínima y máxima. La transformación de enmascaramiento de datos devuelve una cadena de caracteres aleatorios con una longitud de cadena que se encuentra entre los valores mínimo y máximo.

#### **Formato de máscara**

Defina el tipo de carácter de sustitución para cada carácter de los datos de entrada. Puede limitar cada carácter a un tipo de carácter alfabético, numérico o alfanumérico.

#### **Caracteres de cadena de origen**

Defina los caracteres de la cadena de origen que quiera enmascarar. Por ejemplo enmascare el carácter del signo de número (#) siempre que aparezca en los datos de entrada. La transformación de enmascaramiento de datos enmascara todos los caracteres de entrada cuando la opción Caracteres de cadena de origen está en blanco.

#### **Caracteres de reemplazo de cadena de resultado**

Sustituya los caracteres de la cadena de destino por los caracteres que defina en la opción Caracteres de cadena de resultado. Por ejemplo, introduzca los siguientes caracteres para configurar cada máscara y que contenga caracteres alfabéticos en mayúscula de la A a la Z:

ABCDEFGHIJKLMNOPQRSTUVWXYZ

## **Enmascaramiento de valores numéricos**

Cuando se enmascaran datos numéricos, se puede configurar un intervalo de valores de salida para una columna. La transformación de enmascaramiento de datos devuelve un valor que se encuentra entre los valores mínimo y máximo del intervalo en función de la precisión del puerto. Para definir el intervalo, configure los intervalos mínimo y máximo o configure un intervalo de desenfoque en función de una variación del valor de origen inicial.

Puede configurar los siguientes parámetros de enmascaramiento para datos numéricos:

#### **Intervalo**

Defina un intervalo de valores de salida. La transformación de enmascaramiento de datos devuelve datos numéricos entre los valores mínimo y máximo.

#### **Intervalo de desenfoque**

Defina un intervalo de valores de salida que se encuentren dentro de una variación fija o una variación de porcentaje de los datos de origen. La transformación de enmascaramiento de datos devuelve datos numéricos que están cerca del valor de los datos de origen. Puede configurar un intervalo y un intervalo de desenfoque.

## **Enmascaramiento de valores de fecha**

Para enmascarar valores de fecha con enmascaramiento aleatorio, puede configurar un intervalo de fechas de salida o seleccionar una variación. Cuando se configura una variación, se selecciona una parte de la fecha para desenfocarla. Seleccione el año, el mes, el día, la hora, los minutos o los segundos. La transformación de enmascaramiento de datos devuelve una fecha que está dentro del intervalo configurado.

Puede configurar las siguientes reglas de enmascaramiento cuando se enmascara un valor de fecha y hora:

### Intervalo

Establece los valores mínimo y máximo que se van a devolver para el valor de fecha y hora seleccionado.

### Desenfoque

Enmascara una fecha en función de una variación que aplique a una unidad de la fecha. La transformación de enmascaramiento de datos devuelve una fecha que está dentro de la variación. Puede desenfocar el año, el mes, el día, la hora, los minutos o los segundos. Seleccione una variación baja o alta para aplicarla.

## Enmascaramiento de expresiones

El enmascaramiento de expresiones aplica una expresión a un puerto para cambiar los datos o crear nuevos datos. Al configurar máscaras de expresión, cree una expresión en el Editor de expresiones. Seleccione puertos de entrada y salida, funciones, variables y operadores para generar expresiones.

Puede concatenar datos de varios puertos para crear un valor para otro puerto. Por ejemplo, debe crear un nombre de inicio de sesión. El origen tiene las columnas de nombre de pila y apellido. Enmascare el nombre de pila y el apellido desde los archivos de búsqueda. En la transformación de enmascaramiento de datos, cree otro puerto llamado Inicio de sesión. Para el puerto Inicio de sesión, configure una expresión para concatenar la primera letra del nombre de pila con el apellido:

```
SUBSTR (FIRSTNM, 1, 1) || LASTNM
```

Seleccione funciones, puertos, variables y operadores desde la interfaz de apuntar y hacer clic para reducir al mínimo el número de errores en la construcción de expresiones.

El Editor de expresiones muestra los puertos de salida que no se han configurado para el enmascaramiento de expresiones. No puede utilizar la salida desde una expresión como entrada a otra expresión. Si añade el manualmente el nombre del puerto de salida a la expresión, puede obtener resultados inesperados.

Cuando crea una expresión, compruebe que la expresión devuelve un valor que coincide con el tipo de datos del puerto. La transformación de enmascaramiento de datos devuelve cero si el tipo de datos del puerto de expresión es numérico y el tipo de datos de la expresión no es el mismo. La transformación de enmascaramiento de datos devuelve valores nulos si el tipo de datos del puerto de expresión es una cadena y el tipo de datos de la expresión no es el mismo.

## Enmascaramiento de expresiones repetible

Configure una asignación de expresión repetible cuando una columna de origen se encuentra en más de una tabla y tiene que enmascarar la columna para cada tabla con el mismo valor.

Al configurar máscaras de expresión repetibles, la transformación de enmascaramiento de datos guarda los resultados de una expresión en una tabla de almacenamiento. Si la columna se encuentra en otra tabla de origen, la transformación de enmascaramiento de datos devuelve el valor enmascarado de la tabla de almacenamiento, en lugar de partir de la expresión.

## Nombre del diccionario

Al configurar máscaras de expresión repetibles debe especificar un nombre de diccionario. El nombre del diccionario es una clave que permite varias transformaciones de enmascaramiento de datos para generar los mismos valores enmascarados desde los mismos valores de origen. Defina el mismo nombre de diccionario en cada transformación de enmascaramiento de datos. El nombre del diccionario puede ser cualquier texto.

## Tabla de almacenamiento

La tabla de almacenamiento contiene los resultados del enmascaramiento de expresiones repetible entre sesiones. Una fila de la tabla de almacenamiento contiene la columna de origen y un par de valores enmascarados. La tabla de almacenamiento para el enmascaramiento de expresiones es una tabla independiente de la tabla de almacenamiento para la máscara de sustitución.

Cada vez que la transformación de enmascaramiento de datos enmascara un valor con una expresión repetible, busca en la tabla de almacenamiento por el nombre de diccionario, la configuración regional, el nombre de columna y el valor de entrada. Si encuentra una fila en la tabla de almacenamiento, devuelve el valor enmascarado de la tabla de almacenamiento. Si la transformación de enmascaramiento de datos no encuentra una fila, genera un valor enmascarado de la expresión para la columna.

Es necesario cifrar las tablas de almacenamiento para el enmascaramiento de expresiones cuando tiene datos no cifrados en el almacenamiento y utiliza el mismo nombre de diccionario como clave.

## Cifrado de tablas de almacenamiento para el enmascaramiento de expresiones

Puede utilizar funciones de codificación del idioma de transformación para cifrar tablas de almacenamiento. Es necesario que cifre las tablas de almacenamiento cuando el cifrado de almacenamiento está habilitado.

1. Cree una asignación con la tabla de almacenamiento IDM\_EXPRESSION\_STORAGE como origen.
2. Cree una transformación de enmascaramiento de datos.
3. Aplique la técnica de enmascaramiento de expresión en los puertos con valor enmascarado.
4. Utilice la siguiente expresión en el puerto MASKEDVALUE:

```
Enc_Base64(AES_Encrypt(MASKEDVALUE, Key))
```

5. Vincule los puertos con el destino.

## Ejemplo

Por ejemplo, una tabla Employees contiene las siguientes columnas:

```
FirstName  
LastName  
LoginID
```

En la transformación de enmascaramiento de datos, enmascare LoginID con una expresión que combine FirstName y LastName. Configure la máscara de expresión para que sea repetible. Especifique un nombre de diccionario como clave para el enmascaramiento repetible.

La tabla Computer\_Users contiene LoginID, pero no contiene las columnas FirstName o LastName:

```
Dept  
LoginID  
Password
```

Para enmascarar LoginID en Computer\_Users con el mismo LoginID que Employees, configure el enmascaramiento de expresiones para la columna LoginID. Habilite el enmascaramiento repetible y especifique el mismo nombre de diccionario que ha definido para la tabla LoginID Employees. El servicio de integración recupera los valores de LoginID desde la tabla de almacenamiento.

Cree una expresión predeterminada para utilizar cuando el servicio de integración no pueda encontrar una fila en la tabla de almacenamiento para LoginID. La tabla Computer\_Users no tiene las columnas FirstName o LastName, de modo que la expresión crea un LoginID con menos sentido.

## Scripts de la tabla de almacenamiento

Informatica proporciona los scripts que puede ejecutar para crear la tabla de almacenamiento. Los scripts están en la ubicación siguiente:

```
<PowerCenter installation directory>\client\bin\Extensions\DataMasking
```

El directorio contiene un script para las bases de datos de Sybase, Microsoft SQL Server, IBM DB2 y Oracle. Cada script se llama <Expression\_<database type>.

## Reglas y directrices para el enmascaramiento de expresiones

Utilice las siguientes reglas y directrices para el enmascaramiento de expresiones:

- No puede utilizar la salida desde una expresión como entrada a otra expresión. Si añade el manualmente el nombre del puerto de salida a la expresión, puede obtener resultados inesperados.
- Utilice el método de apuntar y hacer clic para generar expresiones. Seleccione funciones, puertos, variables y operadores desde la interfaz de apuntar y hacer clic para reducir al mínimo el número de errores en la construcción de expresiones.
- Si la transformación de enmascaramiento de datos está configurada para el enmascaramiento repetible y la tabla de almacenamiento no existe, el servicio de integración sustituye los datos de origen con los valores predeterminados.

## Enmascaramiento de claves

Una columna configurada para el enmascaramiento de claves devuelve datos enmascarados deterministas siempre que el valor de origen y el valor de inicialización son los mismos. La transformación de enmascaramiento de datos devuelve valores únicos para la columna.

Cuando se configura una columna para el enmascaramiento de claves, la transformación de enmascaramiento de datos crea un valor de inicialización para la columna. Puede modificar el valor de inicialización para producir datos repetibles entre diferentes transformaciones de enmascaramiento de datos. Por ejemplo, configure el enmascaramiento de claves para aplicar la integridad referencial. Utilice el mismo valor de inicialización para enmascarar una clave principal en una tabla y el valor de clave externa en otra tabla.

Puede definir reglas de enmascaramiento que afecten al formato de los datos que devuelve la transformación de enmascaramiento de datos. Enmascare los valores de cadena y numéricos con el enmascaramiento de claves.

## Enmascaramiento de valores de cadena

Puede configurar el enmascaramiento de claves para generar salidas repetibles para cadenas. Configure un formato de máscara que defina las limitaciones de cada carácter en la cadena de salida. Configure los caracteres de cadena de origen que definan qué caracteres de origen enmascarar. Configure los caracteres de reemplazo de cadena de resultado para limitar los datos enmascarados a determinados caracteres.

Puede configurar las siguientes reglas de enmascaramiento para cadenas de enmascaramiento de claves:

### Semilla

Aplique un valor de inicialización para generar datos enmascarados deterministas para una columna. Puede introducir un número entre 1 y 1.000.

### Formato de máscara

Defina el tipo de carácter de sustitución para cada carácter de los datos de entrada. Puede limitar cada carácter a un tipo de carácter alfabético, numérico o alfanumérico.

### Caracteres de cadena de origen

Defina los caracteres de la cadena de origen que quiera enmascarar. Por ejemplo enmascare el carácter del signo de número (#) siempre que aparezca en los datos de entrada. La transformación de enmascaramiento de datos enmascara todos los caracteres de entrada cuando la opción Caracteres de cadena de origen está en blanco. La transformación de enmascaramiento de datos no siempre devuelve datos únicos si el número de caracteres de cadena de origen es inferior al número de caracteres de cadena de resultado.

### Caracteres de cadena de resultado

Sustituya los caracteres de la cadena de destino por los caracteres que defina en la opción Caracteres de cadena de resultado. Por ejemplo, introduzca los siguientes caracteres para configurar cada máscara y que contenga todos los caracteres alfabéticos en mayúsculas:

ABCDEFGHIJKLMNOPQRSTUVWXYZ

## Enmascaramiento de valores numéricos

Configure el enmascaramiento de claves de datos de origen numéricos para generar una salida determinista. Cuando se configura en una columna el enmascaramiento de claves numérico, se asigna un valor de inicialización aleatorio a la columna. Cuando la transformación de enmascaramiento de datos enmascara los datos de origen, aplica un algoritmo de enmascaramiento que requiere una semilla.

Puede cambiar el valor de inicialización de una columna para obtener resultados repetibles si el mismo valor de origen aparece en una columna diferente. Por ejemplo, si quiere conservar una relación de claves principal-externa entre dos tablas. En cada transformación de enmascaramiento de datos, introduzca el mismo valor de inicialización para la columna de claves primarias que el valor de inicialización para la columna de claves externas. La transformación de enmascaramiento de datos proporciona resultados deterministas para los mismo valores numéricos. La integridad referencial se mantiene entre las tablas.

## Enmascaramiento de valores de fecha y hora

Si puede configurar el enmascaramiento de claves para valores de fecha y hora, la transformación de enmascaramiento de datos requiere un número aleatorio como semilla. Puede cambiar la semilla para que coincida con el valor de inicialización de otra columna y así que se devuelvan valores de fecha y hora repetibles entre las columnas.

La transformación de enmascaramiento de datos puede enmascarar fechas entre 1753 y 2400 con enmascaramiento de claves. Si el año de origen es un año bisiesto, la transformación de enmascaramiento de datos devuelve un año que es también un año bisiesto. Si el mes de origen tiene 31 días, la transformación de enmascaramiento de datos devuelve un mes que tiene también 31 días. Si el mes de origen es febrero, la transformación de enmascaramiento de datos devuelve el mes de febrero.

La transformación de enmascaramiento de datos siempre genera fechas válidas.

## Máscara de sustitución

La máscara de sustitución reemplaza una columna de datos con datos similares pero no relacionados. Utilice la máscara de sustitución para reemplazar datos de producción con datos de prueba realistas. Al configurar la máscara de sustitución, defina el diccionario que contiene los valores de sustitución.

La transformación de enmascaramiento de datos realiza una búsqueda en el diccionario que haya configurado. La transformación de enmascaramiento de datos reemplaza datos de origen con datos del diccionario. Los archivos de diccionario pueden contener datos de cadena, valores de fecha y hora, enteros y números de coma flotante. Especifique los valores de fecha y hora en el siguiente formato:

mm/dd/yyyy

Puede sustituir datos con valores repetibles o no repetibles. Si selecciona valores repetibles, la transformación de enmascaramiento de datos genera resultados deterministas para los mismos datos de origen y valor de semilla. Debe configurar un valor de semilla para sustituir datos con resultados deterministas. El servicio de integración mantiene una tabla de almacenamiento de origen y valores enmascarados para el enmascaramiento repetible.

Puede sustituir más de una columna de datos con valores enmascarados desde la misma fila del diccionario. Configure la máscara de sustitución para una columna de entrada. Configure el enmascaramiento de datos dependiente para otras columnas que reciben datos enmascarados de la fila del diccionario.

## Diccionarios

Un diccionario es una tabla de referencia que contiene los datos de sustitución y un número de serie para cada fila de la tabla. Cree una tabla de referencia para la máscara de sustitución desde un archivo sin formato o una tabla relacional que importe al repositorio de modelos.

La transformación de enmascaramiento de datos genera un número para recuperar una fila del diccionario por el número de serie. La transformación de enmascaramiento de datos genera una clave hash para la máscara de sustitución repetible o un número aleatorio para la máscara no repetible. Puede configurar una condición de búsqueda adicional si configura la máscara de sustitución repetible.

Puede configurar un diccionario para enmascarar más de un puerto en la transformación de enmascaramiento de datos.

Cuando la transformación de enmascaramiento de datos recupera los datos de sustitución desde un diccionario, la transformación no comprueba si el valor de los datos de sustitución es el mismo que el valor original. Por ejemplo, la transformación de enmascaramiento de datos puede sustituir el nombre John con el mismo nombre (John) desde un archivo de diccionario.

El siguiente ejemplo muestra una tabla de diccionario que contiene el nombre de pila y el sexo:

SNO	SEX	FIRSTNAME
1	M	Adam
2	M	Adeel
3	M	Adil
4	F	Alice
5	F	Alison

En esta diccionario, el primer campo en la fila es el número de serie y el segundo campo es el sexo. El servicio de integración siempre busca un registro de diccionario por el número de serie. Puede añadir el sexo como una condición de búsqueda si configura el enmascaramiento repetible. El servicio de integración recupera una fila desde el diccionario mediante una clave hash y encuentra una fila con el sexo que coincide con el sexo de los datos de origen.

Utilice las siguientes reglas y directrices cuando cree una tabla de referencia:

- Cada registro de la tabla debe tener un número de serie.
- Los números de serie son enteros secuenciales que empieza en el uno. No puede faltar un número en la secuencia de los números de serie.
- La columna de número de serie puede estar en cualquier lugar de la fila de una tabla. Puede tener cualquier etiqueta.

Si utiliza una tabla de archivo sin formato para crear la tabla de referencia, utilice las siguientes reglas y directrices:

- La primera fila de la tabla del archivo sin formato debe tener etiquetas de columnas para identificar los campos de cada registro. Los campos están separados por comas. Si la primera fila no contiene etiquetas de columnas, el servicio de integración toma los valores de los campos de la primera fila como los nombres de columna.
- Si crea una tabla de archivo sin formato en Windows y la copia en un equipo UNIX, compruebe que el formato de archivo es correcto para UNIX. Por ejemplo, Windows y UNIX utilizan diferentes caracteres para el final del marcador de línea.

## Tablas de almacenamiento

La transformación de enmascaramiento de datos mantiene tablas de almacenamiento para la sustitución repetible entre sesiones. Una fila de la tabla de almacenamiento contiene la columna de origen y un par de valores enmascarados. Cada vez que la transformación de enmascaramiento de datos enmascara un valor con una sustitución repetible, busca en la tabla de almacenamiento por el nombre de diccionario, la configuración regional, el nombre de columna, el valor de entrada y la semilla. Si encuentra una fila, devuelve el valor enmascarado de la tabla de almacenamiento. Si la transformación de enmascaramiento de datos no encuentra una fila, recupera una del diccionario con una clave hash.

El formato del nombre de diccionario en la tabla de almacenamiento es diferente para un diccionario de archivo sin formato y un diccionario relacional. Un nombre de diccionario de archivo sin formato se identifica por el nombre de archivo. El nombre del diccionario relacional tiene la siguiente sintaxis:

```
<Connection object>_<dictionary table name>
```

Informatica proporciona los scripts que puede ejecutar para crear una tabla de almacenamiento relacional. Los scripts están en la ubicación siguiente:

```
<PowerCenter Client installation directory>\client\bin\Extensions\DataMasking
```

El directorio contiene un script para las bases de datos de Sybase, Microsoft SQL Server, IBM DB2 y Oracle. Cada script se llama Substitution\_<database type>. Puede crear una tabla en una base de datos diferente si configura las instrucciones de SQL y la clave principal se restringe.

Es necesario para cifrar las tablas de almacenamiento para la máscara de sustitución cuando tiene datos no cifrados en el almacenamiento y utiliza el mismo valor de semilla y diccionario para cifrar las mismas columnas.

## Cifrado de tablas de almacenamiento para la máscara de sustitución

Puede utilizar funciones de codificación del idioma de transformación para cifrar tablas de almacenamiento. Es necesario que cifre las tablas de almacenamiento cuando el cifrado de almacenamiento está habilitado.

1. Cree una asignación con la tabla de almacenamiento IDM\_SUBSTITUTION\_STORAGE como origen.
2. Cree una transformación de enmascaramiento de datos.
3. Aplique la técnica de máscara de sustitución en el valor de entrada y en los puertos de valores enmascarados.

4. Utilice la siguiente expresión en el puerto INPUTVALUE:

```
Enc_Base64(AES_Encrypt(INPUTVALUE, Key))
```

5. Utilice la siguiente expresión en el puerto MASKEDVALUE:

```
Enc_Base64(AES_Encrypt(MASKEDVALUE, Key))
```

6. Vincule los puertos con el destino.

## Propiedades de la máscara de sustitución

Puede configurar las siguientes reglas de enmascaramiento para la máscara de sustitución:

- **Salida repetible.** Devuelve resultados deterministas entre sesiones. La transformación de enmascaramiento de datos almacena valores enmascarados en la tabla de almacenamiento.
- **Valor de semilla.** Aplique un valor de semilla para generar datos enmascarados deterministas para una columna. Especifique un número entre 1 y 1.000.
- **Salida única.** Fuerce la transformación de enmascaramiento de datos para crear valores de salida únicos para valores de entrada únicos. No se puede enmascarar dos valores de entrada en el mismo valor de salida. El diccionario debe tener suficientes filas únicas para habilitar salidas únicas. Cuando deshabilita la salida única, es posible que la transformación de enmascaramiento de datos no enmascare valores de entrada en valores de salida únicos. El diccionario puede contener menos filas.
- **Puerto único.** El puerto utilizado para identificar los registros únicos para la máscara de sustitución. Por ejemplo, desea enmascarar los nombres de pila en una tabla llamado Cliente. Si selecciona la columna de tabla que contiene los nombres de pila como el puerto único, la transformación de enmascaramiento de datos reemplaza los nombres de pila duplicados por el mismo valor enmascarado. Si selecciona la columna Customer\_ID como el puerto único, la transformación de enmascaramiento de datos reemplaza cada nombre de pila con un valor único.
- **Información del diccionario.** Configure la tabla de referencia que contiene los valores de los datos de sustitución. Haga clic en **Seleccionar origen** para seleccionar una tabla de referencia.
  - **Nombre del diccionario.** Muestra el nombre de la tabla de referencia que seleccione.
  - **Columna de número de serie.** Seleccione la columna en el diccionario que contiene el número de serie.
  - **Columna de salida.** Seleccione la columna para volver a la transformación de enmascaramiento de datos.
- **Condición de búsqueda.** Configure una condición de búsqueda para cualificar qué fila del diccionario utilizar para la máscara de sustitución. La condición de búsqueda es similar a la cláusula WHERE de una consulta SQL. Cuando configure una condición de búsqueda compare el valor de una columna en el origen con una columna en el diccionario.

Por ejemplo, desea enmascarar el nombre de pila. Los datos de origen y el diccionario tienen una columna de nombre de pila y una columna de sexo. Puede añadir una condición para que cada nombre de pila de mujer sea reemplazado con un nombre de mujer del diccionario. La condición de búsqueda compara el sexo del origen con el sexo del diccionario.

  - **Puerto de entrada.** Columna de datos de origen que utilizar en la búsqueda.
  - **Columna de diccionario.** Columna del diccionario que comparar con el puerto de entrada.

## Reglas y directrices de la máscara de sustitución

Utilice las siguientes reglas y directrices para la máscara de sustitución:

- Si una tabla de almacenamiento no existe para una única máscara de sustitución repetible, la sesión genera un error.

- Si el diccionario no contiene filas, la transformación de enmascaramiento de datos devuelve un mensaje de error.
- Cuando la transformación de enmascaramiento de datos encuentra un valor de entrada con la configuración regional, el diccionario y la semilla en la tabla de almacenamiento, recupera el valor enmascarado, incluso si la fila ya no está en el diccionario.
- Si elimina un objeto de conexión o modifica el diccionario, debe truncar la tabla de almacenamiento. De lo contrario, puede obtener resultados inesperados.
- Si el número de valores en el diccionario es inferior al número de valores únicos de los datos de origen, la transformación de enmascaramiento de datos no puede enmascarar los datos con valores únicos repetibles. La transformación de enmascaramiento de datos devuelve un mensaje de error.

## Enmascaramiento dependiente

El enmascaramiento dependiente sustituye varias columnas de datos de origen con los datos de la misma fila del diccionario.

Cuando la transformación de enmascaramiento de datos realiza la máscara de sustitución para varias columnas, los datos enmascarados pueden contener combinaciones no realistas de los campos. Puede configurar el enmascaramiento dependiente para sustituir datos por varias columnas de entrada desde la misma fila del diccionario. Los datos enmascarados reciben combinaciones válidas como "Nueva York, Nueva York" o "Chicago, Illinois".

Al configurar el enmascaramiento dependiente, primero debe configurar una columna de entrada para la máscara de sustitución. Configure otras columnas de entrada para que dependan de la columna de sustitución. Por ejemplo, seleccione la columna de código postal para la máscara de sustitución y seleccione las columnas de ciudad y estado para que dependan de la columna de código postal. El enmascaramiento dependiente garantiza que los valores de ciudad y estado sustituidos son válidos para valor del código postal sustituido.

**Nota:** No se puede configurar una columna para el enmascaramiento dependiente sin configurar primero una columna para la máscara de sustitución.

Configure las siguientes reglas de enmascaramiento cuando configure una columna para el enmascaramiento de dependientes:

### Columna de dependientes

El nombre de la columna de entrada que ha configurado para la máscara de sustitución. La transformación de enmascaramiento de datos recupera los datos de sustitución de un diccionario utilizando las reglas de enmascaramiento de esa columna. La columna que ha configurado para la máscara de sustitución se convierte en la columna de clave para recuperar los datos enmascarados del diccionario.

### Columna de salida

El nombre de la columna del diccionario que contiene el valor para la columna que está configurando con el enmascaramiento dependiente.

## Ejemplo de enmascaramiento dependiente

Un diccionario de enmascaramiento de datos puede contener filas de dirección con los siguientes valores:

SNO	CALLE	CIUDAD	ESTADO	CÓDIGO POSTAL	PAÍS
1	32 Apple Lane	Chicago	IL	61523	ES
2	776 Ash Street	Dallas	TX	75240	ES
3	2229 Big Square	Atleeville	TN	38057	ES
4	6698 Cowboy Street	Houston	TX	77001	ES

Es necesario enmascarar datos de origen con combinaciones válidas de la ciudad, estado y código postal del diccionario de direcciones.

Configure el puerto de código postal para la máscara de sustitución. Especifique las siguientes reglas de enmascaramiento para el puerto de código postal:

Regla	Valor
Nombre del diccionario	Dirección
Columna de número de serie	SNO
Columna de salida	CÓDIGO POSTAL

Configure el puerto de ciudad para el enmascaramiento dependiente. Especifique las siguientes reglas de enmascaramiento para el puerto de ciudad:

Regla	Valor
Columna de dependientes	CÓDIGO POSTAL
Columna de salida	Ciudad

Configure el puerto de estado para el enmascaramiento dependiente. Especifique las siguientes reglas de enmascaramiento para el puerto de estado:

Regla	Valor
Columna de dependientes	CÓDIGO POSTAL
Columna de salida	Estado

Cuando la transformación de enmascaramiento de datos enmascara el código postal, devuelve la ciudad y el estado correctos para el código postal de la fila de diccionario.

# Reglas de enmascaramiento

Las reglas de enmascaramiento son opciones que puede configurar después de seleccionar la técnica de enmascaramiento.

Si selecciona una técnica de enmascaramiento aleatoria o de claves, puede configurar el formato de máscara, los caracteres de cadena de origen y los caracteres de cadena de resultado. Puede configurar el intervalo o el desenfoque con el enmascaramiento aleatorio.

En la siguiente tabla se describen las reglas de enmascaramiento que puede configurar para cada técnica de enmascaramiento:

Regla de enmascaramiento	Descripción	Técnica de enmascaramiento	Tipo de datos de origen
Formato de máscara	Máscara que limita cada carácter de una cadena de salida a un carácter alfabético, numérico o alfanumérico.	Aleatoria y de claves	Cadena
Caracteres de cadena de origen	Conjunto de caracteres de origen que se van a enmascarar o excluir del enmascaramiento.	Aleatoria y de claves	Cadena
Caracteres de reemplazo de cadena de resultado	Conjunto de caracteres que se van a incluir o excluir de la máscara.	Aleatoria y de claves	Cadena

Regla de enmascaramiento	Descripción	Técnica de enmascaramiento	Tipo de datos de origen
Intervalo	<p>Un intervalo de valores de salida.</p> <ul style="list-style-type: none"> <li>- Numeric. La transformación de enmascaramiento de datos devuelve datos numéricos que se encuentran entre los valores mínimo y máximo.</li> <li>- Cadena. Devuelve una cadena de caracteres aleatorios entre la longitud de cadena mínima y la máxima.</li> <li>- Date/Time. Devuelve una fecha y una hora que están entre los valores de fecha y hora mínimo y máximo.</li> </ul>	Aleatoria	Numérico Cadena Fecha/Hora
Desenfoco	Intervalo de valores de salida con una variación fija o de porcentaje de los datos de origen. La transformación de enmascaramiento de datos devuelve datos que están cerca del valor de los datos de origen. Las columnas de fecha y hora requieren una variación fija. las columnas requieren una variación fija.	Aleatoria	Numérico Fecha/Hora

## Formato de máscara

Configure un formato de máscara para limitar cada carácter de la columna de salida a un carácter alfabético, numérico o alfanumérico. Utilice los siguientes caracteres para definir un formato de máscara:

A, D, N, X, +, R

**Nota:** El formato de máscara contiene caracteres en mayúscula. Si introduce un carácter de máscara en minúscula, la transformación de enmascaramiento de datos convierte este carácter a mayúsculas.

En la siguiente tabla se describen los caracteres de formato de máscara:

Carácter	Descripción
A	Caracteres alfabéticos. Por ejemplo, caracteres ASCII de la "a" a la "z" y de la "A" a la "Z".
D	Dígitos de 0 a 9. La transformación de enmascaramiento de datos devuelve una "X" para los caracteres distintos de los dígitos de 0 a 9.
N	Caracteres alfanuméricos. Por ejemplo, caracteres ASCII de la "a" a la "z", de la "A" a la "Z" y del 0 al 9.
X	Cualquier carácter. Por ejemplo, un carácter alfanumérico o un símbolo.
+	Sin máscara.
R	Caracteres restantes. R especifica que los caracteres restantes de la cadena pueden ser de cualquier tipo. R debe aparecer como último carácter de la máscara.

Por ejemplo, el nombre de un departamento tiene el siguiente formato:

```
nnn-<department_name>
```

Puede configurar una máscara para hacer que los tres primeros caracteres sean numéricos, el nombre del departamento sea alfabético y el guión se quede en la salida. Configure el siguiente formato de máscara:

```
DDD+AAAAAAAAAAAAAAAA
```

La transformación de enmascaramiento de datos reemplaza los tres primeros caracteres por caracteres numéricos. No reemplaza el cuarto carácter. La transformación de enmascaramiento de datos reemplaza los caracteres restantes por caracteres alfabéticos.

Si no define un formato de máscara, la transformación de enmascaramiento de datos reemplaza cada uno de los caracteres de origen por un carácter. Si el formato de la máscara es más largo que la cadena de entrada, la transformación de enmascaramiento de datos ignora los caracteres extra del formato de máscara. Si el formato de la máscara es más corto que la cadena de origen, la transformación de enmascaramiento de datos no enmascara los caracteres finales de la cadena de origen.

**Nota:** No puede configurar un formato de máscara con la opción de intervalo.

## Caracteres de cadena de origen

Los caracteres de cadena de origen son caracteres de origen que selecciona para enmascararlos o no enmascararlos. La posición de los caracteres en la cadena de origen es indiferente. Los caracteres de origen distinguen mayúsculas de minúsculas.

Puede configurar tantos caracteres como desee. Cuando la opción Caracteres está en blanco, la transformación de enmascaramiento de datos reemplaza todos los caracteres de origen de la columna.

Seleccione una de las siguientes opciones para los caracteres de cadena de origen:

### Enmascarar solo

La transformación de enmascaramiento de datos enmascara los caracteres del origen que haya configurado como caracteres de cadena de origen. Por ejemplo, si introduce los caracteres A, B y c, la transformación de enmascaramiento de datos reemplaza A, B o c por un carácter diferente siempre que aparezca en los datos de origen. Cualquier carácter de origen que no sea A, B o c no cambiará. La máscara distingue mayúsculas de minúsculas.

### Enmascarar todos excepto

Enmascara todos los caracteres exceptuando los caracteres de cadena de origen que aparezcan en la cadena de origen. Por ejemplo, si introduce el carácter de origen de filtro "-" y selecciona Enmascarar todos excepto, la transformación de enmascaramiento de datos no reemplaza el carácter "-" cuando aparece en los datos de origen. El resto de caracteres de origen cambiará.

### Ejemplo de cadena de origen

Un archivo de origen tiene una columna que se llama Dependientes. La columna Dependientes contiene más de un nombre separados por comas. Tiene que enmascarar la columna Dependientes y conservar la coma en los datos de prueba para delimitar los nombres.

En la columna Dependientes, seleccione Caracteres de cadena de origen. Seleccione No enmascarar y escriba "," como el carácter de origen que hay que ignorar. No escriba las comillas.

La transformación de enmascaramiento de datos reemplaza a todos los caracteres de la cadena de origen exceptuando las comas.

## Caracteres de reemplazo de cadena de resultado

Los caracteres de reemplazo de cadena de resultado son caracteres que selecciona como caracteres de sustitución en los datos enmascarados. Si configura caracteres de reemplazo de cadena de resultado, la transformación de enmascaramiento de datos reemplaza los caracteres de la cadena de origen por los caracteres de reemplazo de cadena de resultado. Para evitar que se genere la misma salida para valores de entrada diferentes, configure un intervalo amplio de caracteres de sustitución, o enmascare solo unos cuantos caracteres de origen. La posición de cada carácter en la cadena es indiferente.

Seleccione una de las siguientes opciones para los caracteres de reemplazo de cadena de resultado:

#### Utilizar solo

Enmascare el origen solo con los caracteres que defina como caracteres de reemplazo de cadena de resultado. Por ejemplo, si introduce los caracteres A, B y c, la transformación de enmascaramiento de datos reemplaza cada carácter de la columna de origen con una A, B o c. La palabra "horse" se podría reemplazar por "BAcBA".

#### Utilizar todos excepto

Enmascare el origen con cualquier carácter exceptuando los caracteres que defina como caracteres de reemplazo de cadena de resultado. Por ejemplo, si introduce A, B y c como caracteres de reemplazo de cadena de resultado, los datos enmascarados no tendrán nunca los caracteres A, B o c.

### Ejemplo de caracteres de reemplazo de cadena de resultado

Para reemplazar todas las comas de la columna Dependientes por punto y coma, realice las siguientes tareas:

1. Configure la coma como un carácter de cadena de origen y seleccione Enmascarar solo.  
La transformación de enmascaramiento de datos enmascara solo la coma cada vez que aparece en la columna Dependientes.
2. Configure el punto y coma como un carácter de reemplazo de cadena de resultado y seleccione Utilizar solo.  
La transformación de enmascaramiento de datos reemplaza las comas de la columna Dependientes por un punto y coma.

## Intervalo

Establezca un intervalo para los datos numéricos, de fecha o de cadena. Al definir un intervalo para los valores numéricos o de fecha, la transformación de enmascaramiento de datos enmascara los datos de origen con un valor comprendido entre los valores mínimo y máximo. Al configurar un intervalo para una cadena se está estableciendo un intervalo de longitudes de cadena.

### Intervalo de cadenas

Al configurar el enmascaramiento de cadenas aleatorias, la transformación de enmascaramiento de datos genera cadenas cuya longitud no es la misma que la de la cadena de origen. De forma opcional, puede configurar una anchura mínima y máxima. Estos valores de anchura deben ser números positivos y enteros. Cada anchura debe ser menor o igual que la precisión del puerto.

### Intervalo numérico

Establezca los valores mínimo y máximo de una columna numérica. El valor máximo debe ser menor o igual que la precisión del puerto. El intervalo predeterminado va del uno a la longitud de la precisión del puerto.

### Intervalo de fechas

Establezca los valores mínimo y máximo para un valor de fecha y hora. Los campos mínimo y máximo contienen las fechas mínima y máxima predeterminadas. El formato de la fecha y hora predeterminadas es MM/DD/AAAA HH24:MI:SS. La fecha y hora máxima debe ser posterior a la mínima.

## Desenfoque

El desenfoque crea un valor de salida de una variación fija o de porcentaje del valor de fecha de origen. Configure el desenfoque para devolver un valor aleatorio próximo al valor original. Es posible desenfocar valores numéricos y de fecha.

### Desenfoque de valores numéricos

Seleccione una variación fija o de porcentaje para desenfocar un valor numérico de origen. El valor de desenfoque inferior es una variación por debajo del valor de origen. El valor de desenfoque superior es una variación por encima del valor de origen. Tanto el valor inferior como el superior deben ser igual o mayor que cero. Cuando la transformación de enmascaramiento de datos devuelva una fecha enmascarada, la fecha numérica estará dentro del intervalo establecido.

En la siguiente tabla se indican los resultados de enmascaramiento para los valores del intervalo de desenfoque cuando el valor de origen de entrada es 66:

Tipo de desenfoque	Bajo	Alto	Resultado
Fijo	0	10	Entre 66 y 76
Fijo	10	0	Entre 56 y 66
Fijo	10	10	Entre 56 y 76
Porcentaje	0	50	Entre 66 y 99

Tipo de desenfoque	Bajo	Alto	Resultado
Porcentaje	50	0	Entre 33 y 66
Porcentaje	50	50	Entre 33 y 99

## Desenfoque de valores de fecha

Configure el desenfoque para enmascarar una fecha como una variación de la fecha de origen. Seleccione una unidad de la fecha en la que desee aplicar la variación. Puede seleccionar el año, el mes, el día o la hora. Introduzca los límites inferior y superior para establecer una variación tanto por arriba como por abajo de la unidad de la fecha de origen. La transformación de enmascaramiento de datos aplica la variación y devuelve una fecha que esté incluida en ella.

Por ejemplo, para restringir la fecha enmascarada a una fecha incluida en los dos años siguientes a la fecha de origen, seleccione el año como la unidad. Introduzca dos fechas como límites inferior y superior. Si una fecha de origen es el 02/02/2006, la transformación de enmascaramiento de datos devolverá una fecha comprendida entre el 02/02/2004 y el 02/02/2008.

De forma predeterminada, la unidad de desenfoque es el año.

## Formatos de máscara especiales

Los formatos de máscara especiales son máscaras que puede aplicar a los tipos de datos comunes. Con un formato de máscara especial, la transformación de enmascaramiento de datos devuelve un valor enmascarado con un formato realista, pero que no es un valor válido.

Por ejemplo, cuando enmascara un SSN, la transformación de enmascaramiento de datos devuelve un SSN que tiene el formato correcto pero no es válido. Puede configurar máscaras repetibles para números de la seguridad social.

Configure máscaras especiales para los siguientes tipos de datos:

- Números de la seguridad social
- Números de tarjeta de crédito
- Números de teléfono
- Direcciones URL
- Direcciones de correo electrónico
- Direcciones IP
- Números de la seguridad social

Cuando el formato de datos de origen o el tipo de datos no es válido para una máscara, el servicio de integración de datos aplica una máscara predeterminada a los datos. El servicio de integración aplica valores enmascarados desde el archivo de valores predeterminados. Puede editar el archivo de valores predeterminados y modificar estos valores predeterminados.

## Enmascaramiento del número de la tarjeta de crédito

La transformación de enmascaramiento de datos genera lógicamente un número de tarjeta de crédito válido cuando enmascara un número de tarjeta de crédito válido. La longitud del número de tarjeta de crédito de origen debe encontrarse entre 13 y 19 dígitos. El número de tarjeta de crédito de entrada debe tener una suma de comprobación válida en función de las reglas del sector de tarjetas de crédito.

El número de tarjeta de crédito de origen puede contener números, espacios y guiones. Si la tarjeta de crédito tiene caracteres incorrectos, o una longitud que no sea la adecuada, el servicio de integración escribe un error para el registro de la sesión. El servicio de integración aplica una máscara de números de tarjetas de crédito predeterminada cuando los datos de origen no son válidos.

La transformación de enmascaramiento de datos no enmascara el número de identificación bancaria (BIN) de seis dígitos. Por ejemplo, la transformación de enmascaramiento de datos puede enmascarar el número de tarjeta de crédito 4539 1596 8210 2773 como 4539 1516 0556 7067. La transformación de enmascaramiento de datos crea un número enmascarado con una suma de comprobación válida.

## Enmascaramiento de la dirección de correo electrónico

Utilice la transformación de enmascaramiento de datos para enmascarar la dirección de correo electrónico que contiene el valor de cadena. La transformación de enmascaramiento de datos puede enmascarar una dirección de correo electrónico con caracteres ASCII aleatorios o reemplazar la dirección de correo electrónico con una dirección de correo electrónico realista.

Puede aplicar los siguientes tipos de enmascaramiento con la dirección de correo electrónico:

### **Enmascaramiento de correo electrónico estándar**

La transformación de enmascaramiento de datos devuelve caracteres ASCII aleatorios cuando enmascara una dirección de correo electrónico. Por ejemplo, la transformación de enmascaramiento de datos puede enmascarar Georgesmith@yahoo.com como KtrlupQAPyk@vdSKh.BIC. El valor predeterminado es estándar.

### **Enmascaramiento de correo electrónico avanzado**

La transformación de enmascaramiento de datos enmascara la dirección de correo electrónico con otra dirección de correo electrónico realista derivada de los puertos de salida de la transformación o de las columnas del diccionario.

## Enmascaramiento de correo electrónico avanzado

Con el tipo de enmascaramiento de correo electrónico avanzado, puede enmascarar la dirección de correo electrónico con otra dirección de correo electrónico realista. La transformación de enmascaramiento de datos crea la dirección de correo electrónico a partir de las columnas del diccionario o de los puertos de salida de la transformación.

Puede crear la parte local en la dirección de correo electrónico a partir de los puertos de salida de asignación. O puede crear la parte local de la dirección de correo electrónico a partir de las columnas de la tabla relacional o del archivo sin formato.

La transformación de enmascaramiento de datos puede crear el nombre de dominio para la dirección de correo electrónico a partir de un valor constante o de un valor aleatorio en el diccionario del dominio.

Puede crear un enmascaramiento de correo electrónico avanzado basado en las siguientes opciones:

### **Dirección de correo electrónico basada en puertos dependientes**

Puede crear una dirección de correo electrónico basada en los puertos de salida de la transformación de enmascaramiento de datos. Seleccione los puertos de salida de la transformación para las columnas de nombre y de apellido. La transformación de enmascaramiento de datos enmascara el nombre, el apellido, o ambos en función de los valores que especifique para la longitud del nombre y el apellido.

### Dirección de correo electrónico basada en un diccionario

Puede crear una dirección de correo electrónico basada en las columnas de un diccionario. Seleccione una tabla de referencia como el origen para el diccionario.

Seleccione las columnas del diccionario para el nombre y el apellido. La transformación de enmascaramiento de datos enmascara el nombre, el apellido, o ambos en función de los valores que especifique para la longitud del nombre y el apellido.

## Parámetros de configuración para un tipo de enmascaramiento de direcciones de correo electrónico avanzado

Especifique los parámetros de configuración cuando configure el enmascaramiento de direcciones de correo electrónico avanzado.

Puede especificar los siguientes parámetros de configuración:

### Delimitador

Puede seleccionar un delimitador, tales como un punto, guión, o guión bajo, para separar el nombre y el apellido en la dirección de correo electrónico. Si no desea separar el nombre y el apellido en la dirección de correo electrónico, deje en blanco el delimitador.

### Columna FirstName

Seleccione un puerto de salida de transformación de enmascaramiento de datos o una columna de diccionario para enmascarar el nombre en la dirección de correo electrónico.

### Columna LastName

Seleccione un puerto de salida de transformación de enmascaramiento de datos o una columna de diccionario para enmascarar el apellido en la dirección de correo electrónico.

### Longitud de las columnas FirstName o LastName

Restringe el número de caracteres para enmascarar las columnas de nombre y apellido. Por ejemplo, los datos de entrada son Timothy para el nombre y Smith para el apellido. Seleccione 5 como la longitud de la columna de nombre. Seleccione 1 como la longitud de la columna de apellido con un punto como delimitador. La transformación de enmascaramiento de datos genera la siguiente dirección de correo electrónico:

```
timot.s@<nombre_de_dominio>
```

### DomainName

Puede utilizar un valor constante, como gmail.com, para el nombre de dominio. También puede especificar otro archivo de diccionario que contenga una lista de nombres de dominio. El diccionario de dominio puede ser un archivo sin formato o una tabla relacional.

## Enmascaramiento de direcciones IP

La transformación de enmascaramiento de datos enmascara una dirección IP como otra dirección IP dividiéndola en cuatro números, separados por un punto. El primer número es la red. La transformación de enmascaramiento de datos enmascara el número de la red dentro de un intervalo de red.

La transformación de enmascaramiento de datos enmascara una dirección IP de Clase A como una dirección IP de Clase A y una dirección 10.x.x.x como una dirección 10.x.x.x. La transformación de enmascaramiento de datos no enmascara la dirección de red privada y de clase. Por ejemplo, la transformación de enmascaramiento de datos puede enmascarar 11.12.23.34 como 75.32.42.52 y 10.23.24.32 como 10.61.74.84.

**Nota:** Cuando enmascara muchas direcciones IP, la transformación de enmascaramiento de datos puede devolver valores no exclusivos porque no enmascara la clase o red privada de las direcciones IP.

## Enmascaramiento del número de teléfono

La transformación de enmascaramiento de datos enmascara un número de teléfono sin cambiar el formato del número de teléfono original. Por ejemplo, la transformación de enmascaramiento de datos puede enmascarar el número (607)382 0658 como (408)256 3106.

Los datos de origen pueden contener números, espacios, guiones y paréntesis. El servicio de integración no enmascara caracteres alfabéticos o especiales.

La transformación de enmascaramiento de datos puede enmascarar datos de cadenas, enteros y bigint.

## Enmascaramiento del número de seguridad social

La transformación de enmascaramiento de datos genera un número de la seguridad social que no es válido según la lista de grupo alto de la administración de la seguridad social. La lista de grupo alto contiene números válidos que ha emitido la Administración de la Seguridad Social.

La lista de grupo alto predeterminada es un archivo de texto en la ubicación siguiente:

```
<Installation Directory>\infa_shared\SrcFiles\highgroup.txt
```

Para utilizar la el archivo de la lista de grupo alto en los flujos de trabajo, copie el archivo de texto en el directorio de origen que configure para el servicio de integración de datos.

La transformación de enmascaramiento de datos genera números de SSN que no están en la lista de grupo alto. La Administración de la Seguridad Social actualiza la lista de grupo alto cada mes. Descargue la última versión de la lista en la siguiente ubicación:

```
http://www.socialsecurity.gov/employer/ssns/highgroup.txt
```

## Formato de números de la seguridad social

La transformación de enmascaramiento de datos acepta cualquier formato de SSN que contenga nueve dígitos. Los dígitos se pueden limitar a cualquier conjunto de caracteres. Por ejemplo, la transformación de enmascaramiento de datos acepta el siguiente formato: +=54-\*9944\$#789-,\*()".

## Requisito de código de área

La transformación de enmascaramiento de datos devuelve un número de la seguridad social que no es válido con el mismo formato que el origen. Los primeros tres dígitos del SSN indican el código de área. La transformación de enmascaramiento de datos no enmascara el código de área. Sí lo hace con el número de grupo y el número de serie. El SSN de origen debe contener un código de área válido. La transformación de enmascaramiento de datos posiciona el código de área en la lista de grupos altos y establece un intervalo de números sin utilizar que se pueden aplicar como datos enmascarados. Si el SSN no es válido, la transformación de enmascaramiento de datos no enmascarará los datos de origen.

## Enmascaramiento de números de la seguridad social repetibles

La transformación de enmascaramiento de datos devolverá números de seguridad social deterministas con enmascaramiento repetible. La transformación de enmascaramiento de datos no puede devolver todos los números de seguridad social exclusivos porque es incapaz de devolver los números de seguridad social válidos que la administración de la seguridad social ha emitido.

## Enmascaramiento de direcciones URL

La transformación de enmascaramiento de datos analiza una URL buscando la cadena "://" y analizando la subcadena situada a la derecha de la misma. La dirección URL de origen debe contener la cadena "://". La dirección URL de origen puede contener números y caracteres alfabéticos.

La transformación de enmascaramiento de datos no enmascara el protocolo de la dirección URL. Por ejemplo, si la dirección URL es `http://www.yahoo.com`, la transformación de enmascaramiento de datos puede devolver `http://MgLaHjCa.VsD/`. La transformación de enmascaramiento de datos puede generar una URL que no es válida.

**Nota:** La transformación de enmascaramiento de datos siempre devuelve caracteres ASCII para una dirección URL.

## Enmascaramiento del número de seguro social

La transformación de enmascaramiento de datos enmascara un número de la seguridad social que tiene nueve dígitos. Los dígitos se pueden limitar a cualquier conjunto de caracteres.

Si el número no contiene delimitadores, el número enmascarado no contiene delimitadores tampoco. En caso contrario, el número enmascarado tiene el siguiente formato:

xxx-xxx-xxx

### Números de SIN repetibles

Puede configurar la transformación de enmascaramiento de datos para que devuelva valores de SIN repetibles. Si configura un puerto para el enmascaramiento de SIN repetible, la transformación de enmascaramiento de datos devolverá datos enmascarados deterministas cada vez que el valor de SIN de origen y el valor de inicialización sean iguales.

Para obtener números de SIN repetibles, habilite **Valores repetibles** e introduzca un número de semilla. La transformación de enmascaramiento de datos devuelve valores únicos para cada SIN.

### Dígito de inicio de SIN

Puede definir el primer dígito del SIN enmascarado.

Habilite **Dígito de inicio** e introduzca el dígito. La transformación de enmascaramiento de datos crea números de SIN enmascarados que empiezan por el número que introduzca.

## Archivo de valores predeterminados

Cuando el formato de datos de origen o el tipo de datos no es válido para una máscara, el servicio de integración de datos aplica una máscara predeterminada a los datos. El servicio de integración aplica valores enmascarados desde el archivo de valores predeterminados. Puede editar el archivo de valores predeterminados y modificar estos valores predeterminados.

El archivo de valores predeterminados es un archivo XML que se encuentra en la siguiente ubicación:

```
<Directorio de instalación>\infa_shared\SrcFiles\defaultValue.xml
```

Para utilizar el archivo de valores predeterminados en los flujos de trabajo, copie el archivo de valores predeterminados en el directorio de origen que haya configurado para el servicio de integración de datos.

El archivo defaultvalue.xml contiene los siguientes pares nombre-valor:

```
<?xml version="1.0" standalone="yes" ?>
<defaultValue
  default_char = "X"
  default_digit = "9"
  default_date = "11/11/1111 00:00:00"
  default_email = "abc@xyz.com"
  default_ip = "99.99.9.999"
  default_url = "http://www.xyz.com"
  default_phone = "999 999 999 9999"
  default_ssn = "999-99-9999"
  default_cc = "9999 9999 9999 9999"
  default_sin = "999-999-999"
  default_seed = "500"/>
```

## TEMAS RELACIONADOS

- [“Configure el servicio de integración de datos” en la página 137](#)

# Configuración de la transformación de enmascaramiento de datos

Lleve a cabo los pasos siguientes para configurar la transformación de enmascaramiento de datos.

1. Configure las opciones de ejecución para el servicio de integración de datos.
2. Cree la transformación.
3. Defina los puertos de entrada.
4. Configure las reglas de enmascaramiento de cada puerto que desee cambiar.
5. Obtenga una vista previa de los datos para comprobar los resultados.

## Configure el servicio de integración de datos

Puede configurar opciones de ejecución para el servicio de integración de datos en Informatica Administrator (Administrator Tool).

Configure las opciones de ejecución para establecer los siguientes directorios predeterminados:

- Directorio de inicio. Contiene el directorio de origen y el directorio de la memoria caché.
- Directorio de origen. Contiene los archivos de origen para los flujos de trabajo. Por ejemplo, el directorio de origen puede contener los archivos highgrp.txt y defaultvalue.xml.
- Directorio de la memoria caché. Contiene los archivos de memoria caché para la máscara de sustitución.

Para definir valores para las opciones de ejecución, abra Administrator Tool y seleccione el servicio de integración de datos en el **Navegador de dominio**. Haga clic en **Procesos** en la barra de herramientas para ver el nodo **Opciones de ejecución**.

## TEMAS RELACIONADOS

- [“Archivo de valores predeterminados” en la página 136](#)

## Cómo definir los puertos

Añada los puertos de entrada de enmascaramiento de datos en la vista **Resumen**. Cuando se crea un puerto de entrada, Developer Tool crea un puerto de salida correspondiente de forma predeterminada. El puerto de salida tiene el mismo nombre que el puerto de entrada.

1. En la vista **Resumen**, haga clic en **Nuevo** para añadir un puerto.
2. Configure el tipo de datos, la precisión y la escala para la columna.  
Tiene que configurar el tipo de datos de la columna antes de definir las reglas de enmascaramiento para la columna.
3. Para configurar el enmascaramiento de datos para el puerto, haga clic en la flecha de la columna de tipo de enmascaramiento en la vista **Resumen**.

## Cómo configurar un enmascaramiento de datos para cada puerto

Seleccione una técnica de enmascaramiento y las reglas de enmascaramiento correspondientes para un puerto en el cuadro de diálogo Enmascaramiento de datos. El cuadro de diálogo Enmascaramiento de datos se muestra al hacer clic en la columna de enmascaramiento de datos de la ficha **Puertos**.

1. Habilite **Aplicar enmascaramiento** para configurar el enmascaramiento para el puerto seleccionado.  
Developer Tool muestra una lista de técnicas de enmascaramiento que puede utilizar en función de los tipos de datos del puerto que vaya a enmascarar.
2. Seleccione una técnica de enmascaramiento de la lista.  
Developer Tool muestra diferentes reglas de enmascaramiento en función de la técnica de enmascaramiento que seleccione. Algunos formatos de máscara especiales no tienen reglas de enmascaramiento que se puedan configurar.
3. Configure las reglas de enmascaramiento.
4. Haga clic en **Aceptar** para aplicar la configuración de enmascaramiento de datos al puerto.  
Cuando se define un enmascaramiento de datos para un puerto, Developer Tool crea un puerto de salida que se llama **out-<nombre del puerto>**. El **<nombre del puerto>** es el mismo que el del puerto de entrada. La transformación de enmascaramiento de datos devuelve los datos enmascarados en el puerto **out-<nombre del puerto>**.

## Vista previa de los datos enmascarados

Puede comparar los datos enmascarados con los datos originales si visualiza los resultados de la transformación de enmascaramiento de datos en el **Visor de datos**.

1. Después de configurar los puertos de transformación de enmascaramiento de datos, cree una asignación que incluya el origen del objeto de datos físicos y la transformación de enmascaramiento de datos.
2. Conecte el origen a la transformación de enmascaramiento de datos.
3. Compruebe que el origen tiene datos en una ubicación compartida a la que pueda acceder el servicio de integración de datos.
4. Haga clic en la transformación de enmascaramiento de datos para seleccionarla en la asignación.
5. Haga clic en **Visor de datos** y en **Ejecutar**.  
Developer Tool muestra datos de todos los puertos de salida de transformación de enmascaramiento de datos. Los puertos que tienen el prefijo de **salida** contienen datos enmascarados. Puede comparar los datos enmascarados con los datos originales en la vista **Visor de datos**.

## Cómo crear una transformación de enmascaramiento de datos

Cree una transformación de enmascaramiento de datos en Developer Tool.

Antes de crear la transformación de enmascaramiento de datos, cree el origen. Importe un archivo sin formato o una tabla de base de datos relacional como objeto de datos físicos.

1. Seleccione un proyecto o una carpeta en la vista **Explorador de objetos**.
2. Haga clic en **Archivo > Nueva > Transformación**.  
A continuación, aparece el cuadro de diálogo **Nueva**.
3. Seleccione la transformación de enmascaramiento de datos.
4. Haga clic en **Siguiente**.
5. Especifique un nombre para la transformación.
6. Haga clic en **Finalizar**.

La transformación aparece en el editor.

## Propiedades de la sesión de transformación de enmascaramiento de datos

Puede configurar las propiedades de la sesión de transformación de enmascaramiento de datos para aumentar el rendimiento.

Configure las siguientes propiedades de sesión:

### Tamaño de la memoria caché

El tamaño de la memoria caché del diccionario en la memoria principal. Aumente el tamaño de memoria para mejorar el rendimiento. El tamaño mínimo recomendado es 32 MB para 100.000 registros. El valor predeterminado es 8 MB.

### Directorio de la memoria caché

La ubicación de la memoria caché del diccionario. Debe tener permisos de escritura para el directorio. El valor predeterminado es \$PMCacheDir.

### Tabla de almacenamiento compartido

Habilita la compartición de la tabla de almacenamiento entre las instancias de transformación de enmascaramiento de datos. Habilite la tabla de almacenamiento compartido cuando dos instancias de transformación de enmascaramiento de datos usen la misma columna del diccionario para la conexión de base de datos, valor de semilla y configuración regional. También puede habilitar la tabla de almacenamiento compartido cuando dos puertos en la misma transformación de enmascaramiento de datos usen la misma columna del diccionario para la conexión, la semilla y la configuración regional. Deshabilite la tabla de almacenamiento compartido cuando los puertos o las transformaciones de enmascaramiento de datos no compartan la columna del diccionario. El valor predeterminado es deshabilitada.

### Intervalo de confirmación de almacenamiento

El número de filas para confirmar al mismo tiempo para la tabla de almacenamiento. Aumente el valor para aumentar el rendimiento. Configure el intervalo de confirmación si no configura la tabla de almacenamiento compartido. El valor predeterminado es 100.000.

### Almacenamiento cifrado

Cifra las tablas de almacenamiento, tales como `IDM_SUBSTITUTION_STORAGE` y `IDM_EXPRESSION_STORAGE`. Compruebe que dispone de datos cifrados en las tablas de almacenamiento antes de habilitar la propiedad de almacenamiento cifrado. Borre esta opción si no desea cifrar las tablas de almacenamiento. El valor predeterminado es deshabilitado.

### Clave de cifrado de almacenamiento

La transformación de enmascaramiento de datos cifra el almacenamiento en función de la clave de cifrado de almacenamiento. Utilice la misma clave de cifrado para cada ejecución de sesión de la misma instancia de transformación de enmascaramiento de datos.

### Nombre del propietario del diccionario de sustitución

Nombre del propietario de la tabla del diccionario de sustitución cuando se selecciona el tipo de máscara de sustitución. Si el usuario de la base de datos especificado en la conexión de la base de datos no es el propietario de la tabla del diccionario de sustitución en una sesión, debe especificar el propietario de la tabla.

### Nombre del propietario del almacenamiento

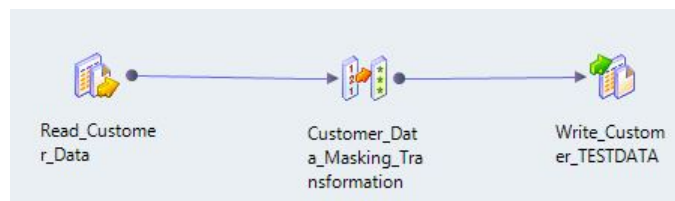
Nombre del propietario de tabla para `IDM_SUBSTITUTION_STORAGE` o `IDM_EXPRESSION_STORAGE` cuando selecciona el tipo de máscara de sustitución repetible única o de expresión repetible.

## Ejemplo de enmascaramiento de datos

El desarrollador tiene que crear datos de prueba para las aplicaciones del cliente. Los datos deben contener datos del cliente que sean realistas y a los que otros desarrolladores puedan acceder en el entorno de desarrollo de la compañía.

El desarrollador crea un servicio de datos que devuelve datos de cliente enmascarados como el ID, el número de tarjeta de crédito y los ingresos del cliente. La asignación incluye una transformación de enmascaramiento de datos que transforma los datos del cliente.

La siguiente figura muestra la asignación:



La asignación tiene las siguientes transformaciones:

- `Read_Customer_Data`. Contiene la información de la tarjeta de crédito y los ingresos del cliente.
- Transformación de `Customer_Data_Masking`. Enmascara todas las columnas excepto `FirstName` y `LastName`. La transformación de enmascaramiento de datos pasa las columnas enmascaradas al destino.
- `Customer_TestData`. La transformación de salida que recibe los datos enmascarados del cliente.

## Read\_Customer Data

Los datos de cliente contienen las siguientes columnas:

Column	Tipo de datos
CustomerID	Entero
LastName	Cadena
FirstName	Cadena
CreditCard	Cadena
Income	Entero
Join_Date	Fecha y hora (DD/MM/AAAA)

En la siguiente tabla aparecen datos de cliente de ejemplo.

CustomerID	LastName	FirstName	CreditCard	Income	JoinDate
0095	Bergeron	Barbara	4539-1686-3069-3957	12000	31/12/1999
0102	Brosseau	Derrick	5545-4091-5232-8948	4000	03/03/2011
0105	Anderson	Lauren	1234-5678-9012-3456	5000	04/03/2009
0106	Boonstra	Pauline	4217-9981-5613-6588	2000	07/07/2007
0107	Chan	Brian	4533-3156-8865-3156	4500	06/18/1995

## Transformación de enmascaramiento de datos de clientes

La transformación de enmascaramiento de datos enmascara todas las columnas de la fila del cliente excepto el nombre y el apellido.

La transformación de enmascaramiento de datos realiza los siguientes tipos de enmascaramiento:

- Enmascaramiento de claves
- Enmascaramiento aleatorio
- Enmascaramiento de tarjeta de crédito

En la tabla siguiente se muestran las reglas de enmascaramiento para cada puerto en la transformación de enmascaramiento de datos:

Puerto de entrada	Tipo de enmascaramiento	Reglas de enmascaramiento	Descripción
CustomerID	De claves	La semilla es 934. El ID de cliente no tiene ningún formato de máscara. Los caracteres de reemplazo de la cadena de resultado son 1234567890.	La máscara CustomerID es determinista. El ID del cliente enmascarado contiene números.
LastName	Sin máscara	-	-

Puerto de entrada	Tipo de enmascaramiento	Reglas de enmascaramiento	Descripción
FirstName	Sin máscara	-	-
CreditCard	CreditCard	-	La transformación de enmascaramiento de datos enmascara el número de la tarjeta de crédito con otro número que tiene una suma de comprobación válida.
Income	Aleatoria	Desenfoco Porcentaje Límite inferior = 1 Límite superior = 10	El ingreso enmascarado se encuentra dentro del diez por ciento del ingreso de origen.
JoinDate	Aleatoria	Desenfoco Unidad = Año Límite inferior = 5 Límite superior = 5	La fecha enmascarada se encuentra en el periodo de 5 años posterior a la fecha original.

## Resultados de datos de prueba de cliente

La transformación Customer\_TestData recibe datos de clientes realistas provenientes de la transformación de enmascaramiento de datos.

El destino Customer\_TestData recibe los siguientes datos:

out-CustomerID	out-LastName	outFirstName	out-CreditCard	out-Income	out-JoinDate
3954	Bergeron	Barbara	4539-1625-5074-4106	11500	03/22/2001
3962	Brosseau	Derrick	5545-4042-8767-5974	4300	04/17/2007
3964	Anderson	Lauren	1234-5687-2487-9053	5433	09/13/2006
3965	Boonstra	Pauline	4217-9935-7437-4879	1820	02/03/2010
3966	Chan	Brian	4533-3143-4061-8001	4811	10/30/2000

El ingreso está dentro del diez por ciento del ingreso original. La fecha de unión se encuentra en el periodo de 5 años posterior a la fecha original.

## Propiedades avanzadas de Transformación de enmascaramiento de datos

Configure propiedades que ayudan a determinar cómo el servicio de integración de datos procesa los datos para la Transformación de enmascaramiento de datos.

Puede configurar niveles de seguimiento de registros.

Configure la siguiente propiedad en la ficha **Avanzadas**:

**Nivel de seguimiento**

Cantidad de detalles que aparecen en el registro de esta transformación. Puede elegir simplificado, normal, inicialización detallada o datos detallados. El valor predeterminado es Normal.

## CAPÍTULO 11

# Transformación de decisión

Este capítulo incluye los siguientes temas:

- [Resumen de la transformación de decisión, 144](#)
- [Funciones de transformación de decisión, 145](#)
- [Instrucciones condicionales de transformación de decisión, 146](#)
- [Operadores de transformación de decisión, 147](#)
- [Control de valores NULL en transformaciones de decisión, 148](#)
- [Cómo configurar una estrategia de decisión, 148](#)
- [Propiedades avanzadas de Transformación de decisión, 149](#)

## Resumen de la transformación de decisión

La transformación de decisión es una transformación pasiva que evalúa condiciones en datos de entrada y crea una salida basada en los resultados de esas condiciones.

Configure una transformación de decisión para generar valores diferentes según los valores que se encuentran en los campos de entrada. Por ejemplo, si los ingresos de un cliente exceden un importe específico, puede añadir la cadena "Prioridad" al nombre del cliente.

Puede añadir varias estrategias de decisión a la transformación de decisión. Cada estrategia evalúa una instrucción condicional `IF-THEN-ELSE`. Dentro de esta instrucción, puede utilizar condiciones `ELSEIF` o anidar instrucciones `IF-THEN-ELSE` adicionales.

La transformación de decisión es similar a la transformación de expresión en cuanto que permite utilizar instrucciones condicionales y funciones para comprobar los datos de origen. Sin embargo, la transformación de decisión difiere de la transformación de expresión en los siguientes aspectos:

- La transformación de decisión utiliza instrucciones `IF-THEN-ELSE` para evaluar condiciones. La transformación de expresión utiliza instrucciones `IIF`.
- La transformación de decisión contiene funciones que no están disponibles en la transformación de expresión.
- Cada estrategia de decisión puede generar varias salidas.

# Funciones de transformación de decisión

La transformación de decisión proporciona acceso a funciones predefinidas que se utilizan para definir estrategias de decisión.

El editor de transformaciones de decisión contiene una carpeta Decision. Esta carpeta contiene funciones específicas a la transformación de decisión. Asimismo, el editor contiene otras carpetas que proporcionan acceso a funciones de transformación de expresión.

Cuando se hace clic en una función del editor de expresiones, la transformación muestra el uso y los tipos de datos de la función, además de una descripción de lo que hace la función.

**Nota:** No todas las funciones de una transformación de expresión son compatibles con la transformación de decisión. La transformación de decisión solamente proporciona acceso a funciones compatibles de una transformación de expresión.

## Lista de funciones de una transformación de decisión

- ADD\_TO\_DATE
- ASCII
- CEIL
- CHOOSE
- CONCAT
- CONVERT\_BASE
- COS
- COSH
- CRC32
- CUME
- DATE\_DIFF
- EXP
- FLOOR
- FV
- GET\_DATE\_PART
- GREATEST
- IN
- INDEXOF
- INITCAP
- ISNULL
- LAST\_DAY
- LEAST
- LN
- LOG
- LOWER
- LPAD
- MAKE\_DATE\_TIME
- MD5

- METAPHONE
- MOD
- MOVINGAVG
- MOVINGSUM
- NPER
- PMT
- PV
- RAND
- RATE
- REG\_EXTRACT
- REG\_MATCH
- REG\_REPLACE
- REPLACECHR
- REPLACESTR
- REVERSE
- ROUND
- RPAD
- SET\_DATE\_PART
- SIGN
- SIN
- SINH
- SOUNDEX
- SQRT
- TAN
- TANH
- TO\_DATE
- TRUNC
- UPPER

## Instrucciones condicionales de transformación de decisión

La transformación de decisión utiliza instrucciones condicionales IF-THEN-ELSE para evaluar datos de entrada.

Dentro de estas instrucciones condicionales se pueden usar condiciones ELSEIF o anidar instrucciones IF-THEN-ELSE adicionales. Las instrucciones condicionales de una transformación de decisión utilizan el siguiente formato:

```
// Primary condition
IF <Boolean expression>
THEN <Rule Block>
```

```
// Optional - Multiple ELSEIF conditions
ELSEIF <Boolean expression>
THEN <Rule Block>
// Optional ELSE condition
ELSE <Rule Block>
ENDIF
```

Se pueden anidar instrucciones condicionales adicionales dentro de un bloque de reglas.

## Operadores de transformación de decisión

Utilice operadores de transformación de decisión para definir estrategias de decisión.

En la siguiente tabla se describen los operadores de la transformación de decisión:

Tipo de operador	Operador	Descripción
Asignación	:=	Asigna un valor a un puerto.
Booleana	AND	Añade una condición lógica requerida. Para que la expresión booleana superior sea verdadera, todas las condiciones lógicas vinculadas mediante este operador deben ser verdaderas.
Booleana	OR	Añade una condición lógica. Para que la expresión booleana superior sea verdadera, por lo menos una condición lógica vinculada mediante este operador debe ser verdadera.
Booleana	NOT	Añade una condición lógica negativa. Para que la expresión booleana superior sea verdadera, la condición negativa especificada mediante este operador debe ser verdadera.
Decisión	=	Comprueba si los elementos comparados son iguales. Utilícese con tipos de datos de cadena o numéricos.
Decisión	<>	Comprueba si los elementos comparados no son iguales. Utilícese con tipos de datos de cadena o numéricos.
Decisión	<	Comprueba si un valor es menor que otro valor. Utilícese con tipos de datos numéricos.
Decisión	<=	Comprueba si un valor es menor o igual que otro valor. Utilícese con tipos de datos numéricos.
Decisión	>	Comprueba si un valor es mayor que otro valor. Utilícese con tipos de datos numéricos.
Decisión	>=	Comprueba si un valor es mayor o igual que otro valor. Utilícese con tipos de datos numéricos.
Númérica	-	Resta
Númérica	NEG	Negación
Númérica	+	Suma

Tipo de operador	Operador	Descripción
Númerica	*	Multiplicación
Númerica	/	División
Númerica	%	Módulo. Devuelve el resto resultante de dividir un número entre otro.
Cadena		Concatena cadenas.

## Control de valores NULL en transformaciones de decisión

El control de valores NULL determina cómo el servicio de integración de datos procesa datos de valores nulos en una transformación de decisión.

Cuando se habilita el control de valores NULL, la transformación de decisión conserva el formato de datos de entrada NULL. La transformación evalúa las funciones utilizando el valor de entrada NULL.

Al deshabilitar el control de valores NULL, la transformación de decisión asigna un valor predeterminado a los datos de entrada NULL. La transformación evalúa las funciones utilizando el valor predeterminado. Por ejemplo, si un campo de entrada de tipo entero tiene un valor nulo, la transformación de decisión asigna un valor de 0 a la entrada y evalúa las funciones utilizando un valor de entrada de 0.

De forma predeterminada, el control de valores NULL no está habilitado en la transformación de decisión. El control de valores NULL se habilita en la ficha **Estrategias**. Puede habilitar el control de valores NULL tras configurar una estrategia para la transformación.

## Cómo configurar una estrategia de decisión

Para configurar una estrategia de decisión, conecte los datos de origen a la transformación de decisión y edite las propiedades en las vistas de la transformación.

1. Abra una transformación de decisión.
2. Compruebe que la transformación contiene puertos de entrada y de salida.
3. Seleccione la vista **Decisión**.
4. Haga clic en **Añadir**.
5. Especifique un nombre para la estrategia.
6. En el área **Expresión**, introduzca una instrucción condicional **IF-THEN-ELSE**.
7. Para añadir una función a la expresión, explore las funciones que aparecen en la ficha **Funciones** y haga doble clic en un nombre de función.

**Sugerencia:** Para introducir una función rápidamente, escriba las primeras letras del nombre de la función y seleccione **CTRL-Espacio**.

8. Para añadir un puerto a la expresión, explore los puertos que aparecen en la ficha **Puertos**. Haga doble clic en un nombre de puerto para añadirlo a la expresión. Si lo desea, haga clic en **Editar puertos de salida** para editar los ajustes de los puertos de salida o para añadir puertos de salida.
9. Si lo desea, añada líneas de comentario escribiendo "/" seguido de los comentarios.
10. Haga clic en **Validar** para determinar si la expresión de decisión es válida.
11. Haga clic en **Aceptar** para guardar la estrategia.
12. Si lo desea, añada estrategias adicionales. Cada estrategia debe utilizar puertos de salida únicos. Las estrategias no pueden compartir puertos de salida.

## Propiedades avanzadas de Transformación de decisión

Configure propiedades que ayudan a determinar cómo el servicio de integración de datos procesa los datos para la Transformación de decisión.

Puede configurar niveles de seguimiento de registros.

Configure la siguiente propiedad en la ficha **Avanzadas**:

### Nivel de seguimiento

Cantidad de detalles que aparecen en el registro de esta transformación. Puede elegir simplificado, normal, inicialización detallada o datos detallados. El valor predeterminado es Normal.

## CAPÍTULO 12

# Transformación de excepción de registros duplicados

Este capítulo incluye los siguientes temas:

- [Resumen de la transformación de excepción de registros duplicados, 150](#)
- [Flujo del proceso de la excepción de registros duplicados, 151](#)
- [Excepciones de registros duplicados, 151](#)
- [Vista de configuración de excepción de registros duplicados , 152](#)
- [Puertos, 154](#)
- [Propiedades avanzadas de la transformación de excepción de registros duplicados, 156](#)
- [Ejemplo de asignación de excepción de registros duplicados, 157](#)
- [Cómo crear una transformación de excepción de registros duplicados, 163](#)

## Resumen de la transformación de excepción de registros duplicados

La transformación de excepción de registros duplicados es una transformación activa que lee la salida de los procesos de calidad de datos e identifica registros duplicados que requieren la revisión manual.

La transformación de excepción de registros duplicados recibe la entrada de otra transformación o de un objeto de datos en otra asignación. La entrada a la transformación de excepción debe contener un valor numérico de puntuación de coincidencia que la transformación pueda utilizar para determinar si el registro es un duplicado. Establezca un umbral de puntuación de coincidencia superior e inferior en la transformación de excepción de registros duplicados.

La transformación de excepción de registros duplicados realiza una de las siguientes acciones:

- Si la puntuación de coincidencia es mayor o igual al umbral superior, la transformación trata el registro como un duplicado y lo escribe en un destino de base de datos.
- Si la puntuación de coincidencia es menor que el umbral superior y mayor que el umbral inferior de la transformación trata el registro como un posible duplicado y lo escribe en el registro en un destino diferente para la revisión manual. Si el registro pertenece a un clúster, la transformación escribe todos los registros del clúster en el destino.
- Cuando un clúster tiene una puntuación de coincidencia menor que el umbral inferior, todos los registros del clúster van al grupo de salida de registros únicos. Los clústeres de tamaño 1 se distribuyen al grupo

único, independientemente de la puntuación de coincidencia. De forma predeterminada, la transformación de excepción no escribe registros únicos en un destino. Puede configurar la transformación para devolver los registros únicos.

- Si una puntuación de coincidencia de un clúster no está en el intervalo de 0 - 100, la transformación de excepción ignora todas las filas del clúster. El servicio de integración de datos registra un mensaje que incluye el clusterID.

## Flujo del proceso de la excepción de registros duplicados

La transformación de excepción de registros duplicados analiza la salida de otras transformaciones de calidad de datos y crea las tablas que contienen los registros con distintos niveles de calidad de datos.

Puede configurar las transformaciones de calidad de datos en una única asignación individual o puede crear asignaciones en distintas etapas del proceso.

Puede usar Informatica Data Director para Data Quality para revisar y actualizar los registros duplicados que requieren revisión manual.

Utilice Developer Tool para completar las siguientes tareas:

1. Cree una asignación que genera valores de puntuación para problemas de calidad de datos.
2. Utilice una transformación de coincidencia en modo de clústeres para generar valores de puntuación para excepciones de registros duplicados.
3. Configure la transformación de excepción de registros duplicados para leer la salida de transformación de coincidencia. Configure la transformación para escribir los registros en las tablas de base de datos según los valores de puntuación de coincidencia de los registros.
4. Configure los objetos de datos de destino para los registros de consolidación automática.
5. Haga clic en la opción **Generar tabla de registros duplicados** para crear la tabla de registros duplicados y añádala al lienzo de la asignación.
6. Añada la asignación a un flujo de trabajo.
7. Configure una tarea humana para asignar la revisión manual de posibles registros duplicados a los usuarios. Los usuarios pueden revisar y actualizar los registros en Data Director para Data Quality.

## Excepciones de registros duplicados

Puede utilizar una transformación de excepción de registros duplicados para identificar clústeres de datos duplicados que necesitan revisión manual. Las puntuaciones de coincidencia de registros en clústeres determina los posibles duplicados. Puede configurar los umbrales superior e inferior para las puntuaciones de coincidencia de la transformación. El los umbrales superior e inferior defina el grado de similitud.

Un clúster contiene registros relacionados que una operación de coincidencia agrupa simultáneamente. La transformación de coincidencia crea clústeres con la operación de análisis de duplicados y la operación de resolución de identidad. Cada registro de un clúster tiene el mismo ID de clúster. Cuando la puntuación de coincidencia menor de un clúster está entre los umbrales superior e inferior, la transformación de excepción de registros duplicados identifica el clúster como un clúster de excepción de registros duplicados. La

transformación de coincidencia añade una columna de valor de ID de clúster todos los registros. Los registros duplicados reciben el mismo ID de clúster.

La puntuación de registro más baja determina el tipo de clúster. Un clúster podría tener 11 registros que tienen una puntuación de coincidencia de 0,95 y un registro con la puntuación de coincidencia de 0,79. Si el umbral superior es 0,9 y el umbral inferior es 0,8, la transformación de excepción escribe los registros en la tabla de registros únicos.

## Vista de configuración de excepción de registros duplicados

Defina los umbrales de puntuación de coincidencia y configure dónde la transformación de excepción de registros duplicados escribirá los diferentes tipos de datos de salida.

La siguiente figura muestra las propiedades que puede configurar:

The screenshot shows a configuration window with two main sections. The first section, 'Umbral de revisión manual', contains two input fields: 'Umbral inferior' with a value of 0,80 and 'Umbral superior' with a value of 0,95. The second section, 'Opciones de distribución de datos', contains a table with three columns: 'Tipo', 'Salida estándar', and 'Tabla de registro duplicado'. The table has three rows: 'Consolidación automática (sobre el umbral superior)' with a checked box in 'Salida estándar' and an unchecked box in 'Tabla de registro duplicado'; 'Consolidación manual (entre umbrales)' with unchecked boxes in both columns; and 'Registros únicos (por debajo del umbral inferior)' with an unchecked box in 'Salida estándar' and a checked box in 'Tabla de registro duplicado'. Below the table is a checkbox labeled 'Crear un grupo de salida aparte para los registros únicos' which is unchecked, and a button labeled 'Generar tabla de registros duplicados'.

Tipo	Salida estándar	Tabla de registro duplicado
Consolidación automática (sobre el umbral superior)	<input checked="" type="checkbox"/>	<input type="checkbox"/>
Consolidación manual (entre umbrales)	<input type="checkbox"/>	<input type="checkbox"/>
Registros únicos (por debajo del umbral inferior)	<input type="checkbox"/>	<input checked="" type="checkbox"/>

☐ Crear un grupo de salida aparte para los registros únicos

Generar tabla de registros duplicados

Puede configurar las propiedades siguientes:

### Umbral inferior

El límite inferior para el intervalo de puntuaciones de registros duplicados. La transformación procesa los registros con puntuaciones de coincidencia inferiores a este valor como registros únicos. El valor del umbral inferior es un número de 0 a 1.

### Umbral superior

El límite superior para el intervalo de puntuaciones de registros duplicados. La transformación procesa los registros con puntuaciones de coincidencia mayores o iguales que el umbral superior como registros duplicados. El valor del umbral superior es un número mayor que el umbral inferior.

### Consolidación automática

Clústeres en los que todos los registros tienen puntuaciones de coincidencia mayores que el umbral superior. Los clústeres de consolidación automática no necesitan revisión. Los registros son duplicados. Puede utilizar la transformación de consolidación para combinar los registros. De forma predeterminada, la transformación de excepción de registros duplicados escribe clústeres de consolidación automática en puertos de salida estándar.

### Consolidación manual

Clústeres en los que todos los registros tienen puntuaciones de coincidencia mayores o iguales que el umbral inferior y al menos un registro tiene una puntuación de coincidencia menor que el umbral superior. Debe realizar una revisión manual de los clústeres para determinar si contienen registros genuinos duplicados. De forma predeterminada, la transformación de excepción de registros duplicados escribe registros de consolidación manual en la tabla de registros duplicados.

### Consolidación única

Clústeres con un tamaño de clúster igual a uno o clústeres en los que cualquier registro tiene una puntuación de coincidencia menor que el umbral inferior. Los clústeres de registro único no son duplicados. De forma predeterminada, la transformación de excepción de registros duplicados no escribe registros únicos en una tabla de salida.

### Salida estándar

Los tipos de registro que la transformación escribe en los puertos de salida estándar.

El valor predeterminado es registros de consolidación automática.

### Tabla de registros duplicados

Los tipos de registro que la transformación escribe en los puertos de salida de registros duplicados. El valor predeterminado es registros de consolidación manual.

### Crear un grupo de salida aparte para los registros únicos

Crea un grupo de salida aparte para los registros únicos. Si no crea una tabla aparte para los registros únicos, puede configurar la transformación para escribir los registros únicos en uno de los otros grupos. O bien, puede omitir la escritura de registros únicos en una tabla de salida. De forma predeterminada esta opción está deshabilitada.

### Generar tabla de registros duplicados

Crea un objeto de base de datos para contener los datos de clústeres de registros duplicados. Cuando se selecciona esta opción, Developer Tool crea el objeto de base de datos. Developer Tool añade el objeto en el repositorio de modelos, añade una instancia del objeto al lienzo de la asignación y vincula los puertos al objeto.

## Cómo generar una tabla de registros duplicados

Puede generar una tabla de registros duplicados desde una instancia de transformación de excepción de registros duplicados en una asignación.

1. En la vista **Configuración**, haga clic en **Generar tabla de registros duplicados** para generar la tabla. A continuación, aparece el cuadro de diálogo **Crear objeto de datos relacionales**.
2. Busque y seleccione una conexión a la base de datos donde guardar la tabla.
3. Especifique un nombre para la tabla de registros duplicados de la base de datos.
4. Especifique un nombre para el objeto de la tabla de registros duplicados del repositorio de modelos.
5. Haga clic en **Finalizar**.  
Developer Tool añade la nueva tabla al lienzo de la asignación.

# Puertos

La transformación de excepción de registros duplicados tiene varios grupos de puertos de entrada y salida.

La siguiente figura muestra un ejemplo de los puertos de entrada y salida:

Puertos								
	Nombre	Tipo	Precisión	Escala	Entrada	Salida	Predeterminado	Descripción
Entradas								
Datos (3)								
1	Employee	decimal	10	0	<input checked="" type="checkbox"/>	<input type="checkbox"/>		
2	Name	string	50	0	<input checked="" type="checkbox"/>	<input type="checkbox"/>		
3	Addr1	string	50	0	<input checked="" type="checkbox"/>	<input type="checkbox"/>		
Control (3)								
1	Puntuación	double	15	0	<input checked="" type="checkbox"/>	<input type="checkbox"/>		
2	Row_Identifier	string	25	0	<input checked="" type="checkbox"/>	<input type="checkbox"/>		
3	Cluster_ID	integer	10	0	<input checked="" type="checkbox"/>	<input type="checkbox"/>		
Salidas								
Salida estándar (6)								
1	Puntuación	double	15	0	<input type="checkbox"/>	<input checked="" type="checkbox"/>		
2	Row_Identifier	string	25	0	<input type="checkbox"/>	<input checked="" type="checkbox"/>		
3	Cluster_ID	integer	10	0	<input type="checkbox"/>	<input checked="" type="checkbox"/>		
4	Employee	decimal	10	0	<input type="checkbox"/>	<input checked="" type="checkbox"/>		
5	Name	string	50	0	<input type="checkbox"/>	<input checked="" type="checkbox"/>		
6	Addr1	string	50	0	<input type="checkbox"/>	<input checked="" type="checkbox"/>		
Datos del clúster (9)								
1	Row_Identifier	bigint	19	0	<input type="checkbox"/>	<input checked="" type="checkbox"/>		
2	Sequential_Cl...	bigint	19	0	<input type="checkbox"/>	<input checked="" type="checkbox"/>		
3	Cluster_ID	integer	10	0	<input type="checkbox"/>	<input checked="" type="checkbox"/>		

## Puertos de entrada de transformación de excepción de registros duplicados

Una transformación de excepción de registros duplicados tiene un grupo de datos y un grupo de control de puertos de entrada.

El grupo **Datos** contiene puertos definidos por el usuario que reciben los datos de origen.

Los puertos de **control** reciben los metadatos que la transformación de coincidencia añade a los datos de origen. La siguiente tabla describe los puertos de **control**:

Puerto	Descripción
Puntuación	Valor decimal entre 0 y 1. Identifica el grado de similitud con el registro que enlazaba el registro con el clúster.
Row_Identifier	Identificador único del registro.
Cluster_ID	Identificador del clúster de coincidencias al que pertenece el registro.

## Puertos de salida de transformación de excepción de registros duplicados

La transformación de excepción de registros duplicados tiene dos grupos de salida múltiple. De forma predeterminada, la transformación escribe registros duplicados para el grupo **Salida estándar**. La transformación escribe coincidencias potenciales en el grupo **Datos del clúster**. Se puede añadir un grupo de salida para registros únicos.

Puede cambiar los tipos de registro que la transformación escribe en los puertos de salida cambiando la configuración predeterminada en la vista **Configuración**.

En la siguiente tabla se describen los puertos de salida para el grupo **Salida estándar**:

Puerto	Descripción
Puntuación	Valor decimal entre 0 y 1. Identifica el grado de similitud entre el registro y otro registro en un clúster.
Row_Identifier	Identificador único del registro.
Cluster_ID	ID de clúster que la transformación de coincidencia asignado al registro.
Puertos definidos por el usuario	Los campos de datos de origen.

La siguiente tabla describe los puertos de salida del grupo **Datos del clúster**:

Puerto	Descripción
Row_Identifier	Identificador único del registro.
Sequential_Cluster_ID	Identifica el clúster en una tarea humana. Un flujo de trabajo utiliza el ID de clúster secuencial para asignar un clúster a una instancia de una tarea humana.
Cluster_ID	Identifica el clúster al que pertenece el registro. La transformación de coincidencia asigna un ID de clúster a todos los registros.
Puntuación	Valor decimal de 0 a 1. Identifica el grado de similitud con el registro que enlazaba el registro con el clúster.
Is_Master	Valor de la cadena que indica si el registro es el registro preferido del clúster. De forma predeterminada, la primera fila del clúster es el registro de localidad. El valor es Y o N.
Workflow_ID	ID que identifica el flujo de trabajo para el registro en una tarea. Cuando se ejecuta la asignación fuera de un flujo de trabajo, el ID de flujo de trabajo es DummyWorkflowID.
Puertos definidos por el usuario	Los puertos de datos de origen.

## Cómo crear puertos

Añada cada puerto de entrada al grupo de datos. Cuando se añade el puerto de entrada, Developer Tool añade un puerto de salida con el mismo nombre al grupo de salida estándar, al grupo de datos del clúster y al grupo de registros únicos.

1. Seleccione el grupo de entrada.  
El grupo se resalta.
2. Haga clic en **Nuevo (Insertar)**.  
Developer Tool añade un campo al grupo de datos, grupo de salida estándar, al grupo de datos del clúster y al grupo de registros únicos.
3. Cambie el nombre del campo según sea necesario.  
Developer Tool cambia el nombre de los otros grupos.
4. Introduzca el resto de los puertos que tenga que añadir al origen de datos.

## Propiedades avanzadas de la transformación de excepción de registros duplicados

La transformación de consolidación contiene propiedades avanzadas para la ordenación, la salida, la configuración de la memoria caché y el nivel de seguimiento.

Puede configurar las propiedades siguientes:

### Ordenar

Indica que la transformación de excepción de registros duplicados ordenará los registros por ID de clúster. Esta opción está habilitada de forma predeterminada. Deshabilite esta propiedad si la transformación de excepción de registros duplicados recibe los registros ordenados.

### Directorio de archivo caché

El servicio de integración de datos utiliza el directorio del archivo de memoria caché para crear archivos temporales cuando ordena datos. Después de que el servicio de integración de datos ordena los datos, elimina los archivos temporales. Puede configurar un directorio del equipo donde se ejecuta el servicio de integración de datos para usarlo como directorio del archivo de memoria caché. El valor predeterminado es el parámetro del sistema CacheDir.

### Tamaño de archivo caché

Determina la cantidad máxima de memoria necesaria cuando la transformación de registros duplicados realiza operaciones de ordenación. El servicio de integración de datos pasa todos los datos a la transformación de registros duplicados antes de realizar una operación de ordenación. El valor predeterminado es 400.000 bytes.

### Nivel de seguimiento

Cantidad de detalles que aparecen en el registro de esta transformación. Puede elegir simplificado, normal, inicialización detallada o datos detallados. El valor predeterminado es Normal.

# Ejemplo de asignación de excepción de registros duplicados

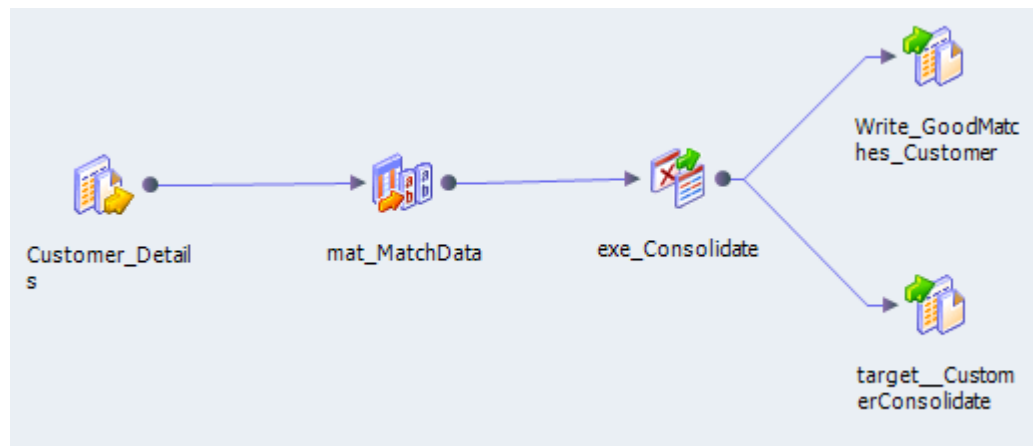
Una organización ejecuta un proyecto de datos para revisar los datos de clientes. La organización determina que los datos de clientes contienen varios registros duplicados. La organización necesita que se revisen manualmente algunos de los registros que pueden estar duplicados.

Cree una asignación de calidad de datos para identificar los registros de clientes duplicados. La asignación incluye la transformación de coincidencia. La transformación de excepción de registros duplicados recibe los resultados de la transformación de coincidencia. La transformación de excepción de registros duplicados escribe cada clúster del registro con un estado incierto en una tabla de base de datos. Un analista de datos revisa los datos contenidos en Informatica Data Director para Data Quality y determina qué registros son los registros duplicados.

## Asignación de excepción de registros duplicados

Configure una asignación de excepción de registros duplicados que examine los registros de los clientes y encuentra registros duplicados.

La siguiente figura muestra la asignación de excepción de registros duplicados:



La asignación contiene los siguientes objetos:

### **Customer\_Details**

El origen de datos que pueden contener los registros duplicados.

### **mat\_MatchData**

Una transformación de coincidencia que examina los datos de los clientes para determinar si los registros coinciden. La transformación de coincidencia crea una puntuación numérica que representa el grado de similitud entre dos valores de columna. Un algoritmo calcula una puntuación de coincidencia como un valor decimal dentro del intervalo de 0 a 1. Un algoritmo asigna una puntuación de uno cuando dos valores de columna son idénticos.

### **exc\_Consolidate**

Una transformación de excepción de registros duplicados que determina qué registros son posibles clientes duplicados, clientes duplicados conocidos o registros de clientes únicos.

#### Tabla Write\_GoodMatches\_Customer

La tabla que recibe todos los registros que no requieren revisión manual. La transformación de excepción de registros duplicados escribe los registros duplicados y los registros únicos en esta tabla.

#### Tabla Target\_CustomerConsolidate

La transformación de excepción escribe los posibles registros duplicados para la tabla Target\_CustomerConsolidate. Los registros de esta tabla requieren revisión manual en Informatica Data Director para Data Quality.

## Transformación de coincidencia

La transformación de coincidencia recibe los datos de los clientes y realiza una coincidencia de identidad.

Configure la transformación de coincidencia para el tipo de salida Clústeres - Coincidir con todos. La transformación de coincidencia devuelve los registros de coincidencia de clústeres. Cada registro de un clúster debe coincidir con al menos otro registro del clúster con una puntuación mayor o igual que el umbral de coincidencia. El umbral de coincidencia es 0.75.

Seleccione la estrategia de coincidencia de divisiones en la ficha **Estrategias** de la transformación de coincidencia. La estrategia de divisiones es una estrategia de coincidencia predefinida que identifica una organización en función de los campos de dirección. En la ficha **Estrategias** de la transformación de coincidencia, elija los puertos de entrada que desea examinar en una coincidencia. Configure la ponderación de estrategia como 0,5.

La siguiente figura muestra la configuración de la estrategia de división para la transformación de coincidencia:

General	Definir estrategias de coincidencia				
Puertos					
Tipo de coincidencia					
<b>Estrategias</b>	Estrategia de coincidencia	Nombre personalizado	Ponderación	Campos de coincidencia	Propiedades
Salida de coincidencia	Division	Division1	0.5	ADDR1_1,ADDR1_2,COMPANY_1,COMPANY_2,ADDR2_1,A...	Población: usa, Nivel de coincidencia: TYPICAL
Avanzado					

La transformación de coincidencia añade la información de los clústeres a cada registro de salida. La transformación también añade un RowID único a cada registro.

## Grupos de entrada de excepción de registros duplicados

La transformación de excepción de registros duplicados tiene dos grupos de entrada. La transformación tiene un grupo de datos que recibe los datos de los clientes. La transformación tiene el grupo de controles que contiene la puntuación de coincidencia para la fila, el identificador de fila, y el ID de clúster.

La siguiente figura muestra los grupos de entrada de la transformación de excepción:

	Nombre	Tipo	Precisión	Escala	Entrada	Salida	Predeterminado	Descripción
Entradas								
Datos (11)								
1	CUST_ID	decimal	20	0	<input checked="" type="checkbox"/>	<input type="checkbox"/>		
2	COMPANY	string	200	0	<input checked="" type="checkbox"/>	<input type="checkbox"/>		
3	CONTACT	string	200	0	<input checked="" type="checkbox"/>	<input type="checkbox"/>		
4	TITLE	string	200	0	<input checked="" type="checkbox"/>	<input type="checkbox"/>		
5	ADDR1	string	200	0	<input checked="" type="checkbox"/>	<input type="checkbox"/>		
6	ADDR2	string	100	0	<input checked="" type="checkbox"/>	<input type="checkbox"/>		
7	ADDR3	string	100	0	<input checked="" type="checkbox"/>	<input type="checkbox"/>		
8	ADDR4	string	50	0	<input checked="" type="checkbox"/>	<input type="checkbox"/>		
9	COUNTRY	string	50	0	<input checked="" type="checkbox"/>	<input type="checkbox"/>		
10	PHONE	string	100	0	<input checked="" type="checkbox"/>	<input type="checkbox"/>		
11	EMAIL	string	100	0	<input checked="" type="checkbox"/>	<input type="checkbox"/>		
Control (3)								
1	Score	double	15	0	<input checked="" type="checkbox"/>	<input type="checkbox"/>		
2	Row_Identifier	string	25	0	<input checked="" type="checkbox"/>	<input type="checkbox"/>		
3	Cluster_ID	integer	10	0	<input checked="" type="checkbox"/>	<input type="checkbox"/>		

El grupo de datos contiene los datos de los clientes. Los datos de los clientes incluyen el ID del cliente, el contacto, la actividad profesional y los campos de dirección. El grupo de controles contiene los metadatos adicionales que la transformación de coincidencia ha añadido para cada registro del cliente. El grupo de controles contiene la puntuación de coincidencia, el ID de fila y el ID de clúster.

## Vista de configuración de ejemplo de excepción de registros duplicados

Defina los umbrales superior e inferior en la vista **Configuración**. Identifique dónde escribe la transformación los registros de clientes duplicados, los posibles registros duplicados y los registros de clientes únicos.

La siguiente figura muestra la vista **Configuración** de la transformación de excepción de registros duplicados:

Umbrales de revisión manual

Umbral inferior : 0,80

Umbral superior : 0,95

Opciones de distribución de datos

Tipo	Salida estándar	Tabla de registro duplicado
Consolidación automática (sobre el umbral superior)	<input checked="" type="checkbox"/>	<input type="checkbox"/>
Consolidación manual (entre umbrales)	<input type="checkbox"/>	<input type="checkbox"/>
Registros únicos (por debajo del umbral inferior)	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>

☐ Crear un grupo de salida aparte para los registros únicos

Generar tabla de registros duplicados

La siguiente tabla describe las opciones de configuración:

Opción	Valor
Umbral inferior	.80
Umbral superior	.95
Consolidación automática	Tabla de salida estándar
Consolidación manual	Tabla de registros duplicados
Registros únicos	Tabla de salida estándar

Haga clic en **Generar tabla de registros duplicados** para crear la tabla de registros duplicados. No cree una tabla independiente para los registros únicos. La transformación escribe los registros únicos en la tabla de salida estándar.

## Registros de tabla de salida estándar

La tabla de destino Write\_GoodMatches\_Customer recibe filas desde el grupo Salida estándar. La tabla recibe los registros únicos y los registros que son duplicados. Estos registros no requieren de revisión manual.

La siguiente figura muestra los registros de la salida estándar que la transformación de excepción devuelve:

Salida									
Nombre: <a href="#">exc Consoli.Good Records</a>									
	Score	Row_Identifier	Cluster_ID	CUST_ID	COMPANY	CONTACT	TITLE	ADDR1	ADDR2 ▲ ADDR3
1	0.9166666666...	1 - 345	260	10111568	SUSQUEHAN...	BALA CYNWYD	United Parcel Serv...	"401 CITY LIN..."	"401 CITY LIN...
2	0.9166666666...	1 - 835	1174	15951678	"ENOW	INC."	LOS ANGELES	US Postal Int...	"10435 SA... 2ND"
3	0.9166666666...	1 - 440	260	10111949	TULLETT & T...	** SYSTEM TE...	Federal Express 2-...	** SYSTEM TE...	"200 OLD C... STE 215S"
4	0.9166666666...	1 - 487	260	10112102	NATIONAL Q...	NEW YORK	Federal Express 2-...	11 PENN PLA...	"200 OLD C... STE 215S"
5	0.9166666666...	1 - 533	260	10112292	PHELPS DOD...	TEMPE	Federal Express 2-...	1501 W. FOU...	"200 OLD C... STE 215S"
6	0.9583333333...	1 - 580	260	10112460	GLEN DOSHAY	RANCHO SA...	Federal Express 2-...	6279 VIA CA...	"200 OLD C... STE 215S"
7	0.9166666666...	1 - 627	260	10112646	HANI QATTAN	HOUSTON	United Parcel Serv...	5404 CEDAR...	"200 OLD C... STE 215S"
8	0.9166666666...	1 - 674	260	10112817	CENTRAL COL...	WESTERVILLE	Federal Express 2-...	975 S. SUNB...	"200 OLD C... STE 215S"
9	0.9583333333...	1 - 721	1174	15951251	EVERETT SEC...	MINNEAPOLIS	US Postal Internat...	2930 NORWE...	"200 OLD C... STE 215S"
10	0.9166666666...	1 - 768	1174	15951449	ROBERT G PE...	NEW YORK	Federal Express 2-...	86 TRINITY P...	"200 OLD C... STE 215S"
11	0.9166666666...	1 - 815	1174	15951607	YMCA RETIRE...	NEW YORK	United Parcel Serv...	140 BROAD...	"200 OLD C... STE 215S"
12	0.9166666666...	1 - 234	260	10111114	"NIPPON OIL...	LTD."	SAN RAMON	United Parcel...	"2680 BISH... SUITE 275"
13	0.9047619047...	1 - 10	4	7121569	RICE VOELKER	MANDEVILLE	Federal Express 2-...	3840 HIGHW...	"2ND & D S... SW HOUSE...
14	0.9166666666...	1 - 432	260	10111932	LA CURRENC...	LOS ANGELES	Federal Express 2-...	BRADLEY INT...	"330 CONG... 6TH FL"
15	0.9166666666...	1 - 572	260	10112429	NAEGELE AN...	CHICAGO	Federal Express 2-...	401 S LASAL...	"330 CONG... 6TH FL"
16	0.9166666666...	1 - 619	260	10112618	BANQUE FRA...	HOUSTON	United Parcel Serv...	333 CLAY ST...	"330 CONG... 6TH FL"
17	0.9166666666...	1 - 666	260	10112777	UNIVERSAL S...	UNIVERSAL CI...	Federal Express 2-...	100 UNIVERS...	"330 CONG... 6TH FL"
18	0.9166666666...	1 - 807	1174	15951579	CREDIT LYON...	NEW YORK	US Postal 2-day	180 MAIDEN...	"330 CONG... 6TH FL"
19	0.9583333333...	1 - 201	260	10110942	"WEST MAUI...	LLC"	LAHAINA	US Postal 2-d...	"5095 NAPI... SUITE 202 N...
20	1	1 - 3	16	1001604	KPMG PEAT...	WOODCLIFF L...	US Postal 2-day	530 CHESNU...	"50 BROAD... 20TH FL"
21	0.9166666666...	1 - 456	260	10111994	"CREDIT AGRI...	INC."	NEW YORK	Federal Expre...	"520 MADL... 3RD FLOOR"
22	0.9166666666...	1 - 238	260	10111126	"SMITH BARN...	INC."	CLAYTON	United Parcel...	"7701 FORS... 12TH FL"
23	0.9166666666...	1 - 526	260	10112268	"BANK OF NE...	LLC"	NEW YORK	US Postal Ov...	"787 SEVEN... 34TH FLR."

El registro contiene los siguientes campos:

#### Puntuación

Puntuación de coincidencia que indica el grado de similitud entre un registro y otro registro del clúster. Los registros con una puntuación de coincidencia de uno son registros duplicados que no necesitan revisarse. Un clúster en el que cualquier registro tiene una puntuación de coincidencia por debajo del umbral inferior no es un clúster duplicado.

#### Row\_Identifier

Un número de fila que identifica de forma exclusiva cada fila en la tabla. Para este ejemplo, el identificador de fila contiene el ID de cliente.

#### Cluster\_ID

Un identificador exclusivo para un clúster. Cada registro en un clúster recibe el mismo ID de clúster. Los primeros cuatro registros de los datos de salida de ejemplo son únicos. Cada registro tiene un ID de clúster único. Las filas entre la cinco y la nueve pertenecen al clúster cinco. Cada registro en este clúster es un duplicado registro debido a las similitudes en los campos de dirección.

#### Campos de datos de origen

El grupo de la tabla de salida estándar también recibe todos los campos de datos de origen.

## Salida del clúster

La tabla Target\_CustomerConsolidate recibe los registros del grupo de salida de clúster. El grupo de salida del clúster devuelve los registros que pueden ser de registros duplicados. Los registros de la tabla Target\_CustomerConsolidate necesitan revisión manual en Informatica Data Director para Data Quality.

La siguiente imagen muestra algunos de los registros y campos de la tabla destino Target\_CustomerConsolidate:

Salida													
Nombre: exc Consoli.Cluster Data													
	Row_Identifier	Sequential_Cluster_ID	Cluster_ID	Score	Is_Master	Workflow_ID	CUST_ID	COMPANY	CONTACT	TITLE	ADDR1	ADDR2	
1	0	0	3	1	Y	DummyWorkflowID	1001622	INVESTEX	NEW YORK		"50 BROAD STREET	20TH FL"	
2	1	0	3	1	N	DummyWorkflowID	1001622	INVESTEX	NEW YORK		"50 BROAD STREET	20TH FL"	
3	2	1	4	1	Y	DummyWorkflowID	7121564		INC."		Federal Express 2-day		
4	3	1	4	1	N	DummyWorkflowID	7121564		INC."		Federal Express 2-day		
5	4	1	4		N	DummyWorkflowID	7121565		NEW YORK		767 FIFTH AVENUE		
6	5	1	4		N	DummyWorkflowID	7121566	K2 ADVISORS	NEW YORK	Courier			
7	6	1	4		N	DummyWorkflowID	7121567	MFP INVESTORS	SHORT HILLS			51 JFK PARKWAY	
8	7	1	4		N	DummyWorkflowID	7121568	EASTON	FORT LEE			ONE PARKER PLAZA	
9	8	1	4		N	DummyWorkflowID	7121569	RICE VOELKER	MANDEVILLE				
10	9	2	16	1	Y	DummyWorkflowID	1001590	E-AGENCY	OAKLAND		291 3RD STREET		
11	10	2	16	1	N	DummyWorkflowID	1001590	E-AGENCY	OAKLAND		291 3RD STREET		
12	11	2	16		N	DummyWorkflowID	1001599	BANK ONE	ELGIN				
13	12	2	16	1	N	DummyWorkflowID	1001604			US Postal 2-day			
14	13	2	16	1	N	DummyWorkflowID	1001658		WASHINGTON			"2ND & D STREET	
15	14	2	16		N	DummyWorkflowID	1001660		ENGLEWOOD		6803 S. TUCSON WAY		
16	15	2	16		N	DummyWorkflowID	1001659		FT. WORTH	Courier	301 COMMERCE ST.		
17	16	2	16	1	N	DummyWorkflowID	1001658		NEW YORK			875 3RD AVE.	
18	17	2	16		N	DummyWorkflowID	1001691	EPICENTRIC		US Postal 2-day	333 BRYANT STREET		
19	18	2	16		N	DummyWorkflowID	1001664		SAN ANTONIO			1100 N.E. LOOP 410	
20	19	2	16		N	DummyWorkflowID	1001694		BOSTON	US Postal 2-day	"125 BROAD STREET		
21	20	2	16	1	N	DummyWorkflowID	1001729						
22	21	2	16		N	DummyWorkflowID	1001724		NEW YORK				
23	22	2	16		N	DummyWorkflowID	1001732			Courier			

El registro contiene los siguientes campos:

#### Row\_Identifier

Un número que identifica de forma exclusiva cada fila de la tabla.

#### Sequential\_Cluster\_ID

Un identificador secuencial de cada clúster para revisar en una tarea de humana. La transformación de excepción de registros duplicados añade el ID secuencial de clúster a los registros del grupo de salida de datos del clúster.

#### Cluster\_ID

Un identificador exclusivo para un clúster. La transformación de coincidencia asigna un ID de clúster a todos los registros de salida. Los registros duplicados y los posibles registros duplicados comparten un ID de clúster. Un registro único recibe un ID del clúster, pero el registro no comparte el número de ID con ningún otro registro.

#### Puntuación

Puntuación de coincidencia que indica el grado de similitud entre un registro y otro registro del clúster. Los registros que requieren revisión manual tienen puntuaciones de coincidencia inferiores que .95 y mayores que .80.

#### Is\_Master

Indica si el registro es el registro preferido del clúster.

#### WorkflowID

El identificador de flujo de trabajo es DummyWorkflowID porque la transformación no está en un flujo de trabajo.

#### Campos de registros

Los demás campos del registro contienen los datos de origen.

# Cómo crear una transformación de excepción de registros duplicados

Cuando configure una transformación de excepción de registros duplicados, configure los puertos de entrada. Defina los umbrales superior e inferior para determinar las coincidencias. Configure dónde escribir los registros duplicados y los registros únicos.

1. Cree una transformación de excepción de registros duplicados reutilizable o no reutilizable.
  - Para crear una transformación reutilizable, seleccione **Archivo > Nuevo > Transformación**.
  - Para crear una transformación no reutilizable, abra una asignación y añada la transformación en el lienzo de la asignación.

Seleccione una transformación de excepción de registros duplicados.

2. En la vista Configuración, configure los umbrales de puntuación de coincidencia superior e inferior.
3. En la sección **Opciones de distribución de datos**, configure las propiedades estándar de la salida y de la tabla de excepciones para establecer dónde la transformación escribirá cada tipo de registro.

También puede modificar dónde escribir los registros duplicados, registros duplicados para revisar y registros únicos.

4. También puede generar una tabla de registros únicos. Introduzca la conexión de base de datos y la información del nombre de tabla para la nueva tabla. La tabla de base de datos debe encontrarse en un esquema predeterminado. Si genera una tabla de registros únicos, la transformación crea un objeto de base de datos en el repositorio de modelos.
5. Configure los puertos de entrada. Cuando se añade un puerto de entrada, Developer Tool añade el mismo nombre de puerto a los grupos de salida.
  - Si crea una transformación reutilizable, seleccione la ficha **Puertos** y añada puertos para los datos que desea conectar a la transformación.
  - Si crea una transformación no reutilizable, añada otros objetos al lienzo de la asignación y arrastre los puertos de entrada para la transformación.
6. Conecte los puertos de salida de la transformación a uno o varios destinos de datos. Conecte los puertos de salida a los objetos de datos que correspondan a las opciones de salida que ha establecido en la vista **Configuración**.
  - Si crea una transformación reutilizable, añada la transformación a una asignación y conecte los puertos de salida.
  - Si crea una transformación no reutilizable, la transformación conecta los puertos a la tabla de datos del clúster. Debe conectar los puertos de salida al resto de los destinos de datos.

## CAPÍTULO 13

# Transformación de expresión

Este capítulo incluye los siguientes temas:

- [Resumen de la transformación de expresión, 164](#)
- [Cómo desarrollar una transformación de expresión, 164](#)
- [Propiedades avanzadas de la transformación de expresión, 165](#)
- [Puertos de transformación de expresión, 165](#)

## Resumen de la transformación de expresión

La transformación de expresión es una transformación pasiva que se puede usar para realizar cálculos no agregados en una fila individual. También puede usar la transformación de expresión para probar instrucciones condicionales antes de pasar los resultados a un destino o a otras transformaciones.

Por ejemplo, quizás necesite corregir salarios de empleados, concatenar nombres y apellidos o convertir cadenas en números.

Puede especificar varias expresiones en una transformación de expresión individual creando una expresión para cada puerto de salida. Por ejemplo, quizá quiera calcular distintos tipos de retenciones de impuesto para cada nómina de empleado, como impuestos de renta locales y federales, seguridad social y mutua médica. Puesto que estos cálculos necesitan el salario del empleado y la categoría de retención, es posible que también pueden necesitar la tasa de impuesto correspondiente, por lo que puede crear puertos de transferencia para el salario y para la categoría de retención. También puede crear un puerto de salida aparte para cada cálculo.

## Cómo desarrollar una transformación de expresión

Cuando se desarrolla una transformación de expresión, es necesario tener en cuenta factores, como los tipos de puerto y los tipos de expresión que se desea crear.

Tenga en cuenta los siguientes factores cuando desarrolle una transformación de expresión:

- El tipo de puerto, como entrada, salida, transferencia o variable. Necesita incluir un puerto de entrada o de transferencia y un puerto de salida para utilizar una expresión en una transformación de expresión.
- Las expresiones que desea utilizar en la transformación.

# Propiedades avanzadas de la transformación de expresión

Configure las propiedades para ayudarle a determinar cómo el servicio de integración de datos procesa los datos para la transformación de expresión.

Puede configurar niveles de seguimiento de registros.

Configure la siguiente propiedad en la ficha **Avanzadas**:

## Nivel de seguimiento

Cantidad de detalles que aparecen en el registro de esta transformación. Puede elegir simplificado, normal, inicialización detallada o datos detallados. El valor predeterminado es Normal.

# Puertos de transformación de expresión

Una transformación de expresión tiene distintos tipos de puerto que permiten realizar diversas tareas de transformación, como pasar datos a transformaciones de un nivel inferior e introducir expresiones.

Una transformación de expresión tiene los siguientes tipos de puerto:

## Entrada

Recibe datos desde transformaciones de un nivel superior. Los puertos de entrada proporcionan los valores utilizados en un cálculo.

## Salida

Proporciona el valor de devolución de la expresión. La expresión se introduce como una opción de configuración del puerto de salida. También puede configurar un valor predeterminado para cada puerto.

## Transferencia

Pasa los datos inalterados.

## Variable

Se usan para variables locales.

## CAPÍTULO 14

# Transformación de filtro

Este capítulo incluye los siguientes temas:

- [Resumen de la transformación de filtro, 166](#)
- [Propiedades avanzadas de la transformación de filtro, 166](#)
- [Condición de filtro, 167](#)
- [Consejos sobre el rendimiento de la transformación de filtro, 167](#)

## Resumen de la transformación de filtro

Utilice la transformación de filtro para filtrar filas de una asignación. Como transformación activa, la transformación de filtro puede cambiar el número de filas que pasan por ella.

La transformación de filtro permite que pasen las filas que cumplen la condición de filtro especificada. Suelta las filas que no cumplen la condición. Se pueden filtrar los datos según una o más condiciones.

Una condición de filtro devuelve TRUE o FALSE para cada fila que evalúa el servicio de integración de datos si una fila cumple o no la condición especificada. Para cada fila que devuelve TRUE, el servicio de integración de datos pasa por la transformación. Para cada fila que devuelve FALSE, el servicio de integración de datos se suelta y escribe un mensaje en el registro.

No se pueden concatenar puertos de más de una transformación en la transformación de filtro. Los puertos de entrada para el filtro deben proceder de una transformación individual.

## Propiedades avanzadas de la transformación de filtro

Configure las propiedades para ayudarle a determinar cómo el servicio de integración de datos procesa los datos para la transformación de filtro.

Puede configurar niveles de seguimiento de registros.

Configure la siguiente propiedad en la ficha **Avanzadas**:

### Nivel de seguimiento

Cantidad de detalles que aparecen en el registro de esta transformación. Puede elegir simplificado, normal, inicialización detallada o datos detallados. El valor predeterminado es Normal.

## Condición de filtro

La condición de filtro es una expresión que devuelve TRUE o FALSE.

Especifique condiciones en el editor de expresiones. La condición de filtro distingue entre mayúsculas y minúsculas.

Puede utilizar como filtro cualquier expresión que devuelve un valor individual. Por ejemplo, si desea filtrar las filas de los empleados cuyo salario es inferior a 30.000 USD, especifique la siguiente condición:

```
SALARY > 30000
```

Puede especificar varios componentes para la condición utilizando los operadores lógicos AND y OR. Si desea filtrar los empleados que ganan menos de 30.000 USD y más de 100.000 USD, especifique la siguiente condición:

```
SALARY > 30000 AND SALARY < 100000
```

También puede especificar una constante para la condición de filtro. El equivalente numérico de FALSE es cero (0). Todo valor distinto de cero es el equivalente de TRUE. Por ejemplo, la transformación contiene un puerto llamado NUMBER\_OF\_UNITS con un tipo de datos numérico. Configure una condición de filtro para que devuelva FALSE si el valor de NUMBER\_OF\_UNITS es igual a cero. De lo contrario, la condición devuelve TRUE.

No necesita especificar TRUE o FALSE como valores en la expresión. TRUE y FALSE son valores de devolución implícitos de cualquier condición que se define. Si la condición de filtro se evalúa como NULL, la fila se trata como FALSE.

## Cómo filtrar filas con valores nulos

Para filtrar filas que contienen valores nulos o espacios, utilice las funciones ISNULL e IS\_SPACES para probar el valor del puerto.

Por ejemplo, si desea filtrar las filas que contienen el valor NULL en el puerto FIRST\_NAME, utilice la siguiente condición:

```
IIF (ISNULL (FIRST_NAME) , FALSE, TRUE)
```

Esta condición indica que, si el puerto FIRST\_NAME es NULL, el valor de devolución es FALSE y que la fila debe descartarse. De lo contrario, la fila pasa a la siguiente transformación.

## Consejos sobre el rendimiento de la transformación de filtro

Siga los consejos para aumentar el rendimiento de la transformación de filtro.

**Utilice la transformación de filtro en los primeros lugares de la asignación.**

Mantenga la transformación de filtro lo más cerca posible de los orígenes en la asignación. En lugar de pasar las filas que tiene previsto descartar mediante la asignación, puede filtrar los datos no deseados en los primeros lugares del flujo de datos desde los orígenes a los destinos.

## CAPÍTULO 15

# Transformación de Java

Este capítulo incluye los siguientes temas:

- [Resumen de la transformación de Java, 168](#)
- [Cómo diseñar una transformación de Java, 171](#)
- [Puertos de transformación de Java, 172](#)
- [Propiedades avanzadas de la transformación de Java, 173](#)
- [Cómo desarrollar código Java, 176](#)
- [Propiedades de Java de una transformación de Java, 180](#)
- [Optimización de filtro con la transformación de Java, 183](#)
- [Cómo crear una transformación de Java, 185](#)
- [Cómo compilar una transformación de Java, 186](#)
- [Solución de problemas de una transformación de Java, 187](#)

## Resumen de la transformación de Java

Utilice la transformación de Java para ampliar las funciones de Developer Tool.

La transformación de Java ofrece una simple interfaz de programación nativa para definir funciones de transformación con el lenguaje de programación Java. Puede utilizar la transformación de Java para definir funciones simples o moderadamente complejas sin tener conocimientos avanzados del lenguaje de programación Java o de un entorno de desarrollo externo de Java. La transformación de Java puede ser una transformación activa o pasiva.

Developer Tool utiliza el kit de desarrollo de Java (JDK) para compilar el código de Java y generar el código byte para la transformación. Developer Tool almacena el código byte en el repositorio de modelos.

El servicio de integración de datos utiliza el Java Runtime Environment (JRE) para ejecutar el código byte en tiempo de ejecución. Cuando el servicio de integración de datos ejecuta una asignación con una transformación de Java, utiliza el JRE para ejecutar el código byte y procesar filas de entrada y generar filas de salida.

Cree transformaciones de Java escribiendo snippets de código Java que definan una lógica de transformación. Defina el comportamiento de la transformación de Java de acuerdo con los siguientes eventos:

- La transformación recibe una fila de entrada.
- La transformación ha procesado todas las filas de entrada.

## Transformaciones de Java reutilizables y no reutilizables

Puede crear una transformación de Java reutilizable o no reutilizable.

Las transformaciones reutilizables pueden existir en varias asignaciones.

Las transformaciones no reutilizables existen en una única asignación.

Las vistas del editor donde se definen las propiedades y se crea el código Java difieren según se crea una transformación de Java reutilizable o no reutilizable.

### Vistas del editor para una transformación de Java reutilizable

Se definen propiedades y se crea código Java para una transformación de Java en vistas del editor.

Para crear una transformación de Java reutilizable, véase ["Cómo crear una transformación de Java reutilizable" en la página 185](#).

Para las transformaciones de Java reutilizables, están disponibles las siguientes vistas:

#### Resumen

Escriba el nombre y la descripción de la transformación y cree y configure los puertos de entrada y de salida.

#### Java

Defina, compile y solucione errores de compilación del código Java. En la vista **Java** están disponibles las siguientes fichas:

- Fichas de introducción de código, en las que se pueden definir snippets de código Java.
- La ficha **Código completo**, en la que se puede ver el código completo de la clase Java de la transformación de Java.

#### Avanzadas

Defina propiedades avanzadas para la transformación de Java.

### Vistas del editor para una transformación de Java no reutilizable

Se definen propiedades y se crea código Java para una transformación de Java no reutilizable en vistas del editor.

Para crear una transformación de Java no reutilizable, véase ["Cómo crear una transformación de Java no reutilizable" en la página 186](#).

Para las transformaciones de Java no reutilizables, están disponibles las siguientes vistas:

#### General

Proporcione el nombre y la descripción de la transformación.

#### Puertos

Cree y configure puertos de entrada y de salida.

#### Java

Defina, compile y solucione errores de compilación del código Java. En la vista **Java** están disponibles las siguientes fichas:

- Fichas de introducción de código, en las que se pueden definir snippets de código Java.
- La ficha **Código completo**, en la que se puede ver el código completo de la clase Java de la transformación de Java.

## Avanzadas

Defina propiedades avanzadas para la transformación de Java.

## Transformaciones de Java activas y pasivas

Cuando se crea una transformación de Java, se puede definir como activa o pasiva.

Después de definir el tipo de transformación, no puede modificarlo.

Una transformación de Java ejecuta el código Java que se define en la ficha **En entrada** una vez para cada fila de datos de entrada.

Una transformación de Java maneja las filas de salida conforme al tipo de transformación, tal como se describe a continuación:

- Una transformación de Java pasiva genera una fila de salida para cada fila de entrada de la transformación después de procesar cada una de las filas de entrada.
- Una transformación de Java activa genera varias filas de salida para cada fila de entrada de la transformación.

Utilice el método `generateRow` para generar cada fila de salida. Por ejemplo, si la transformación contiene dos puertos de entrada que representan una fecha de inicio y una fecha de finalización, puede utilizar el método `generateRow` para generar una fila de salida para cada fecha entre la fecha de inicio y la fecha de finalización.

## Conversión de tipos de datos

Una transformación de Java convierte tipos de datos de Developer Tool en tipos de datos de Java según el tipo de puerto de la transformación de Java.

Cuando una transformación de Java lee filas de entrada, convierte los tipos de datos del puerto de entrada en tipos de datos de Java.

Cuando una transformación de Java escribe filas de salida convierte los tipos de datos de Java en tipos de datos de puerto de salida.

Por ejemplo, el siguiente proceso se produce para un puerto de entrada con el tipo de datos Integer en una transformación de Java:

1. La transformación de Java convierte el tipo de datos Integer del puerto de entrada en el tipo de datos de Java primitivo `int`.
2. En la transformación, ésta trata el valor del puerto de entrada como tipo de datos de Java primitivo `int`.
3. Cuando la transformación genera la fila de salida, convierte el tipo de datos de Java primitivo `int` en el tipo de datos Integer.

La siguiente tabla muestra cómo la transformación de Java asigna tipos de datos de Developer Tool a tipos de datos de Java primitivos y complejos:

Tipo de datos de Developer Tool	Tipo de datos de Java
bigint	long
binary	byte[]

Tipo de datos de Developer Tool	Tipo de datos de Java
date/time	Con procesamiento de nanosegundos habilitado, BigDecimal con precisión al nanosegundo Con procesamiento de nanosegundos deshabilitado, long con precisión al milisegundo (número de milisegundos desde el 1 de enero de 1970 00:00:00.000 GMT).
decimal	Con procesamiento de alta precisión deshabilitado, double con precisión 15. Con procesamiento de alta precisión habilitado, BigDecimal
double	double
integer	int
string	String
text	String

En Java, los tipos de datos String, byte[] y BigDecimal son tipos de datos complejos, y los tipos de datos double, int y long son tipos de datos primitivos.

**Nota:** La transformación de Java define los valores nulos en tipos de datos primitivos como cero. Puede usar los métodos API `isNull` y `setNull` en la ficha **En entrada** para definir valores nulos en el puerto de entrada como valores nulos en el puerto de salida. Para un ejemplo, véase [“setNull” en la página 197](#).

## Cómo diseñar una transformación de Java

Cuando se diseña una transformación de Java, es necesario tener en cuenta factores, como el tipo de transformación que se desea crear.

Cuando diseñe una transformación de Java, tenga en cuenta los siguientes aspectos:

- ¿Necesita crear una transformación de Java activa o pasiva?  
Una transformación de Java pasiva genera una fila de salida para cada fila de entrada de la transformación.  
Una transformación de Java activa genera varias filas de salida para cada fila de entrada de la transformación.
- ¿Necesita definir funciones en la transformación de Java? En caso afirmativo, ¿qué expresiones desea incluir en cada función?  
Por ejemplo, puede definir una función que invoque una expresión para consultar los valores de los puertos de entrada y de salida o para buscar los valores de las variables de la transformación de Java.
- ¿Desea crear una transformación de Java reutilizable o no reutilizable?  
Una transformación reutilizable puede existir en varias asignaciones.  
Una transformación no reutilizable puede existir en una única asignación.

# Puertos de transformación de Java

Una transformación de Java puede tener puertos de entrada y de salida.

Para crear y editar los puertos de una transformación de Java no reutilizable, utilice la ficha **Puertos** del editor. Para crear y editar los puertos de una transformación de Java reutilizable, utilice la vista **Resumen** en el editor.

Puede especificar valores predeterminados para los puertos. Después de añadir puertos a una transformación, puede usar los nombres de puerto como variables en snippets de código Java.

## Cómo crear puertos

Cuando se crea una transformación de Java, ésta incluye un grupo de entrada y un grupo de salida.

Cuando se crea un puerto, la herramienta Tool lo añade debajo de la fila o el grupo seleccionado en ese momento.

## Cómo configurar valores de puerto predeterminados

Se pueden definir valores predeterminados para los puertos en una transformación de Java.

La transformación de Java inicializa las variables del puerto con el valor de puerto predeterminado basado en el tipo de datos del puerto.

### Puertos de entrada y de salida

La transformación de Java inicializa el valor de puertos de entrada no conectados o de puertos de salida que no tienen un valor asignado en los snippets de código Java.

La transformación de Java inicializa los puertos en función de los tipos de datos de puerto siguientes:

#### Simple

Si se define un valor predeterminado para el puerto que no es igual a nulo, la transformación inicializa el valor de la variable del puerto con el valor predeterminado. De lo contrario, inicializa el valor de la variable del puerto con 0.

#### Compleja

Si se define un valor predeterminado para el puerto, la transformación crea un nuevo objeto String o byte[] e inicializa el objeto con el valor predeterminado. De lo contrario, la transformación inicializa la variable de puerto con un valor nulo.

**Nota:** Si se accede a una variable de puerto de entrada con un valor nulo en el código Java, se produce una NullPointerException.

Puede habilitar un puerto de entrada como una clave de participación o una clave de ordenación y asignar una dirección de ordenación al ejecutar asignaciones en un entorno Hive. El servicio de integración de datos garantiza la partición de los datos en las tareas del reductor en el clúster Hadoop y ordena la salida por dirección y clave de ordenación. La clave de participación y la clave de ordenación son válidas en un entorno Hive cuando el ámbito de la transformación se establece en Todas las entradas.

Utilice las siguientes propiedades para la partición y la ordenación de los datos al insertar la transformación en un entorno Hive:

#### Clave de partición

El puerto de entrada que determina la partición en la que se almacena una fila de datos concreta. Al seleccionar la clave de partición y la clave de ordenación, el servicio de integración de datos genera una salida ordenada.

#### Clave de ordenación

El puerto de entrada que determina los criterios de ordenación. Debe seleccionar la clave de participación y la clave de ordenación para garantizar que los datos se ordenan.

#### Dirección

Orden ascendente o descendente. El orden ascendente es el predeterminado.

## Propiedades avanzadas de la transformación de Java

La transformación de Java incluye propiedades avanzadas, tanto para el código de la transformación como para la propia transformación.

Cuando se utiliza la transformación en una asignación, se pueden reemplazar las propiedades de la transformación.

Puede definir las siguientes propiedades avanzadas para la transformación de Java en la ficha **Avanzadas**:

#### Nivel de seguimiento

Cantidad de detalles que aparecen en el registro de esta transformación. Puede elegir simplificado, normal, inicialización detallada o datos detallados. El valor predeterminado es Normal.

#### Habilitar alta precisión

Procesa un puerto de tipo de datos decimal con una precisión mayor de 15 y menor de 28 como puerto de tipo de datos de Java BigDecimal.

Deshabilite el procesamiento de alta precisión para procesar un puerto de tipo de datos decimal con una precisión mayor de 15 y menor de 28 como puerto de tipo de datos de Java Double.

La siguiente tabla muestra cómo una transformación de Java trata un valor en un puerto de entrada con tipo de datos decimal dependiendo si se ha habilitado o no la opción de alta precisión:

Ejemplo	Procesamiento de alta precisión habilitado	Procesamiento de alta precisión deshabilitado
Un puerto de entrada con tipo de datos recibe un valor de 40012030304957666903.	La transformación de Java deja el valor como está.	La transformación de Java convierte el valor al valor siguiente: $4.00120303049577 \times 10^{19}$

#### Usar nanosegundos en Fecha/Hora

Convierte los puertos con tipo de datos date/time en puertos con tipo de datos de Java BigDecimal con precisión al nanosegundo.

Deshabilite el procesamiento al nanosegundo de modo que el código Java generado convierta los puertos con tipo de datos date/time en puerto con tipo de datos Java Long y una precisión al milisegundo.

## Classpath

Establece la classpath para directorios de archivos jar o class que están asociados a paquetes Java estándar que se importan en la ficha **Importaciones**.

Los directorios de los archivos jar o class deben ser accesibles en el equipo cliente de Developer Tool para poder compilar el código Java.

Según el sistema operativo, separe cada una de las entradas de la ruta de acceso de clase del siguiente modo:

- En UNIX, utilice el signo de dos puntos para separar cada una de las entradas de classpath.
- En Windows, utilice un punto y coma para separar cada una de las entradas de classpath.

Por ejemplo, si importa el paquete de conversión de Java en la ficha **Importaciones** y define el paquete en converter.jar, debe añadir la ubicación del archivo converter.jar a la classpath antes de compilar el código Java de la transformación de Java.

**Nota:** No es necesario definir la classpath para paquetes Java incorporados. Por ejemplo, puesto que java.io es un paquete Java incorporado, no es necesario definir la ruta de acceso de clase para java.io.

## Está activo

La transformación puede generar más de una fila de salida por cada fila de entrada.

No se puede modificar esta propiedad después de crear la transformación de Java. Si necesita cambiar esta propiedad, cree una nueva transformación de Java.

## Ámbito de transformación

Define el método que utiliza el servicio de integración de datos para aplicar la lógica de transformación a los datos entrantes. Puede seleccionar uno de los siguientes valores:

- Fila. Aplica la lógica de transformación a una fila de datos cada vez. Elija Fila cuando los resultados del procedimiento dependan de una única fila de datos.
- Transacción. Aplica la lógica de transformación a todas las filas de una transacción. Elija Transacción cuando los resultados del procedimiento dependan de todas las filas en la misma transacción, pero no de las filas en otras transacciones. Cuando elija Transacción, debe conectar todos los grupos de entrada con el mismo punto de control de transacción.
- Todas las entradas. Aplica la lógica de transformación a todos los datos entrantes. Cuando elija Todas las entradas, el servicio de integración de datos quita los límites de transacción. Elija Todas las entradas cuando los resultados del procedimiento dependen de todas las filas de datos en el origen.

**Nota:** La propiedad Ámbito de transformación solamente es válida en un entorno de Hive.

## Sin estado

Procesa los datos de las filas de entrada sin depender del valor de la fila anterior. La propiedad Sin estado solamente es válida en un entorno de Hive.

## Cómo configurar la classpath para el cliente de Developer Tool

Puede añadir archivos jar o directorios de archivos de clase a la classpath del Cliente de Developer Tool.

Para definir la classpath para el equipo en el que se ejecuta el cliente de Developer Tool, lleve a cabo una de las siguientes tareas:

- Configure la variable de entorno CLASSPATH. Defina la variable de entorno CLASSPATH en el equipo cliente donde reside el cliente de Developer Tool. Esto se aplica a todos los procesos Java que se ejecutan en la máquina.
- Para una transformación de Java no reutilizable, configure la classpath en propiedades avanzadas de la transformación de Java. Esto se aplica a asignaciones que incluyen esta transformación de Java. El cliente de Developer Tool incluye archivos dentro de la classpath cuando compila el código Java. No se puede configurar esta propiedad para una transformación de Java reutilizable.

Para añadir directorios de archivos jar o de clase a la classpath en una transformación de Java, lleve a cabo los siguientes pasos:

1. En la ficha **Avanzadas**, haga clic en el icono de flecha descendiente en la columna **Valor**, junto a **Classpath**.

Se abrirá el cuadro de diálogo **Editar ruta de clase**.

2. Para añadir una ruta de clase, complete los siguientes pasos:

- a. Haga clic en **Añadir**.

Aparecerá la ventana **Guardar como**.

- b. En la ventana **Guardar como**, localice el directorio donde se encuentra el archivo jar.

- c. Haga clic en **Aceptar**.

La classpath aparece en el cuadro de diálogo **Editar classpath**.

3. Para eliminar un archivo jar o un directorio de archivos de clase, seleccione el directorio de archivos jar o de clase y haga clic en **Quitar**.

El directorio desaparece de la lista de directorios.

## Cómo configurar la classpath para el servicio de integración de datos

Se pueden añadir directorios de archivos jar o class que necesita la classpath en tiempo de ejecución en el nodo del servicio de integración de datos.

Ponga los archivos jar que se necesitan en tiempo de ejecución en el siguiente directorio del nodo del servicio de integración de datos:

```
$INFA_HOME/services/shared/jars
```

Los archivos jar que se encuentran en esta ubicación se cargan dinámicamente. Cualquier archivo de clase que necesita una asignación individual en tiempo de ejecución se encuentra y se carga desde este directorio.

**Nota:** La transformación de Java añade los archivos jar de este directorio a la classpath a nivel de asignación.

# Cómo desarrollar código Java

Utilice las fichas de introducción de código de la vista **Java** para escribir y compilar código Java que defina el comportamiento de una transformación ante sucesos de transformación específicos.

Puede desarrollar snippets de código en las fichas de introducción de código en el orden que desee. Puede ver, aunque no editar, el código Java completo en la ficha **Código completo**.

Después de desarrollar snippets de código, puede compilar los snippets de código o el código Java completo y ver los resultados de la compilación en la ventana **Resultados** de las propiedades **Compilación**, en la vista **Java**.

Cada ficha de introducción de código contiene componentes que utiliza para escribir, ver y compilar código Java:

## Propiedades de código

Ofrece controles que permiten ver y escribir código Java, incluyendo métodos API de una transformación de Java. La siguiente tabla describe los controles que están disponibles en las propiedades **Código**:

Control	Descripción
Navegador	<p>Muestra puertos de entrada, puertos de salida y métodos API de transformación de Java invocables.</p> <p>Haga clic en un elemento dentro del navegador para mostrar una descripción del elemento.</p> <p>Haga doble clic en un elemento para añadirlo a la ventana <b>Código Java</b>. También puede arrastrar un elemento desde el navegador hasta la ventana <b>Código Java</b>.</p> <p>El navegador está disponible en las siguientes fichas de introducción de código:</p> <ul style="list-style-type: none"><li>- <b>Aplicaciones auxiliares</b></li><li>- <b>En entrada</b></li><li>- <b>Al final</b></li></ul>
Ventana Código Java	<p>Permite ver o escribir código Java para la transformación. La ventana <b>Código Java</b> muestra código Java mediante el resaltado de sintaxis básica de Java.</p> <p><b>Nota:</b> En la ficha <b>Código completo</b> puede ver, aunque no editar, la clase completa del código para la transformación de Java.</p> <p>La ventana <b>Código Java</b> está disponible en las siguientes fichas de introducción de código:</p> <ul style="list-style-type: none"><li>- <b>Importaciones</b></li><li>- <b>Aplicaciones auxiliares</b></li><li>- <b>En entrada</b></li><li>- <b>Al final</b></li><li>- <b>Funciones</b></li><li>- <b>Interfaces del optimizador</b></li><li>- <b>Código completo</b></li></ul>

Control	Descripción
<b>Comando Nueva función</b>	Abre el cuadro de diálogo <b>Definir función</b> que se usa para definir funciones que invocan expresiones de Java. El comando <b>Función</b> está disponible en la ficha <b>Funciones</b> .
Barra de herramientas de edición	Permite hacer clic en iconos de herramientas como cortar y pegar para editar código Java. La barra de herramientas de edición está disponible en las siguientes fichas de introducción de código: <ul style="list-style-type: none"> <li>- <b>Importaciones</b></li> <li>- <b>Aplicaciones auxiliares</b></li> <li>- <b>En entrada</b></li> <li>- <b>Al final</b></li> <li>- <b>Funciones</b></li> <li>- <b>Interfaces del optimizador</b></li> </ul>

#### Propiedades de compilación

Proporciona controles para compilar y depurar código Java. La siguiente tabla describe los controles en las propiedades **Compilación**:

Control	Descripción
<b>Comando Compilar</b>	Compila el código Java de la transformación.
<b>Ventana Resultados</b>	Muestra los resultados de compilación de la clase de la transformación de Java y permite buscar el origen de errores en el código. Para buscar un error en el código, haga clic con el botón derecho en un mensaje de error de la ventana <b>Resultados</b> y selecciónelo para ver el error en el código del snippet o en el código completo. También puede hacer doble clic en un mensaje de error en la ventana <b>Resultados</b> para encontrar el origen del mismo.

## Cómo crear snippets de código Java

Para crear snippets de código que definan el comportamiento de una transformación, utilice la ventana **Código Java** de las fichas de introducción de código .

1. Haga clic en la ficha de introducción de código apropiada.

La siguiente tabla describe las tareas que se pueden completar en las fichas de introducción de código en la vista **Java**:

Ficha	Descripción
Importaciones	Importa paquetes de Java de terceros, incorporados y personalizados para una transformación de Java activa o pasiva. Después de importar paquetes, puede utilizarlos en otras fichas de introducción de código.
Aplicaciones auxiliares	Declara variables y métodos definidos por el usuario para la clase de transformación de Java en una transformación de Java activa o pasiva. Después de declarar las variables y los métodos, puede usarlos en cualquier otra ficha de introducción de código, excepto en la ficha <b>Importaciones</b> .
En entrada	Indica el comportamiento de una transformación de Java activa o pasiva cuando recibe una fila de entrada. El código Java que se define en esta ficha se ejecuta una vez para cada fila de entrada.  En esta ficha también puede acceder y utilizar datos de puertos de entrada y de salida, variables y métodos API de transformación de Java.
Al final	Indica el comportamiento de una transformación de Java activa o pasiva cuando procesa todos los datos de entrada.  En esta ficha también puede definir los datos para transformaciones activas e invocar métodos API de transformación de Java.
Funciones	Define las funciones que invocan expresiones en una transformación de Java mediante el lenguaje de programación Java. Por ejemplo, puede definir una función que invoque una expresión que busque los valores de puertos de entrada o de salida o que busque los valores de variables de una transformación de Java.  En la ficha <b>Funciones</b> puede definir funciones manualmente o hacer clic en <b>Nueva función</b> para invocar el cuadro de diálogo <b>Definir función</b> , que permite definir una función fácilmente.
Interfaces del optimizador	Define la optimización de primera selección o de filtro de inserción. Seleccione el método de optimización en el navegador. Actualice los fragmentos de código para habilitar la optimización. Defina los puertos de entrada y los puertos de salida asociados para insertar la lógica de filtro.
Código completo	Solo lectura. En esta ficha puede ver, aunque no editar, la clase completa del código de la transformación de Java.

- Para acceder a variables de columna de entrada o de salida en el snippet, expanda la lista **Entrada** o **Salida** en el navegador y haga doble clic en el nombre del puerto.
- Para invocar un método API de transformación de Java en el snippet, expanda la lista **API invocables** en el navegador y haga doble clic en el nombre del método. En caso necesario, configure los valores de entrada apropiados para el método.
- Escriba el código Java apropiado, basado en el tipo de ficha de introducción de código.  
  
Visualice el código de clase completo de la transformación de Java, en las ventanas **Código Java**, en la ficha **Código completo**.

## Cómo importar paquetes de Java

En la ficha **Importaciones** se pueden importar paquetes de Java para transformaciones de Java activas o pasivas.

Puede importar paquetes de Java desarrollados por terceros, incorporados o personalizados. Después de importar paquetes de Java, puede utilizarlos en otras fichas de introducción de código.

**Nota:** En la ficha **Importaciones** no se pueden declarar o usar variables estáticas, variables de instancia o métodos de usuario.

En el Developer Tool, cuando se exportan o importan datos que contienen una transformación de Java, los archivos jar o class que contienen los paquetes de terceros o paquetes personalizados requeridos por la transformación de Java no se incluyen en la exportación ni en la importación.

Si se importan metadatos que contienen una transformación de Java, debe copiar los archivos jar o class que contienen los paquetes de terceros o paquetes personalizados al cliente de Developer Tool y al nodo del servicio de integración de datos.

Por ejemplo, para importar el paquete Java I/O, especifique el siguiente código en la ficha **Importaciones**:

```
import java.io.*;
```

Cuando se importan paquetes de Java que no son estándar, añada el paquete o la clase a la classpath en la de la transformación de Java.

## Cómo definir el código de una aplicación auxiliar

En la ficha **Aplicaciones auxiliares**, puede declarar variables y métodos definidos por el usuario para la clase de una transformación de Java en transformaciones de Java activas o pasivas.

Después de declarar variables y métodos en la ficha **Aplicaciones auxiliares**, puede utilizar las variables y los métodos en cualquier ficha de introducción de código, excepto en la ficha **Importaciones**.

En la ficha **Aplicaciones auxiliares**, puede declarar los siguientes tipos de código, variables y métodos:

- Código estático y variables estáticas.

Dentro de un bloque estático puede declarar variables estáticas y código estático. Todas las instancias de una transformación de Java reutilizable en una asignación comparten código estático y variables estáticas. En una transformación de Java, el código estático se ejecuta antes de cualquier otro código.

Por ejemplo, el siguiente código declara una variable estática para almacenar el umbral de error para todas las instancias de una transformación de Java en una asignación:

```
static int errorThreshold;
```

Utilice esta variable para almacenar el umbral de error para la transformación y acceder al mismo desde todas las instancias de la transformación de Java de una asignación.

**Nota:** Debe sincronizar las variables estáticas en una transformación de Java reutilizable.

- Variables de instancia.

Varias instancias de una transformación de Java reutilizable en una asignación no comparten variables estáticas. Declare variables de instancia con un prefijo para evitar conflictos e inicializar variables de instancia no primitivas.

Por ejemplo, el siguiente código utiliza una variable booleana para decidir si se genera o no una fila de salida:

```
// boolean to decide whether to generate an output row
// based on validity of input
private boolean generateRow;
```

- Métodos estáticos o instancias definidas por el usuario.

Amplía las funciones de la transformación de Java. Los métodos de Java declarados en la ficha **Aplicaciones auxiliares** pueden usar o modificar variables de salida o variables de instancia declaradas localmente. No se puede acceder a variables de entrada desde métodos de Java en la ficha **Aplicaciones auxiliares**.

Por ejemplo, utilice el siguiente código en la ficha **Aplicaciones auxiliares** para declarar una función que añada dos enteros:

```
private int myTXAdd (int num1,int num2)
{
    return num1+num2;
}
```

## Propiedades de Java de una transformación de Java

Utilice las fichas de introducción de código de la vista **Java** para escribir y compilar código Java que defina el comportamiento de una transformación ante sucesos de transformación específicos.

Las siguientes fichas son las fichas para la introducción de código:

- **Importaciones**
- **Aplicaciones auxiliares**
- **En entrada**
- **Al final**
- **Funciones**
- **Interfaces del optimizador**

Visualice el código de clase completo para la transformación de Java en la ficha **Código completo**.

### Ficha Importaciones

En la ficha **Importaciones** se pueden importar paquetes de Java de terceros, incorporados o personalizados para transformaciones de Java activas o pasivas.

Para importar un paquete de Java, introduzca el código para importar el paquete en la ventana **Código Java**, en las propiedades **Código** de la ficha **Importaciones**.

Por ejemplo, puede introducir el siguiente código para importar el paquete java.io:

```
import java.io.*;
```

Para compilar el código que importa paquetes de Java, haga clic en **Compilar**, en las propiedades **Compilación** de la ficha **Importaciones**. Los resultados de la compilación aparecen en la ventana **Resultados** de la ficha **Importaciones**.

Después de importar paquetes de Java, puede utilizarlos en otras fichas de introducción de código.

### Ficha Aplicaciones auxiliares

En la ficha **Aplicaciones auxiliares** puede declarar variables y métodos definidos por el usuario para la clase de transformación de Java en un archivo o en una transformación de Java pasiva.

Para declarar variables y métodos definidos por el usuario, introduzca el código en la ventana **Código Java**, en las propiedades **Código** de la ficha **Aplicaciones auxiliares**.

Para compilar el código de la aplicación auxiliar para la transformación de Java, haga clic en **Compilar**, en las propiedades **Compilación** de la ficha **Aplicaciones auxiliares**. Los resultados de la compilación aparecen en la ventana **Resultados** de la ficha **Aplicaciones auxiliares**.

Después de declarar las variables y los métodos, puede usarlos en cualquier otra ficha de introducción de código, excepto en la ficha **Importaciones**.

## Ficha En entrada

En la ficha **En entrada** se indica el comportamiento de una transformación de Java activa o pasiva cuando recibe una fila de entrada. En esta ficha también puede acceder y utilizar datos de puertos de entrada y de salida, variables y métodos API de transformación de Java.

El código Java que se define en esta ficha se ejecuta una vez para cada fila de entrada.

Para definir cómo se comporta una transformación de Java cuando recibe una fila de entrada, especifique el código en la ventana **Código Java**, en las propiedades **Código** de la ficha **En entrada**.

Desde el explorador de la ficha **En entrada**, puede acceder a y definir las siguientes variables y métodos API:

- Variables de puerto de entrada y de puerto de salida. Acceda a los datos de los puertos de entrada y de salida utilizando el nombre del puerto como nombre de la variable. Por ejemplo, si "in\_int" es un puerto de entrada con el tipo de datos Integer, puede acceder a los datos de este puerto haciendo referencia al mismo como la variable "in\_int" con el tipo de datos primitivo de Java int. No es necesario que declare puertos de entrada y de salida como variables.

No asigne un valor a una variable de puerto de entrada. Si asigna un valor a una variable de entrada en la ficha **En entrada**, no se pueden obtener los datos de entrada para el puerto correspondiente en la fila actual.

- Variables de instancia y métodos definidos por el usuario. Utilice cualquier variable de instancia o estática o método definido por el usuario que haya declarado en la ficha **Aplicaciones auxiliares**.

Por ejemplo, una transformación de Java activa tiene dos puertos de entrada BASE\_SALARY y BONUSES, con un tipo de datos Integer y un puerto de salida individual TOTAL\_COMP, con un tipo de datos Integer. Cree un método definido por el usuario en la ficha **Aplicaciones auxiliares**, myTXAdd, que añada dos enteros y devuelva el resultado. Utilice el siguiente código Java en la ficha **En entrada** para asignar los valores totales de los puertos de entrada al puerto de salida y generar una fila de salida:

```
TOTAL_COMP = myTXAdd (BASE_SALARY, BONUSES);  
generateRow();
```

Cuando la transformación de Java recibe una fila de entrada, añade los valores de los puertos de entrada BASE\_SALARY y BONUSES, asigna el valor al puerto de salida TOTAL\_COMP y genera una fila de salida.

- Métodos API de transformación de Java. Puede invocar los métodos API proporcionados por la transformación de Java.

Para compilar el código de la transformación de Java, haga clic en **Compilar**, en las propiedades **Compilación** de la ficha **En entrada**. Los resultados de la compilación aparecen en la ventana **Resultados** de la ficha **En entrada**.

## Ficha Al final

En la ficha **Al final** se define el comportamiento de una transformación de Java activa o pasiva después de procesar todos los datos de entrada. En esta ficha también puede definir los datos de salida para las transformaciones activas y llamar métodos API de una transformación de Java.

Para definir el comportamiento de una transformación de Java después de que haya procesado todos los datos de entrada, especifique el código en la ventana **Código Java** en las propiedades **Código** de la ficha **Al final**.

Puede acceder a las siguientes variables y definir las para métodos API en la ficha **Al final**:

- Variables de puerto de salida. Puede usar los nombres de cualquier puerto de salida que haya definido como variables en la ficha **Puertos** o definir datos de salida para las transformaciones de Java activas.

- Variables de instancia y métodos definidos por el usuario. Utilice cualquier variable de instancia o método definido por el usuario que haya declarado en la ficha **Aplicaciones auxiliares**.
- Métodos API de transformación de Java. Invoque los métodos API proporcionados por la transformación de Java.

Por ejemplo, utilice el siguiente código Java para escribir información en el registro cuando se alcance el final de los datos:

```
logInfo("Number of null rows for partition is: " + partCountNullRows);
```

Para compilar el código de la transformación de Java, haga clic en **Compilar**, en las propiedades **Compilación** de la ficha **Al final**. Los resultados de la compilación aparecen en la ventana **Resultados** de la ficha **Al final**.

## Ficha Funciones

En la ficha **Funciones** se definen funciones que invocan expresiones en una transformación de Java con el lenguaje de programación Java.

Por ejemplo, puede definir una función que invoque una expresión para consultar los valores de los puertos de entrada y de salida o para buscar los valores de las variables de la transformación de Java.

Para definir una función, puede definir funciones manualmente en la ventana **Código Java**, en las propiedades **Código** de la ficha **Funciones**, o puede hacer clic en **Nueva función** para abrir el cuadro de diálogo **Definir función**, que permite definir fácilmente una función.

Para compilar el código, haga clic en **Compilar**, en las propiedades **Compilación** de la ficha **Funciones**. Los resultados de la compilación aparecen en la ventana **Resultados** de la ficha **Funciones**.

## Ficha Código completo

En la ficha **Código completo** puede ver, aunque no editar, el código completo de la clase para la transformación de Java, y compilar el código.

Puede ver el código completo de la clase en la ventana **Código Java**, en las propiedades **Código**.

Para compilar el código completo de la transformación de Java, haga clic en **Compilar**, en las propiedades **Compilación** de la ficha **Código completo**. Los resultados de la compilación aparecen en la ventana **Resultados** de la ficha **Código completo**.

# Optimización de filtro con la transformación de Java

El servicio de integración de datos puede aplicar la optimización del filtro para las transformaciones de Java activas. Cuando defina la transformación de Java, puede añadir código para la optimización del filtro en la ficha **Interfaces del optimizador** de la transformación de Java.

## Optimización de primera selección con la transformación de Java

Puede habilitar una transformación de Java activa o pasiva para la optimización de primera selección si la transformación Java no tiene ningún efecto secundario. El optimizador pasa la lógica de filtro a través de la transformación de Java y modifica la condición de filtro según sea necesario.

Para ver los fragmentos de código para la optimización de primera selección, seleccione PredicatePushOptimization en el navegador de la ficha **Interfaces del optimizador**.

### allowPredicatePush

Booleano. Habilita la primera selección. Cambie la función para devolver un resultado TRUE y el mensaje para poder habilitar la primera selección. El valor predeterminado es FALSE y la función devuelve un mensaje que no es compatible con la optimización.

```
public ResultAndMessage allowPredicatePush(boolean ignoreOrderOfOp) {
    // To Enable PredicatePushOptimization, this function should return true
    //return new ResultAndMessage(true, "");
    return new ResultAndMessage(false, "Predicate Push Optimization Is Not Supported");
}
```

### canGenerateOutputFieldEvalError

Booleano. Indica si la transformación de Java puede devolver o no un error de campo de salida, tal como un error de dividir por cero. Cambie la función devolver un resultado FALSE si la transformación de Java no genera errores del campo de salida. Cuando la transformación de Java puede generar errores de campos, el servicio de integración de datos no puede utilizar la optimización de primera selección.

```
public boolean canGenerateOutputFieldEvalError() {
    // If this Java transformation can never generate an output field evaluation error,
    // return false.
    return true;
}
```

### getInputExpr

Devuelve una expresión de Informatica que describe qué valores de entrada de los campos de entrada tienen un campo de salida. El optimizador necesita saber qué campos de entrada tienen un campo de salida para insertar la lógica de filtro a la transformación.

```
public InfaExpression getInputExpr(TransformationField field,
    TransformationDataInterface group) {
    // This should return an Informatica expression for output fields in terms of input
    fields
    // We will only push predicate that use fields for which input expressions are
    defined.
    // For example, if you have two input fields in0 and in1 and three output fields
    out0, out1, out2
    // out0 is the pass-through of in1, out2 is sum of in1 and in2, and out3 is
    unknown, the code should be:
    //if (field.getName().equals("out0"))
    //    return new InfaExpression("in0", instance);
    //else if (field.getName().equals("out1"))
    //    return new InfaExpression("in0 + in1", instance);
    //else if (field.getName().equals("out2"))
    //    return null;
}
```

```

        return null;
    }

```

Por ejemplo, una asignación contiene una expresión de filtro, "out0 > 8". Out0 es el valor del puerto de salida de out0 en la transformación de Java. Puede definir el valor de out0 como el valor del puerto de entrada in0 + 5. El optimizador puede insertar la siguiente expresión "(in0 + 5) > 8" más allá de la transformación de Java con la optimización de primera selección. Puede devolver NULL si un campo de salida no tiene una expresión de campo de entrada. El optimizador no inserta expresiones de filtro después de los campos de salida sin expresiones de entrada.

Puede incluir el siguiente código:

```

if (field.getName().equals("out0"))
    return new InfaExpression("in0 + 5", instance);
else if (field.getName().equals("out2"))
    return null;

```

## inputGroupsPushPredicateTo

Devuelve una lista de grupos que pueden recibir la lógica de filtro. La transformación de Java tiene un grupo de entrada. No modifique esta función para la transformación de Java.

```

public List<TransformationDataInterface> inputGroupsPushPredicateTo(
    List<TransformationField> fields) {
    // This functions returns a list of input data interfaces to push predicates to.
    // Since JavaTx only has one input data interface, you should not have to modify
    this function
    AbstractTransformation tx = instance.getTransformation();
    List<DataInterface> dis = tx.getDataInterfaces();
    List<TransformationDataInterface> inputDIs = new
    ArrayList<TransformationDataInterface>();
    for (DataInterface di : dis){
        TransformationDataInterface tdi = (TransformationDataInterface) di;
        if (tdi.isInput())
            inputDIs.add(tdi);
    }
    if(inputDIs.size() == 1)
        return inputDIs;
    else
        return null;
}

```

## Optimización de inserción con la transformación de Java

Puede habilitar una transformación de Java activa para la optimización de inserción si no tiene ningún efecto secundario y la optimización no afecta a los resultados de la asignación.

Cuando configure la optimización de inserción para la transformación de Java, puede definir una forma para que la transformación de Java almacene la condición de filtro que recibe desde el optimizador. Añada el código que examina la condición de filtro. Si la transformación de Java puede absorber la lógica de filtro, entonces la transformación de Java devuelve una condición TRUE al optimizador. El optimizador elimina la transformación de filtro de la asignación optimizada.

Cuando configure la transformación de Java escriba el código que almacena la condición de filtro como metadatos de transformación durante la optimización. También puede escribir el código para recuperar la condición de filtro en tiempo de ejecución y para quitar las filas según la lógica de filtro.

Cuando defina la transformación de Java, puede añadir código para la optimización de inserción en la ficha **Interfaces del optimizador** de la transformación de Java. Para acceder a los fragmentos de código para la optimización de inserción, seleccione FilterPushdownOptimization en el navegador de la ficha **Interfaces del optimizador** de la transformación.

Developer Tool muestra los fragmentos de código para habilitar la optimización de inserción y para recibir la condición de filtro del optimizador. Actualice los fragmentos de código para habilitar la optimización y para guardar la lógica de filtro como metadatos de transformación.

### isFilterSupported

Devuelve TRUE para habilitar la optimización de inserción. Devuelve FALSE para deshabilitar la optimización de inserción.

Cambie la función para que devuelva TRUE para poder habilitar la optimización de inserción.

```
public ResultAndMessage isFilterSupported() {
    // To enable filter push-into optimization this function should return true
    // return new ResultAndMessage(true, "");
    return new ResultAndMessage(false, "Filter push-into optimization is not
supported");
}
```

### pushFilter

Recibe la condición de filtro del optimizador.

Añada el código para examinar el filtro y determinar si la lógica de filtro puede utilizarse en la transformación. Si la transformación puede absorber el filtro, entonces use el siguiente método para almacenar la condición de filtro como metadatos de transformación:

```
storeMetadata(String key, String data)
```

La clave es un identificador de los metadatos. Puede definir cualquier cadena como clave. Los datos son los datos que desea almacenar en orden para determinar qué filas soltar en tiempo de ejecución. Por ejemplo, los datos pueden ser la condición de filtro que la transformación de Java recibe desde el optimizador.

```
public ResultAndMessage pushFilter(InfaExpression condition) {
    // Add code to absorb the filter
    // If filter is successfully absorbed return new ResultAndMessage(true, ""); and
the optimizer
    // will remove the filter from the mapping
    // If the filter is not absorbed, return new ResultAndMessage(false, msg);
    return new ResultAndMessage(false, "Filter push-into optimization is not
supported");
}
```

## Cómo crear una transformación de Java

En Developer Tool, puede crear una transformación de Java reutilizable o no reutilizable.

### Cómo crear una transformación de Java reutilizable

Puede haber transformaciones reutilizables en varias asignaciones.

Cree una transformación de Java reutilizable en Developer Tool.

1. Seleccione un proyecto o una carpeta en la vista **Explorador de objetos**.
2. Haga clic en **Archivo > Nueva > Transformación**.  
A continuación, aparece el cuadro de diálogo **Nueva**.
3. Seleccione la transformación de Java.
4. Haga clic en **Siguiente**.

5. Especifique un nombre para la transformación.
6. Para crear una transformación activa, seleccione la opción **Crear como activa**.
7. Haga clic en **Finalizar**.  
La transformación aparece en el editor.
8. En la vista **Puertos**, haga clic en el botón **Nuevo** para añadir un puerto a la transformación.
9. Edite el puerto para definir el nombre, el tipo de datos y la precisión.  
Utilice nombres de puerto como variables en los snippets de código Java.
10. En la vista **Java**, utilice las fichas de introducción de código para escribir y compilar el código Java de la transformación.
11. En la vista **Java**, utilice la ficha **Funciones** para definir funciones que invoquen expresiones.
12. En cualquier ficha de introducción de código, haga doble clic en los mensajes de error que aparecen en la ventana **Resultados**, en las propiedades **Compilación** para localizar y solucionar errores de compilación del código Java de la transformación.
13. En la vista **Avanzadas**, edite las propiedades de la transformación.

## Cómo crear una transformación de Java no reutilizable

Las transformaciones no reutilizables existen en una única asignación.

Cree una transformación de Java no reutilizable en Developer Tool.

1. En una asignación o en un mapplet, arrastre una transformación de Java desde la paleta Transformación hasta el editor.
2. En el cuadro de diálogo **Nueva transformación de Java**, especifique un nombre para la transformación.
3. Para crear una transformación activa, seleccione la opción **Crear como activo**.
4. Haga clic en **Finalizar**.  
La transformación aparece en el editor.
5. En la ficha **General**, edite el nombre y la descripción de la transformación.
6. En la ficha **Puertos**, haga clic en el botón **Nuevo** para añadir puertos a la transformación.
7. Edite el puerto para definir el nombre, el tipo de datos y la precisión.  
Utilice nombres de puerto como variables en los snippets de código Java.
8. En la vista **Java**, utilice las fichas de introducción de código para escribir y compilar el código Java de la transformación.
9. En la vista **Java**, utilice la ficha **Funciones** para definir funciones que invoquen expresiones.
10. En cualquier ficha de introducción de código, haga doble clic en los mensajes de error que aparecen en la ventana **Resultados**, en las propiedades **Compilación** para localizar y solucionar errores de compilación del código Java de la transformación.
11. En la vista **Avanzadas**, edite las propiedades de la transformación.

## Cómo compilar una transformación de Java

cliente de PowerCenter utiliza el compilador de Java para compilar el código Java y generar el código byte para la transformación.

El compilador de Java compila el código Java y muestra los resultados de la compilación en la ventana **Resultados** en las propiedades **Compilación** en las fichas de introducción de código. El compilador de Java se instala junto con el Developer Tool en el directorio `java/bin`.

Para compilar el código completo de la transformación de Java, haga clic en **Compilar**, en las propiedades **Compilación**, en la ficha **Código completo**.

Cuando se crea una transformación de Java, ésta contiene una clase de Java que define las funciones básicas de la transformación de Java. El código completo de la clase de Java contiene la clase de plantilla para la transformación, además del código Java que se define en las fichas de introducción de código.

Cuando se compila una transformación de Java, el Developer Tool añade el código desde las fichas de introducción de código para la transformación para generar el código de clase completo para la transformación. Luego, el Developer Tool invoca el compilador de Java para compilar el código de clase completo. El compilador de Java compila la transformación y genera el código byte para la transformación.

Los resultados de la compilación se muestran en la ventana **Resultados**. Utilice los resultados de la compilación para identificar y localizar errores del código Java.

## Solución de problemas de una transformación de Java

En la ventana **Resultados** de las propiedades **Compilación** de cualquier ficha de introducción de código, puede encontrar y solucionar los errores del código Java.

Se pueden producir errores en una transformación de Java debido a un error del código en una ficha de introducción de código o en el código completo de la clase de la transformación de Java.

Para solucionar los problemas de una transformación de Java, lleve a cabo los siguientes pasos de alto nivel:

1. Busque la causa del error en el código del snippet de Java en el código completo de la clase de la transformación.
2. Identifique el tipo de error. Utilice los resultados de la compilación de la ventana **Resultados** y la ubicación del error para identificar el tipo de error.
3. Repare el código Java en la ficha de introducción de código.
4. Vuelva a compilar la transformación.

## Cómo buscar el origen de errores de compilación

Para buscar el origen de errores de compilación, utilice los resultados de la compilación que se muestran en la ventana **Resultados** de las propiedades **Compilación** en una ficha de introducción de código o en la ficha **Código completo**.

Cuando se hace doble clic en un mensaje de error, en la ventana **Resultados**, el código de origen que ha causado el error aparece resaltado en la ventana **Código Java**, en la ficha de introducción de código o en la ficha **Código completo**.

Se pueden buscar errores en la ficha **Código completo**, aunque no puede editar el código Java en la ficha **Código completo**. Para corregir los errores que aparecen en la ficha de introducción de código **Código completo**, modifique el código que aparece en la ficha de introducción de código apropiada. Quizás necesite utilizar la ficha **Código completo** para ver los errores causados al añadir código de usuario al código de clase completo de la transformación.

## Cómo encontrar un error en una ficha de introducción de código o en la ficha Código completo

Puede encontrar errores de compilación en una ficha de introducción de código o en la ficha **Código completo**.

1. En la ventana **Resultados** de las propiedades **Compilación** de cualquier ficha de introducción de código o en la ficha **Código completo**, haga doble clic en un mensaje de error.
2. Haga clic en **Mostrar en > Snippet** o en **Mostrar en > Ficha Código completo**.

Developer Tool resalta el origen del error en la ficha seleccionada.

**Nota:** Puede ver, aunque no corregir, errores en la ficha **Código completo**. Para corregir los errores debe ir hasta la ficha de introducción de código apropiada.

## Cómo identificar el origen de errores de compilación

Los errores de compilación pueden aparecer como resultado de errores en el código escrito por el usuario.

Los errores en el código escrito por el usuario también pueden generar un error en el código no escrito por el usuario en la clase. Los errores de compilación se producen en el código, escrito o no por el usuario, para la transformación de Java.

### Errores de código de usuario

Puede haber errores en el código de usuario en las fichas de introducción de código. Los errores de código de usuario incluyen errores estándar de sintaxis Java y de lenguaje.

También puede haber errores de código de usuario cuando el Developer Tool añade el código del usuario desde las fichas de introducción de código al código de clase completo.

Por ejemplo, una transformación de Java tiene un puerto de entrada con el nombre `int1` y un tipo de datos `Integer`. El código completo de la clase declara la variable de puerto de entrada con el siguiente código:

```
int int1;
```

Sin embargo, si se utiliza el mismo nombre de variable en la ficha **En entrada**, el compilador de Java emite un error para una redeclaración de una variable. Para corregir el error, cambie el nombre de la variable en la ficha **En entrada**.

### Errores ajenos al código de usuario

El código de usuario en las fichas de introducción de código puede causar errores en el código no creado por el usuario.

Por ejemplo, una transformación de Java tiene un puerto de entrada y un puerto de salida con los nombres `int1` y `out1` con tipos de datos `Integer`. Escriba el siguiente código en la ficha de introducción de código **En entrada** para calcular el interés para el puerto de entrada `int1` y asignarlo al puerto de salida `out1`:

```
int interest;
interest = CallInterest(int1); // calculate interest
out1 = int1 + interest;
}
```

Cuando se compila la transformación, el Developer Tool añade el código desde la ficha de introducción de código **En entrada** al código de clase completo para la transformación. Cuando el compilador de Java compila el código Java, el corchete no cerrado hace que un método del código de clase completo finalice prematuramente y el compilador de Java emite un error.

## CAPÍTULO 16

# Referencia API de transformación de Java

Este capítulo incluye los siguientes temas:

- [Resumen de los métodos API de transformación de Java, 189](#)
- [defineJExpression, 190](#)
- [failSession, 191](#)
- [generateRow, 192](#)
- [getInRowType, 192](#)
- [getMetadata, 193](#)
- [incrementErrorCount, 193](#)
- [invokeJExpression, 194](#)
- [isNull, 195](#)
- [logError, 196](#)
- [logInfo, 196](#)
- [resetNotification, 197](#)
- [setNull, 197](#)
- [storeMetadata, 198](#)

## Resumen de los métodos API de transformación de Java

En las fichas de introducción de código de la vista **Java** del editor, puede añadir métodos API al código Java para definir el comportamiento de la transformación.

Para añadir un método API al código, expanda la lista **API invocables** en el navegador de la ficha de introducción de código, y luego haga doble clic en el nombre del método que desee añadir al código.

También puede arrastrar el método desde el navegador hasta el snippet de código Java o especificar manualmente el método API en el snippet de código Java.

Puede añadir los siguientes métodos API al código Java en una transformación de Java:

**defineJExpression**

Define la expresión de Java.

**failSession**

Devuelve una excepción con un mensaje de error y la de sesión genera un error.

**generateRow**

Genera una fila de salida para las transformaciones de Java activas.

**getInRowType**

Devuelve el tipo de entrada de la fila actual en la transformación.

**incrementErrorCount**

Incrementa el recuento de errores para la de sesión.

**invokeJExpression**

Invoca una expresión de Java definida mediante el método defineJExpression.

**isNull**

Comprueba si hay un valor nulo en una columna de entrada.

**logError**

Escribe un mensaje de error en el de sesión.

**logInfo**

Escribe un mensaje informativo en registro de .

**resetNotification**

Si el equipo del servicio de integración de datos se ejecuta en modo de reinicio, restablece variables que utiliza en el código Java después de la ejecución de una asignación.

**setNull**

Establece como nulo el valor de una columna de salida en una transformación de Java activa o pasiva.

## defineJExpression

Define una expresión, incluyendo la cadena de la expresión y los parámetros de entrada. Los argumentos para el método defineJExpression incluyen una matriz de objetos JExprParamMetadata que contiene los parámetros de entrada y un valor de cadena que define la sintaxis de la expresión.

Utilice la siguiente sintaxis:

```
defineJExpression(  
    String expression,  
    Object[] paramMetadataArray  
);
```

La tabla siguiente describe los parámetros:

Parámetro	Tipo	Tipo de datos	Descripción
expresión	Entrada	String	Cadena que representa la expresión.
paramMetadataArray	Entrada	Object[]	Matriz de objetos JExprParamMetadata que contiene los parámetros de entrada para la expresión.

Se puede añadir el método `defineExpression` al código Java en cualquier ficha de entrada de código, excepto en las fichas **Importaciones** y **Funciones**.

Para utilizar el método `defineJExpression`, debe crear una instancia de una matriz de objetos `JExprParamMetadata` que representan los parámetros de entrada de la expresión. Defina los valores de metadatos para los parámetros y pase la matriz en forma de parámetro al método `defineJExpression`.

Por ejemplo, el siguiente código Java crea una expresión para buscar el valor de dos cadenas:

```
JExprParamMetadata params[] = new JExprParamMetadata[2];
params[0] = new JExprParamMetadata(EDataType.STRING, 20, 0);
params[1] = new JExprParamMetadata(EDataType.STRING, 20, 0);
defineJExpression(":lkp.mylookup(x1,x2)",params);
```

**Nota:** Debe numerar los parámetros que se pasan a la expresión de forma consecutiva, y comenzar los parámetros con la letra x. Por ejemplo, para pasar tres parámetros a una expresión, nombre los parámetros x1, x2 y x3.

## failSession

Devuelve una excepción con un mensaje de error y genera un fallo en la de la sesión.

Utilice la siguiente sintaxis:

```
failSession(String errorMessage);
```

La tabla siguiente describe el parámetro:

Parámetro	Tipo de parámetro	Tipo de datos	Descripción
errorMessage	Entrada	String	Cadena del mensaje de error

Utilice el método `failSession` para finalizar la de la sesión. No utilice el método `failSession` en un bloque `try/catch` en una ficha de introducción de código.

Se puede añadir el método `failSession` al código Java en cualquier ficha de entrada de código, excepto en las fichas **Importaciones** y **Funciones**.

El siguiente código Java muestra cómo probar el puerto `input1` para un valor nulo y hacer fallar la de la sesión si es nulo:

```
if(isNull("input1")) {
    failSession("Cannot process a null value for port input1.");
}
```

## generateRow

Genera una fila de salida para transformaciones de Java activas.

Utilice la siguiente sintaxis:

```
generateRow();
```

Cuando se llama al método generateRow, la transformación de Java genera una fila de salida utilizando el valor actual de las variables del puerto de salida. Si se desea generar varias filas correspondientes a una fila de entrada, puede llamar al método generateRow más de una vez para cada fila de entrada. Si no se utiliza el método generateRow en una transformación de Java activa, la transformación no generará filas de salida.

Se puede añadir el método generateRow al código Java en cualquier ficha de introducción de código, excepto en las fichas **Importaciones y Funciones**.

Solamente puede llamar al método generateRow en transformaciones activas. Si se llama al método generateRow en una transformación pasiva, la servicio de integración de datos genera un error.

Utilice el siguiente código Java para generar una fila de salida, modificar los valores de los puertos de salida y generar otra fila de salida:

```
// Generate multiple rows.
if(!isNull("input1") && !isNull("input2"))
{
    output1 = input1 + input2;
    output2 = input1 - input2;
}
generateRow();
// Generate another row with modified values.
output1 = output1 * 2;
output2 = output2 * 2;
generateRow();
```

## getInRowType

Devuelve el tipo de entrada de la fila actual en la transformación. El método devuelve un valor de inserción, actualización, eliminación o rechazo.

Utilice la siguiente sintaxis:

```
rowType getInRowType();
```

La tabla siguiente describe el parámetro:

Parámetro	Tipo de parámetro	Tipo de datos	Descripción
rowType	Salida	String	Devuelve el tipo de estrategia de actualización, que es uno de los siguientes valores: <ul style="list-style-type: none"><li>- DELETE</li><li>- INSERT</li><li>- REJECT</li><li>- UPDATE</li></ul>

Puede añadir el método getInRowType al código Java en la ficha de introducción de código **En entrada**.

Puede utilizar el método `getInRowType` en transformaciones activas configuradas para definir la estrategia de actualización. Si se invoca este método en una transformación activa que no está configurada para definir la estrategia de actualización, el de la sesión genera un error.

## getMetadata

Recupera los metadatos de la transformación de Java en tiempo de ejecución. El método `getMetadata` recupera los metadatos que haya guardado con el método `storeMetadata`, como una condición de filtro que el optimizador pasa a la transformación de Java en la función `pushFilter`.

Utilice la siguiente sintaxis:

```
getMetadata (String key);
```

La tabla siguiente describe los parámetros:

Parámetro	Tipo de parámetro	Tipo de datos	Descripción
clave	Entrada	String	Identifica los metadatos. El método <code>getMetadata</code> emplea la clave para determinar qué metadatos recuperar.

Puede añadir el método `getMetadata` al código Java en las siguientes fichas de introducción de código:

- Aplicación auxiliar
- En entrada
- Al final
- Interfaces del optimizador
- Funciones

Puede configurar el método `getMetadata` para recuperar las condiciones de filtro para la optimización de inserción. El método `getMetadata` puede recuperar cada condición de filtro que haya almacenado desde el optimizador.

```
// Retrieve a filter condition
String mydata = getMetadata ("FilterKey");
```

## incrementErrorCount

Incrementa el recuento de errores. Si el recuento de errores alcanza el umbral de error , la de la sesión genera un error.

Utilice la siguiente sintaxis:

```
incrementErrorCount(int nErrors);
```

La tabla siguiente describe el parámetro:

Parámetro	Tipo de parámetro	Tipo de datos	Descripción
Número de errores	Entrada	Integer	Número por el cual incrementar el recuento de errores.

Se puede añadir el método `incrementErrorCount` al código Java en cualquier ficha de introducción de código, excepto en las fichas **Importaciones** y **Funciones**.

El siguiente código Java muestra cómo incrementar el recuento de errores si el puerto de entrada de una transformación tiene un valor nulo:

```
// Check if input employee id and name is null.
if (isNull ("EMP_ID_INP") || isNull ("EMP_NAME_INP"))
{
    incrementErrorCount(1);
    // if input employee id and/or name is null, don't generate a output row for this
    input row
    generateRow = false;
}
```

## invokeJExpression

Invoca una expresión y devuelve el valor para la expresión.

Utilice la siguiente sintaxis:

```
(datatype) invokeJExpression(
    String expression,
    Object[] paramMetadataArray);
```

Los parámetros de entrada para el método `invokeJExpression` son un valor de cadena que representa la expresión y una matriz de objetos que contienen parámetros de entrada de expresión.

La tabla siguiente describe los parámetros:

Parámetro	Tipo de parámetro	Tipo de datos	Descripción
Expresión	Entrada	String	Cadena que representa la expresión.
paramMetadataArray	Entrada	Object[]	Matriz de objetos que contiene los parámetros de entrada para la expresión.

Se puede añadir el método `invokeJExpression` al código Java en cualquier ficha de introducción de código, excepto las fichas **Importaciones** y **Funciones**.

Tenga en cuenta las siguientes reglas y directrices cuando utilice el método `invokeJExpression`:

- Tipo de datos de devolución. El tipo de datos de devolución del método `invokeJExpression` es un objeto. Debe lanzar el valor de devolución de la función con el tipo de datos apropiado.  
Puede devolver valores con los tipos de datos Integer, Double, String y byte[].
- Tipo de fila. El tipo de fila para valores de devolución del método `invokeJExpression` es INSERT.

Utilice la interfaz avanzada para utilizar un tipo de fila diferente para el valor de devolución.

- Valores nulos. Si se pasa un valor nulo como parámetro o si el valor de devolución del método `invokeJExpression` es `NULL`, el valor se trata como un indicador nulo.

Por ejemplo, si el valor de devolución de una expresión es `NULL` y el tipo de datos de devolución es `String`, se devuelve una cadena con un valor nulo.

- Tipo de datos `Date`. Debe convertir los parámetros de entrada con un tipo de datos `Date` al tipo de datos `String`.

Para utilizar la cadena en una expresión como tipo de datos `Date`, utilice la función `to_date()` para convertir la cadena a un tipo de datos `Date`.

Asimismo, debe lanzar el tipo de devolución de cualquier expresión que devuelva un tipo de datos `Date` como tipo de datos `String`.

El siguiente ejemplo concatena las cadenas "John" y "Smith" y devuelve la cadena "John Smith":

```
(String)invokeJExpression("concat(x1,x2)", new Object [] { "John ", "Smith" });
```

**Nota:** Debe numerar los parámetros que pasa a la expresión de forma consecutiva e iniciar el parámetro con la letra x. Por ejemplo, para pasar tres parámetros a una expresión, nombre los parámetros x1, x2 y x3.

## isNull

Comprueba el valor de una columna de entrada para un valor nulo.

Utilice la siguiente sintaxis:

```
Boolean isNull(String satrColName);
```

La tabla siguiente describe el parámetro:

Parámetros	Tipo de parámetro	Tipo de datos	Descripción
strColName	Entrada	String	Nombre de una columna de entrada.

Puede añadir el método `isNull` al código Java en la ficha de introducción de código **En entrada**.

El siguiente código Java muestra cómo comprobar si el valor de la columna de entrada `SALARY` es nulo antes de añadirlo a la variable de instancia `totalSalaries`:

```
// if value of SALARY is not null
if (!isNull("SALARY")) {
    // add to totalSalaries
    TOTAL_SALARIES += SALARY;
}
```

También puede utilizar el siguiente código Java para conseguir los mismos resultados:

```
// if value of SALARY is not null
String strColName = "SALARY";
if (!isNull(strColName)) {
    // add to totalSalaries
    TOTAL_SALARIES += SALARY;
}
```

# logError

Escribe un mensaje de error en el registro de la .

Utilice la siguiente sintaxis:

```
logError(String msg);
```

La tabla siguiente describe el parámetro:

Parámetro	Tipo de parámetro	Tipo de datos	Descripción
msg	Entrada	String	Cadena del mensaje de error

Se puede añadir el método logError al código Java en cualquier ficha de introducción de código, excepto en las fichas **Importaciones** y **Funciones**.

El siguiente código Java muestra cómo registrar un error cuando el puerto de entrada es nulo:

```
// check BASE_SALARY
if (isNull("BASE_SALARY")) {
    logError("Cannot process a null salary field.");
}
```

Cuando se ejecuta el código aparece el siguiente mensaje en el registro de la :

```
[JTX_1013] [ERROR] Cannot process a null salary field.
```

# logInfo

Escribe un mensaje informativo en el registro de la .

Utilice la siguiente sintaxis:

```
logInfo(String msg);
```

La tabla siguiente describe el parámetro:

Parámetro	Tipo de parámetro	Tipo de datos	Descripción
msg	Entrada	String	Cadena de mensaje informativo.

Se puede añadir el método logInfo al código Java en cualquier ficha de introducción de código, excepto en las fichas **Importaciones** y **Funciones**.

El siguiente código Java muestra cómo escribir un mensaje en el registro de la después de que una transformación de Java procesa un umbral de mensajes de 1000 filas:

```
if (numRowsProcessed == messageThreshold) {
    logInfo("Processed " + messageThreshold + " rows.");
}
```

## resetNotification

Si el equipo del servicio de integración de datos se ejecuta en modo de reinicio, restablece variables que utiliza en el código Java después de la ejecución de una asignación.

En el modo de reinicio, el servicio de integración de datos no se desinicializa, sino que se restablece tras una solicitud de modo que el servicio de integración de datos pueda procesar la siguiente solicitud.

Para una transformación de Java, utilice el método `resetNotification` para restablecer variables en el código Java después de ejecutar una sesión.

Utilice la siguiente sintaxis:

```
public int resetNotification(IGroup group) {  
    return EStatus.value;  
}
```

La tabla siguiente describe los parámetros:

Parámetro	Tipo de parámetro	Tipo de datos	Descripción
int	Salida	EStatus.value	Valor de devolución, donde <i>value</i> tiene uno de los siguientes valores: <ul style="list-style-type: none"><li>- SUCCESS. Correcto.</li><li>- FAILURE. Anomalía.</li><li>- NOIMPL. No implementado.</li></ul>
group	Entrada	IGroup	El grupo de entrada.

Puede añadir el método `resetNotification` al código Java en la ficha de introducción de código, en la ficha **Aplicaciones auxiliares**.

El método `resetNotification` no aparece en la lista de API invocables.

Por ejemplo, en el supuesto de que el código Java declare una variable estática llamada `out5_static` la inicialice con el valor 1, el siguiente código Java restablece la variable `out5_static` a 1 después de la siguiente ejecución de asignación:

```
public int resetNotification(IGroup group) {  
    out5_static=1;  
    return EStatus.SUCCESS;  
}
```

Este método no es necesario. No obstante, si el servicio de integración de datos se ejecuta en modo de reinicio y la asignación contiene una transformación de Java que no implementa el método `resetNotification`, el mensaje de advertencia `JSDK_42075` aparecerá en el registro.

## setNull

Establece el valor de una columna de salida como nulo en una transformación de Java activa o pasiva.

Utilice la siguiente sintaxis:

```
setNull(String strColName);
```

La tabla siguiente describe el parámetro:

Parámetro	Tipo de parámetro	Tipo de datos	Descripción
strColName	Entrada	String	Nombre de una columna de salida.

El método `setNull` establece el valor de una columna de salida como nulo en una transformación de Java activa o pasiva. Después de establecer una columna de salida como nula, no se puede modificar el valor hasta que no se haya generado una fila de salida.

Se puede añadir el método `setNull` al código Java en cualquier ficha de introducción de código, excepto en las fichas **Importaciones** y **Funciones**.

El siguiente código Java muestra cómo comprobar el valor de una columna de entrada y definir como nulo el valor correspondiente de una columna de salida:

```
// check value of Q3RESULTS input column
if(isNull("Q3RESULTS")) {
    // set the value of output column to null
    setNull("RESULTS");
}
```

También puede utilizar el siguiente código Java para conseguir los mismos resultados:

```
// check value of Q3RESULTS input column
String strColName = "Q3RESULTS";
if(isNull(strColName)) {
    // set the value of output column to null
    setNull(strColName);
}
```

## storeMetadata

Almacena los metadatos de la transformación de Java que puede recuperar en tiempo de ejecución con el método `getMetadata`.

Utilice la siguiente sintaxis:

```
storeMetadata (String key String data);
```

La tabla siguiente describe los parámetros:

Parámetro	Tipo de parámetro	Tipo de datos	Descripción
clave	Entrada	String	Identifica los metadatos. El método <code>storeMetadata</code> requiere una clave para identificar los metadatos. Defina la clave como cualquier cadena.
datos	Entrada	String	Los datos que desee almacenar como metadatos de transformación de Java.

Puede añadir el método `storeMetadata` al código Java a las siguientes fichas de introducción de código:

- Aplicación auxiliar
- En entrada

- Al final
- Interfaces del optimizador
- Funciones

Puede configurar el método `storeMetadata` en una transformación activa para aceptar las condiciones de filtro para la optimización de inserción. El método `storeMetadata` almacena una condición de filtro que el optimizador inserta desde la asignación a la transformación de Java.

```
// Store a filter condition  
storeMetadata ("FilterKey", condition);
```

## CAPÍTULO 17

# Expresiones de Java

Este capítulo incluye los siguientes temas:

- [Resumen de las expresiones de Java, 200](#)
- [Cómo utilizar el cuadro de diálogo Definir función para definir una expresión, 201](#)
- [Cómo trabajar con la interfaz simple, 203](#)
- [Cómo trabajar con la interfaz avanzada, 205](#)
- [Referencia API de la clase JExpression, 209](#)

## Resumen de las expresiones de Java

Puede invocar expresiones de en una transformación de Java mediante el lenguaje de programación Java.

Utilice expresiones para ampliar las funciones de una transformación de Java. Por ejemplo, puede invocar una expresión en una transformación de Java para buscar los valores de puertos de entrada y de salida, o buscar los valores de variables de una transformación de Java.

Para invocar expresiones en una transformación de Java, genere el código Java o utilice métodos API de transformación de Java para invocar la expresión. Invoque la expresión y utilice el resultado de la expresión en la ficha de introducción de código apropiada. Puede generar el código Java que invoque una expresión o utilizar métodos API para escribir el código Java que invoque la expresión.

La siguiente tabla describe los métodos que se pueden usar para crear e invocar expresiones en una transformación de Java:

Método	Descripción
Definir función	Permite crear una función que invoque una expresión y genere el código para una expresión.
Interfaz simple	Permite invocar un método API individual para invocar una expresión y obtener el resultado de la expresión.
Interfaz avanzada	Permite definir la expresión, invocar la expresión y utilizar el resultado de la expresión. Si está familiarizado con la programación orientada a objetos y desea ejercer un mayor control en la invocación de la expresión, utilice la interfaz avanzada.

## Tipos de función de expresión

Puede crear expresiones para una transformación de Java usando el cuadro de diálogo **Definir función**, o utilizando la interfaz simple o la avanzada.

Puede especificar como parámetros de entrada expresiones que utilizan variables de puerto de entrada o de salida, o variables en el código Java.

Si utiliza el cuadro de diálogo **Definir función**, puede validar la expresión antes de utilizarla en una transformación de Java.

Puede invocar los siguientes tipos de funciones de expresión en una transformación de Java:

Tipo de función de expresión	Descripción
Funciones de lenguaje de transformación	Funciones similares a SQL, diseñadas para manejar expresiones comunes.
Funciones definidas por el usuario	Funciones que se crean en Developer Tool basadas en funciones del lenguaje de transformación.
Funciones personalizadas	Funciones que se crean con la API Custom Function.

También puede usar transformaciones no conectadas y variables incorporadas. Por ejemplo, puede utilizar una transformación de búsqueda no conectada en una expresión.

## Cómo utilizar el cuadro de diálogo Definir función para definir una expresión

Cuando se define una expresión de Java se configura la función, se crea la expresión, y se genera el código que invoca la expresión.

Puede definir la función y crear la expresión en el cuadro de diálogo **Definir función**.

Para crear una función de expresión y utilizar la expresión en una transformación de Java, realice las siguientes tareas de alto nivel:

1. Configure la función que invoca la expresión, incluyendo el nombre, descripción y parámetros de la función. Utilice parámetros de función cuando cree la expresión.
2. Cree la sintaxis de la expresión y válidelas.
3. Genere el código Java que invoca la expresión.

Developer pone el código en la ficha de introducción de código **Funciones**.

Después de generar el código Java, invoque la función generada en la ficha de introducción de código apropiada para invocar una expresión u obtener un objeto JExpression, en función de si utiliza la interfaz simple o la avanzada.

**Nota:** Para validar una expresión cuando se crea la expresión debe usar el cuadro de diálogo **Definir función**.

## Paso 1. Configurar la función

Configure el nombre y descripción de la función y los parámetros de entrada de la función de Java que invoca la expresión.

Tenga en cuenta las siguientes reglas y directrices cuando configure la función:

- Utilice un nombre de función único que no esté en conflicto con una función Java existente en la transformación o con palabras clave de Java reservadas.
- Debe configurar el nombre del parámetro, el tipo de datos de Java, la precisión y la escala. Los parámetros de entrada son los valores que se pasan cuando se llama la función del código Java para la transformación.
- Para pasar el tipo de datos Date a una expresión, utilice el tipo de datos String para el parámetro de entrada.

Si una expresión devuelve el tipo de datos Date, puede utilizar el valor de devolución como tipo de datos String en la interfaz simple y el tipo de datos String o Long en la interfaz avanzada.

## Paso 2. Crear y validar la expresión

Cuando cree la expresión, utilice los parámetros configurados para la función.

También puede utilizar funciones de lenguaje de transformación, funciones personalizadas u otras funciones definidas por el usuario en la expresión. Puede crear y validar la expresión en los cuadros de diálogo **Definir función** , .

## Paso 3. Generar código Java para la expresión

Después de configurar la función, sus parámetros, y de definir y validar la expresión, puede generar el código Java que invoca la expresión.

Developer coloca el código Java generado en la ficha de introducción de código **Funciones**. Utilice el código Java generado para llamar las funciones que invocan la expresión en las fichas de introducción de código. Se puede generar código Java simple o avanzado.

Después de generar el código Java que invoca una expresión, no se puede editar la expresión y volver a validarla. Para modificar una expresión después de generar el código se debe volver a crear la expresión.

## Cómo crear una expresión y generar código Java mediante el cuadro de diálogo Definir función

Puede crear una función que invoque una expresión en el cuadro de diálogo **Definir función**.

Realice los siguientes pasos para crear una función que invoque una expresión:

1. En Developer, abra una transformación de Java o cree una nueva transformación de Java.
2. En la ficha **Código Java**, haga clic en **Nueva función**.  
Aparece el cuadro de diálogo **Definir función**.
3. Escriba un nombre de función.
4. Si lo desea, puede escribir una descripción para la expresión.

Puede escribir hasta 2.000 caracteres

5. Cree los argumentos para la función.

Cuando cree los argumentos, configure el nombre, tipo de datos, precisión y escala del argumento.

6. En la ficha **Expresión**, cree una expresión con los argumentos creados.
  7. Para validar la expresión, haga clic en **Validar**.
  8. Si lo desea, escriba la expresión en el cuadro **Expresión**. Luego, haga clic en **Validar** para validar la expresión.
  9. Para generar el código Java mediante la interfaz avanzada, seleccione la opción **Generar código avanzado**. Luego, haga clic en **Generar**.
- Developer genera la función para invocar la expresión en la ficha de introducción de código **Funciones**.

## Plantillas de expresiones de Java

Se puede generar código Java para una expresión utilizando el código Java simple o avanzado para expresiones.

El código Java para expresiones se genera conforme la plantilla de la expresión.

El siguiente ejemplo muestra la plantilla para una expresión de Java generada para código Java simple:

```
Object function_name (Java datatype x1[,
                        Java datatype x2 ...] )
                        throws SDK Exception
{
    return (Object)invokeJExpression( String expression,
                                     new Object [] { x1[, x2, ... ]} );
}
```

El siguiente ejemplo muestra la plantilla para una expresión de Java generada utilizando la interfaz avanzada:

```
JExpression function_name () throws SDKException
{
    JExprParamMetadata params[] = new JExprParamMetadata[number of parameters];
    params[0] = new JExprParamMetadata (
        EDataType.STRING, // data type
        20, // precision
        0 // scale
    );
    ...
    params[number of parameters - 1] = new JExprParamMetadata (
        EDataType.STRING, // data type
        20, // precision
        0 // scale
    );
    ...
    return defineJExpression(String expression,params);
}
```

## Cómo trabajar con la interfaz simple

Utilice el método API de Java `invokeJExpression` para invocar una expresión en la interfaz simple.

### `invokeJExpression`

Invoca una expresión y devuelve el valor para la expresión.

Utilice la siguiente sintaxis:

```
(datatype) invokeJExpression(  
    String expression,  
    Object[] paramMetadataArray);
```

Los parámetros de entrada para el método `invokeJExpression` son un valor de cadena que representa la expresión y una matriz de objetos que contienen parámetros de entrada de expresión.

La tabla siguiente describe los parámetros:

Parámetro	Tipo de parámetro	Tipo de datos	Descripción
expresión	Entrada	String	Cadena que representa la expresión.
paramMetadataArray	Entrada	Object[]	Matriz de objetos que contiene los parámetros de entrada para la expresión.

Se puede añadir el método `invokeJExpression` al código Java en cualquier ficha de introducción de código, excepto las fichas **Importaciones** y **Funciones**.

Tenga en cuenta las siguientes reglas y directrices cuando utilice el método `invokeJExpression`:

- Tipo de datos de devolución. El tipo de datos de devolución del método `invokeJExpression` es un objeto. Debe lanzar el valor de devolución de la función con el tipo de datos apropiado.  
Puede devolver valores con los tipos de datos Integer, Double, String y byte[].
- Tipo de fila. El tipo de fila para valores de devolución del método `invokeJExpression` es INSERT.  
Utilice la interfaz avanzada para utilizar un tipo de fila diferente para el valor de devolución.
- Valores nulos. Si se pasa un valor nulo como parámetro o si el valor de devolución del método `invokeJExpression` es NULL, el valor se trata como un indicador nulo.  
Por ejemplo, si el valor de devolución de una expresión es NULL y el tipo de datos de devolución es String, se devuelve una cadena con un valor nulo.
- Tipo de datos Date. Debe convertir los parámetros de entrada con un tipo de datos Date al tipo de datos String.  
Para utilizar la cadena en una expresión como tipo de datos Date, utilice la función `to_date()` para convertir la cadena a un tipo de datos Date.  
Asimismo, debe lanzar el tipo de devolución de cualquier expresión que devuelva un tipo de datos Date como tipo de datos String.

**Nota:** Debe numerar los parámetros que pasa a la expresión de forma consecutiva e iniciar el parámetro con la letra x. Por ejemplo, para pasar tres parámetros a una expresión, nombre los parámetros x1, x2 y x3.

## Ejemplo de interfaz simple

Puede definir y llamar expresiones que usan el método API `invokeJExpression` en las fichas de introducción de código **Auxiliares** y **En entrada**.

El siguiente ejemplo muestra cómo realizar una búsqueda en los puertos de entrada NAME y ADDRESS de una transformación de Java y asignar el valor de devolución al puerto de salida COMPANY\_NAME.

Escriba el siguiente código en la ficha de introducción de código **En entrada**:

```
COMPANY_NAME = (String) invokeJExpression(":lkp.my_lookup(X1,X2)", new Object []  
{str1 ,str2} );  
generateRow();
```

## Cómo trabajar con la interfaz avanzada

En la interfaz avanzada se pueden usar métodos API orientados al objeto para definir, invocar y obtener el resultado de una expresión.

La siguiente tabla describe las clases y métodos API que están disponibles en la interfaz avanzada:

Clase o método API	Descripción
Clase EDataType	Enumera los tipos de datos para una expresión.
Clase JExprParamMetadata	Contiene los metadatos para cada uno de los parámetros en una expresión. Los metadatos de un parámetro incluyen el tipo de datos, la precisión y la escala.
Método API defineJExpression	Define la expresión. Incluye la cadena y los parámetros de la expresión .
Método API invokeJExpression	Invoca una expresión.
clase JExpression	Contiene los métodos para crear, invocar y obtener los metadatos y obtener el resultado de una expresión, y también para comprobar el tipo de datos de devolución.

## Cómo invocar una expresión con la interfaz avanzada

Utilice la interfaz avanzada para definir, invocar y obtener el resultado de una expresión.

1. En las fichas de introducción de código **Aplicaciones auxiliares** o **En entrada**, cree una instancia de la clase JExprParamMetadata para cada de parámetro para la expresión y defina el valor de los metadatos. Si lo desea, puede crear una instancia del objeto JExprParamMetadata en el método defineJExpression.
2. Utilice el método defineJExpression para obtener el objeto JExpression para la expresión.
3. En la ficha de introducción de código apropiada, invoque la expresión con el método invokeJExpression.
4. Compruebe el resultado del valor de devolución o con el método isResultNull.
5. Puede obtener el tipo de datos del valor de devolución o los metadatos del valor de devolución con los métodos getResultDataType y getResultMetadata.
6. Obtenga el resultado de la expresión utilizando el método API apropiado. Puede utilizar los métodos getInt, getDouble, getStringBuffer y getBytes.

## Reglas y directrices para trabajar con la interfaz avanzada

Cuando se trabaja con las interfaces avanzadas, debe tener en cuenta ciertas reglas y directrices.

Tenga en cuenta las siguientes reglas y directrices:

- Si se pasa un valor nulo como parámetro, o si el resultado de una expresión es nulo, el valor se tratará como un indicador nulo. Por ejemplo, si el resultado de una expresión es nulo y el tipo de datos de devolución es String, se devuelve una cadena con valor nulo. Puede comprobar el resultado de una expresión mediante el método isResultNull.

- Debe convertir parámetros de entrada con el tipo de datos Date al tipo de datos String antes de poder usarlos en una expresión. Para utilizar la cadena en una expresión como tipos de datos Date, utilice la función to\_date() para convertir la cadena a un tipo de datos Date.

Puede obtener el resultado de una expresión que devuelve un tipo de datos Date como tipo de datos String o Long.

Para obtener el resultado de una expresión que devuelve un tipo de datos Date como tipo de datos String, utilice el método getStringBuffer. Para obtener el resultado de una expresión que devuelve un tipo de datos Date como tipo de datos Long, utilice el método getLong.

## Clase EDataType

Enumera los tipos de datos de Java utilizados en las expresiones. Obtiene el tipo de datos de devolución de una expresión o asigna el tipo de datos para un parámetro en el objeto JExprParamMetadata. No es necesario crear una instancia de la clase EDataType.

Las siguiente tabla muestra los valores enumerados para los tipos de datos de Java en expresiones:

Tipo de datos	Valor enumerado
INT	1
DOUBLE	2
STRING	3
BYTE_ARRAY	4
DATE_AS_LONG	5

El siguiente ejemplo de código Java muestra cómo utilizar la clase EDataType para asignar un tipo de datos String a un objeto JExprParamMetadata:

```
JExprParamMetadata params[] = new JExprParamMetadata[2];
params[0] = new JExprParamMetadata (
    EDataType.STRING, // data type
    20, // precision
    0 // scale
);
...
```

## Clase JExprParamMetadata

Crea una instancia de un objeto que representa los parámetros para una expresión y define los metadatos para los parámetros.

Utilice una matriz de objetos JExprParamMetadata como entrada del método defineJExpression para definir los metadatos para los parámetros de entrada. Puede crear una instancia del objeto JExprParamMetadata en la ficha de introducción de código **Funciones** o en defineJExpression.

Utilice la siguiente sintaxis:

```
JExprParamMetadata paramMetadataArray[] = new JExprParamMetadata[numberOfParameters];
paramMetadataArray[0] = new JExprParamMetadata(datatype, precision, scale);
...
paramMetadataArray[numberOfParameters - 1] = new JExprParamMetadata(datatype,
precision, scale);;
```

La tabla siguiente describe los argumentos:

Argumento	Tipo de argumento	Tipo de datos de argumento	Descripción
datatype	Entrada	EDataType	Tipo de datos del parámetro.
precision	Entrada	Entero	Precisión del parámetro.
scale	Entrada	Entero	Escala del parámetro.

Por ejemplo, utilice el siguiente código Java para crear una instancia de dos objetos JExprParamMetadata con tipo de datos String, precisión 20 y escala 0:

```
JExprParamMetadata params[] = new JExprParamMetadata[2];
params[0] = new JExprParamMetadata(EDataType.STRING, 20, 0);
params[1] = new JExprParamMetadata(EDataType.STRING, 20, 0);
return defineJExpression("LKP.LKP_addresslookup(X1,X2)",params);
```

## defineJExpression

Define una expresión, incluyendo la cadena de la expresión y los parámetros de entrada. Los argumentos para el método defineJExpression incluyen una matriz de objetos JExprParamMetadata que contiene los parámetros de entrada y un valor de cadena que define la sintaxis de la expresión.

Utilice la siguiente sintaxis:

```
defineJExpression(
    String expression,
    Object[] paramMetadataArray
);
```

La tabla siguiente describe los parámetros:

Parámetro	Tipo	Tipo de datos	Descripción
Expresión	Entrada	String	Cadena que representa la expresión.
paramMetadataArray	Entrada	Object[]	Matriz de objetos JExprParamMetadata que contiene los parámetros de entrada para la expresión.

Se puede añadir el método defineExpression al código Java en cualquier ficha de entrada de código, excepto en las fichas **Importaciones y Funciones**.

Para utilizar el método defineJExpression, debe crear una instancia de una matriz de objetos JExprParamMetadata que representan los parámetros de entrada de la expresión. Defina los valores de metadatos para los parámetros, y pase la matriz en forma de parámetro al método defineJExpression.

Por ejemplo, el siguiente código Java crea una expresión para buscar el valor de dos cadenas:

```
JExprParamMetadata params[] = new JExprParamMetadata[2];
params[0] = new JExprParamMetadata(EDataType.STRING, 20, 0);
params[1] = new JExprParamMetadata(EDataType.STRING, 20, 0);
defineJExpression("lkp.mylookup(x1,x2)",params);
```

**Nota:** Debe numerar los parámetros que se pasan a la expresión de forma consecutiva y comenzar los parámetros con la letra x. Por ejemplo, para pasar tres parámetros a una expresión, nombre los parámetros x1, x2 y x3.

## Clase JExpression

Contiene métodos que permiten crear e invocar una expresión, devolver el valor de una expresión y comprobar el tipo de datos de devolución.

La siguiente tabla enumera los métodos de la clase JExpression:

Nombre de método	Descripción
invoke	Invoca una expresión.
getResultDataType	Devuelve el tipo de datos del resultado de la expresión.
getResultMetadata	Devuelve los metadatos del resultado de la expresión.
isResultNull	Comprueba el valor del resultado de un resultado de expresión.
getInt	Devuelve el valor del resultado de una expresión como un tipo de datos Integer.
getDouble	Devuelve el valor del resultado de una expresión como un tipo de datos Double.
getStringBuffer	Devuelve el valor del resultado de una expresión como un tipo de datos String.
getBytes	Devuelve el valor del resultado de una expresión como un tipo de datos byte[].

## Ejemplo de interfaz avanzada

Puede utilizar la interfaz avanzada para crear e invocar una expresión de búsqueda en una transformación de Java.

El siguiente código Java de ejemplo muestra cómo crear una función que llama una expresión y cómo invocar la expresión para obtener el valor de devolución. Este ejemplo pasa los valores de dos puertos de entrada con un tipo de datos String, NAME y COMPANY, a la función myLookup. La función myLookup utiliza una expresión de búsqueda para buscar el valor del puerto de salida ADDRESS.

**Nota:** Este ejemplo asume que tiene una transformación de búsqueda no conectada llamada LKP\_addresslookup.

Utilice el siguiente código Java en la ficha **Aplicaciones auxiliares**:

```
JExprParamMetadata addressLookup() throws SDKException
{
    JExprParamMetadata params[] = new JExprParamMetadata[2];
    params[0] = new JExprParamMetadata (
        EDataType.STRING,           // data type
        50,                         // precision
        0                           // scale
    );
    params[1] = new JExprParamMetadata (
        EDataType.STRING,           // data type
        50,                         // precision
        0                           // scale
    );
    return defineJExpression(":LKP.LKP_addresslookup(X1,X2)",params);
}
JExpression lookup = null;
boolean isJExprObjCreated = false;
```

Utilice el siguiente código Java en la ficha **En entrada** para invocar la expresión y devolver el valor del puerto ADDRESS:

```
...
if(!iisJExprObjCreated)
{
    lookup = addressLookup();
    isJExprObjCreated = true;
}
lookup = addressLookup();
lookup.invoke(new Object [] {NAME,COMPANY}, ERowType.INSERT);
EDataType addressDataType = lookup.getResultDataType();
if(addressDataType == EDataType.STRING)
{
    ADDRESS = (lookup.getStringBuffer()).toString();
} else {
    logError("Expression result datatype is incorrect.");
}
...
```

## Referencia API de la clase JExpression

La clase JExpression contiene métodos API que permiten crear e invocar una expresión, devolver el valor de una expresión, y comprobar el tipo de datos de devolución.

La clase JExpression contiene los siguientes métodos API:

- `getBytes`
- `getDouble`
- `getInt`
- `getLong`
- `getResultDataType`
- `getResultMetadata`
- `getStringBuffer`
- `invoke`
- `isResultNull`

### getBytes

Devuelve el valor del resultado de una expresión como un tipo de datos `byte[]`. Obtiene el resultado de una expresión que cifra datos con la función `AES_ENCRYPT`.

Utilice la siguiente sintaxis:

```
objectName.getBytes();
```

Utilice el siguiente código Java para obtener el resultado de una expresión que cifra los datos binarios con la función `AES_ENCRYPT`, donde `JExprEncryptData` es un objeto `JExpression`:

```
byte[] newBytes = JExprEncryptData.getBytes();
```

### getDouble

Devuelve el valor del resultado de una expresión como un tipo de datos `Double`.

Utilice la siguiente sintaxis:

```
objectName.getDouble();
```

Por ejemplo, utilice el siguiente código Java para obtener el resultado de una expresión que devuelve un valor de salario en forma de número doble, donde JExprSalary es un objeto JExpression:

```
double salary = JExprSalary.getDouble();
```

## getInt

Devuelve el valor del resultado de una expresión como un tipo de datos Integer.

Utilice la siguiente sintaxis:

```
objectName.getInt();
```

Por ejemplo, utilice el siguiente código Java para obtener el resultado de una expresión que devuelve el número de ID de un empleado en forma de número entero, donde findEmpID es un objeto JExpression:

```
int empID = findEmpID.getInt();
```

## getLong

Devuelve el valor del resultado de una expresión como un tipo de datos Long. Obtiene el resultado de una expresión como un tipo de datos Date.

Utilice la siguiente sintaxis:

```
objectName.getLong();
```

Utilice el siguiente código Java para obtener el resultado de una expresión que devuelve un valor Date como tipo de datos Long, donde JExprCurrentDate es un objeto JExpression:

```
long currDate = JExprCurrentDate.getLong();
```

## getResultDataType

Devuelve el tipo de datos del resultado de una expresión. Devuelve un valor de EDataType.

Utilice la siguiente sintaxis:

```
objectName.getResultDataType();
```

Utilice el siguiente código Java de ejemplo para invocar una expresión y asignar el tipo de datos del resultado a la variable dataType:

```
myObject.invoke(new Object[] { NAME,COMPANY }, ERowType INSERT);  
EDataType dataType = myObject.getResultDataType();
```

## getResultMetadata

Devuelve los metadatos para el resultado de una expresión. Puede usar getResultMetadata para obtener la precisión, la escala y el tipo de datos del resultado de una expresión. Se pueden asignar los metadatos del valor de devolución de una expresión a un objeto JExprParamMetadata. Utilice los métodos de objeto getScale, getPrecision y getDataType para recuperar los metadatos del resultado.

Utilice la siguiente sintaxis:

```
objectName.getResultMetadata();
```

Utilice el siguiente código Java de ejemplo para asignar la escala, la precisión y el tipo de datos del valor de devolución de myObject a variables:

```
JExprParamMetadata myMetadata = myObject.getResultMetadata();
int scale = myMetadata.getScale();
int prec = myMetadata.getPrecision();
int datatype = myMetadata.getDataType();
```

**Nota:** El método de objeto `getDataType` devuelve el valor entero del tipo de dato, tal y como se enumera en `EDatatype`.

## getStringBuffer

Devuelve el valor del resultado de una expresión como un tipo de datos `String`.

Utilice la siguiente sintaxis:

```
objectName.getStringBuffer();
```

Utilice el siguiente código Java para obtener el resultado de una expresión que devuelve dos cadenas concatenadas, donde `JExprConcat` es un objeto `JExpression`:

```
String result = JExprConcat.getStringBuffer();
```

## invoke

Invoca una expresión. Los argumentos de `invoke` incluyen un objeto que define los parámetros de entrada, así como el tipo de fila. Debe crear una instancia de un objeto `JExpression` antes de poder usar el método `invoke`. Para el tipo de fila, utilice `ERowType.INSERT`, `ERowType.DELETE` y `ERowType.UPDATE`.

Utilice la siguiente sintaxis:

```
objectName.invoke(
    new Object[] { param1[, ... paramN ]},
    rowType
);
```

En la tabla siguiente se describen los argumentos:

Argumento	Tipo de datos	Entrada/ Salida	Descripción
objectName	JExpression	Entrada	Nombre del objeto <code>JExpression</code> .
parámetros	-	Entrada	Matriz de objetos que contiene los valores de entrada para la expresión.

Por ejemplo, cree una función en la ficha de introducción de código **Funciones** llamada `address_lookup()` que devuelva un objeto `JExpression` que represente la expresión. Utilice el siguiente código para invocar la expresión que utiliza los puertos de entrada `NAME` y `COMPANY`:

```
JExpression myObject = address_lookup();
myObject.invoke(new Object[] {NAME,COMPANY }, ERowType.INSERT);
```

## isResultNull

Comprueba el valor del resultado de una expresión.

Utilice la siguiente sintaxis:

```
objectName.isResultNull();
```

Utilice el siguiente código Java de ejemplo para invocar una expresión y asignar el valor de devolución de la misma a la dirección variable si el valor de devolución no es nulo:

```
JExpression myObject = address_lookup();
myObject.invoke(new Object[] { NAME,COMPANY }, ERowType INSERT);
if(!myObject.isResultNull()) {
    String address = myObject.getStringBuffer();
}
```

## CAPÍTULO 18

# Transformación de incorporación

Este capítulo incluye los siguientes temas:

- [Resumen de la transformación de incorporación, 213](#)
- [Propiedades avanzadas de Transformación de combinación, 214](#)
- [Puertos de transformación de incorporación, 214](#)
- [Cómo definir una condición de unión, 215](#)
- [Tipos de unión, 216](#)
- [Entrada ordenada para una transformación de incorporación, 219](#)
- [Cómo unir datos de un mismo origen, 221](#)
- [Cómo bloquear los canales de origen, 223](#)
- [Consejos sobre el rendimiento de la transformación de incorporación, 224](#)
- [Reglas y directrices para una transformación de incorporación, 225](#)

## Resumen de la transformación de incorporación

Utilice la transformación de incorporación para unir datos de origen de dos orígenes heterogéneos relacionados entre sí y que residen en ubicaciones o sistemas de archivos diferentes. También se pueden unir datos de un mismo origen. La transformación de incorporación es una transformación activa.

La transformación de incorporación une orígenes con, por lo menos, una columna coincidente. La transformación de incorporación utiliza una condición que coincide con uno o más pares de columnas que comparten los dos orígenes.

Los dos canales de entrada incluyen un canal principal y un canal de detalle, o una ramificación principal y una ramificación de detalle. El canal maestro finaliza en la transformación de incorporación, mientras que el canal de detalle continúa hacia el destino.

Para unir más de dos orígenes en una asignación, una la salida de la transformación de incorporación con otro canal de origen. Añada transformaciones de incorporación a la asignación hasta que haya unido todos los canales de origen.

# Propiedades avanzadas de Transformación de combinación

Configure las propiedades para ayudarle a determinar cómo el servicio de integración de datos procesa los datos para la transformación de incorporación.

Configure las siguientes propiedades en la ficha **Avanzadas**:

## Tamaño de la memoria caché de datos de unión

Tamaño de la memoria caché de datos para la transformación. El valor predeterminado es "Auto".

## Tamaño de la memoria caché de índice de incorporación

Tamaño de la memoria caché de índice para la transformación. El valor predeterminado es "Auto".

## Directorio de la memoria caché

Directorio local donde el servicio de integración de datos crea los archivos de memoria caché del índice y los archivos de memoria caché de datos. Compruebe que el directorio existe y que tiene suficiente espacio en disco para los archivos de la memoria caché. El directorio puede ser una unidad asignada o montada.

El valor predeterminado es el parámetro del sistema CacheDir.

## Entrada ordenada

Indica que los datos de entrada se ordenan previamente por grupos. Elija Entrada ordenada para unir datos ordenados. Utilizar la entrada ordenada puede aumentar el rendimiento.

## Orden de clasificación principal

Especifica el orden de clasificación para los datos de origen principales. Elija Ascendente si los datos de origen principal están en orden ascendente. Si elige Ascendente, habilite también la entrada ordenada. El valor predeterminado es "Auto".

## Nivel de seguimiento

Cantidad de detalles que aparecen en el registro de esta transformación. Puede elegir simplificado, normal, inicialización detallada o datos detallados. El valor predeterminado es Normal.

# Puertos de transformación de incorporación

Una transformación de incorporación tiene distintos tipos de puerto que determinan cómo el servicio de integración de datos llevará a cabo la unión.

Una transformación de incorporación tiene los siguientes tipos de puerto:

## Principal

Puertos que enlazan con el origen principal en la asignación.

## Detalle

Puertos que enlazan con el origen de detalle en la asignación.

Puede cambiar un puerto principal a puerto de detalle. También puede cambiar un puerto de detalle a puerto principal. Cuando se cambia el tipo de un puerto, se cambia el tipo de puerto de todos los puertos. Por tanto,

cuando se cambia un puerto principal a puerto de detalle, se cambian todos los puertos principales a puertos de detalle y viceversa.

## Cómo definir una condición de unión

La condición de unión contiene puertos de ambos orígenes de entrada que el servicio de integración de datos utiliza para unir dos filas.

En función del tipo de unión seleccionado, el servicio de integración de datos agrega la fila al conjunto de resultados o la descarta. La transformación de incorporación genera conjuntos de resultados en función del tipo unión, de la condición y de los orígenes de datos de entrada.

Antes de definir una condición de unión, compruebe que los orígenes principal y detallado están configurados para un rendimiento óptimo. Durante la ejecución de una asignación, el servicio de integración de datos compara cada fila del origen principal con el detallado. Para mejorar el rendimiento de una transformación de incorporación no ordenada, utilice como origen principal el que tiene menos filas. Para mejorar el rendimiento de una transformación de incorporación no ordenada, utilice como origen principal el que tiene menos valores de clave duplicados.

En la condición de unión, utilice uno o varios puertos de los orígenes de entrada de una transformación de incorporación. Cuantos más puertos haya, mayor será el tiempo necesario para unir dos orígenes. El orden de los puertos de la condición puede afectar al rendimiento de la transformación de incorporación. Si utiliza varios puertos en la condición de unión, el servicio de integración de datos los compara en el orden en el que los especifique.

Si une los tipos de datos Char y Varchar, el servicio de integración de datos cuenta todos los espacios de los valores Char como parte de la cadena:

```
Char(40) = "abcd"  
Varchar(40) = "abcd"
```

El valor Char es "abcd" junto con 36 espacios en blanco; el servicio de integración de datos no unirá los dos campos porque el de Char contiene espacios al final.

**Nota:** La transformación de incorporación no hace coincidir valores null. Por ejemplo, si tanto EMP\_ID1 como EMP\_ID2 contienen una fila que tiene un valor null, el servicio de integración de datos no los considera como una coincidencia y no une ambas filas. Para unir filas que tienen valores null, sustituya estos por valores predeterminados y, a continuación, únalos.

Puede definir un tipo de condición simple o avanzado.

### Tipo de condición simple

Defina un tipo de condición simple para una transformación de incorporación ordenada o no ordenada.

Una condición simple incluye una o varias condiciones que comparan los orígenes principal y detallado especificados. Las condiciones simples deben tener el siguiente formato:

```
<master_port> operator <detail_port>
```

Para las transformaciones de incorporación ordenadas, la condición debe utilizar el operador de igualdad.

En las transformaciones de incorporación no ordenadas, la condición puede utilizar cualquiera de los siguientes operadores: =, !=, >, >=, <, <=.

Por ejemplo, si dos orígenes con tablas llamadas EMPLOYEE\_AGE y EMPLOYEE\_POSITION contienen números de ID de empleado, la siguiente condición hace coincidir las filas con los empleados que se enumeran en ambos orígenes:

```
EMP_ID1 = EMP_ID2
```

Developer Tool valida los tipos de datos en una condición simple. Los dos puertos de la condición deben tener el mismo tipo de datos. Si en la condición necesita utilizar dos puertos con tipos de datos no coincidentes, convierta los tipos de datos para hacerlos coincidir.

## Tipo de condición avanzada

Defina un tipo de condición avanzada para una transformación de incorporación no ordenada.

Una condición avanzada puede incluir cualquier expresión que evalúe un valor numérico o booleano. Una condición avanzada puede incluir cualquiera de los siguientes operadores: =, !=, >, >=, <, <=.

Para introducir una expresión en la condición de combinación, elija el tipo de condición Avanzadas en la ficha **Unión**. Utilice el editor de expresiones para incluir en la condición expresiones, puertos y operadores.

Por ejemplo, imagine que desea unir orígenes de forma que coincida el nombre completo de un empleado. El origen principal incluye un puerto de nombre FirstName y otro de apellido LastName. El origen de detalle incluye un puerto de nombre completo FullName. Defina la siguiente condición para concatenar los puertos principales y hacer coincidir el nombre completo en los dos orígenes:

```
CONCAT(FirstName, LastName) = FullName
```

## Tipos de unión

En una transformación de incorporación, la unión se puede originar a partir de dos tipos de orígenes.

La transformación de incorporación es compatible con los siguientes tipos de unión:

- Normal
- Outer principal
- Outer detallado
- Outer completo

**Nota:** Un outer join normal o principal se ejecuta más rápidamente que una unión outer completo o una outer de detalle.

Si entre los resultados se incluyen campos que no contienen datos en ninguno de los orígenes, la transformación de incorporación los rellena con valores nulos. Si sabe que un campo da un resultado NULL pero no desea insertar ningún NULL en el destino, puede establecer un valor predeterminado para el puerto correspondiente.

## Unión normal

Con una unión normal, el servicio de integración de datos descarta, de acuerdo con la condición, todas las filas de datos del origen principal y el origen de detalle que no coinciden.

Por ejemplo, tenemos dos orígenes de datos para recambios de automóvil llamados PARTS\_SIZE y PARTS\_COLOR.

El origen de los datos de PARTS\_SIZE es el origen principal y contiene los siguientes datos:

PART_ID1	DESCRIPTION	SIZE
1	Seat Cover	Large
2	Ash Tray	Small
3	Floor Mat	Medium

El origen de los datos de PARTS\_COLOR es el origen detallado y contiene los siguientes datos:

PART_ID2	DESCRIPTION	COLOR
1	Seat Cover	Blue
3	Floor Mat	Black
4	Fuzzy Dice	Yellow

Para unir las dos tablas haciendo coincidir PART\_IDs en ambos orígenes, defina la condición del siguiente modo:

```
PART_ID1 = PART_ID2
```

Cuando se unen estas tablas de muestra con una unión normal, el conjunto de resultados incluye los siguientes datos:

PART_ID	DESCRIPTION	SIZE	COLOR
1	Seat Cover	Large	Blue
3	Floor Mat	Medium	Black

El siguiente ejemplo muestra la instrucción SQL equivalente:

```
SELECT * FROM PARTS_SIZE, PARTS_COLOR WHERE PARTS_SIZE.PART_ID1 = PARTS_COLOR.PART_ID2
```

## Outer Join principal

Un outer join principal mantiene todas las filas de datos del origen de detalle y las filas coincidentes del origen principal. Descarta las filas no coincidentes del origen principal.

Cuando se unen las tablas de muestra con un outer join principal y la misma condición, el conjunto de resultados incluye los siguientes datos:

PART_ID	DESCRIPTION	SIZE	COLOR
1	Seat Cover	Large	Blue
3	Floor Mat	Medium	Black
4	Fuzzy Dice	NULL	Yellow

Puesto que no se ha especificado ningún tamaño para Datos de peluche, el servicio de integración de datos llena el campo con NULL.

El siguiente ejemplo muestra la instrucción SQL equivalente:

```
SELECT * FROM PARTS_SIZE RIGHT OUTER JOIN PARTS_COLOR ON (PARTS_COLOR.PART_ID2 =  
PARTS_SIZE.PART_ID1)
```

## Outer Join de detalles

Un outer join de detalles mantiene todas las filas de datos del origen principal y las filas coincidentes del origen de detalle. Descarta las filas no coincidentes del origen de detalle.

Cuando se unen las tablas de muestra con un outer join de detalles y la misma condición, el conjunto de resultados incluye los siguientes datos:

PART_ID	DESCRIPTION	SIZE	COLOR
1	Seat Cover	Large	Blue
2	Ash Tray	Small	NULL
3	Floor Mat	Medium	Black

Puesto que no se ha especificado ningún color para Cenicero, el servicio de integración de datos llena el campo con NULL.

El siguiente ejemplo muestra la instrucción SQL equivalente:

```
SELECT * FROM PARTS_SIZE LEFT OUTER JOIN PARTS_COLOR ON (PARTS_SIZE.PART_ID1 =  
PARTS_COLOR.PART_ID2)
```

## Outer Join completo

Un outer join completo mantiene todas las filas de datos de tanto orígenes principales como orígenes de detalle.

Cuando se unen las tablas de muestra con un outer join completo y la misma condición, el conjunto de resultados incluye los siguientes datos:

PARTED	DESCRIPTION	SIZE	Color
1	Seat Cover	Large	Blue
2	Ash Tray	Small	NULL
3	Floor Mat	Medium	Black
4	Fuzzy Dice	NULL	Yellow

Puesto que no se especifica ningún color para Cenicero y no se especifica ningún tamaño para los Dados de peluche, el servicio de integración de datos llena los campos con NULL.

El siguiente ejemplo muestra la instrucción SQL equivalente:

```
SELECT * FROM PARTS_SIZE FULL OUTER JOIN PARTS_COLOR ON (PARTS_SIZE.PART_ID1 =  
PARTS_COLOR.PART_ID2)
```

# Entrada ordenada para una transformación de incorporación

Puede aumentar el rendimiento de una transformación de incorporación con la opción de entrada ordenada. Utilice la entrada ordenada cuando los datos estén ordenados.

Cuando se configura la transformación de incorporación para utilizar datos ordenados, el servicio de integración de datos aumenta el rendimiento minimizando la entrada y la salida del disco. Verá el mejor rendimiento cuando se trabaja con conjuntos de datos de gran tamaño.

Para configurar una asignación para que utilice datos ordenados, defina y mantenga un orden de clasificación en la asignación para que el servicio de integración de datos pueda utilizar los datos ordenados cuando procese la transformación de incorporación. Lleve a cabo los siguientes pasos para configurar la asignación:

1. Configure el orden de clasificación de los datos que desee unir.
2. Añada transformaciones que mantengan el orden de los datos ordenados.
3. Configure la transformación de incorporación para que utilice datos ordenados y configure la condición de unión para que utilice los puertos de origen de clasificación. El origen de la ordenación representa el origen de los datos ordenados.

## Cómo configurar el orden de clasificación

Configure el orden de clasificación para asegurarse de que el servicio de integración de datos pasa datos ordenados a la transformación de incorporación.

Para configurar el orden de clasificación, utilice uno de los siguientes métodos:

- Utilice archivos sin formato ordenados. Cuando los archivos sin formato contienen datos ordenados, compruebe que el orden de las columnas de ordenación coincide en cada uno de los archivos de origen.
- Utilice datos relacionales ordenados. Utilice puertos ordenados en el objeto de datos relacionales para ordenar columnas de la base de datos de origen. Configure el orden de los puertos ordenados del mismo modo en cada objeto de datos relacionales.
- Utilice una transformación de incorporación para ordenar datos relacionales o de archivo sin formato. Ponga una transformación de incorporación en los canales principal y de detalle. Configure cada transformación de incorporación para que utilice el mismo orden que los puertos clave de ordenación y la misma dirección de ordenación.

Si se pasan datos desordenados u ordenados incorrectamente a una transformación de incorporación configurada para utilizar datos ordenados, la ejecución de la asignación generará un error. El servicio de integración de datos registra el error en el archivo de registro.

## Cómo añadir transformaciones a la asignación

Añada transformaciones a la asignación que mantiene el orden de los datos ordenados en una transformación de incorporación.

Puede poner la transformación de incorporación directamente después del origen de ordenación para mantener los datos ordenados.

Cuando se añaden transformaciones entre el origen de ordenación y la transformación de incorporación, utilice las siguientes directrices para mantener los datos ordenados:

- No ponga ninguna de las siguientes transformaciones entre el origen de ordenación y la transformación de incorporación.
  - Rango
  - Unión
  - Agregación no ordenada
  - Maplet que contiene una de las transformaciones anteriores
- Puede poner una transformación de agregación ordenada entre el origen de ordenación y la transformación de incorporación si utiliza las siguientes directrices:
  - Configure la transformación de agregación para una entrada ordenada.
  - Utilice los mismos puertos para las columnas agrupar por en la transformación de agregación que los puertos en el origen de ordenación.
  - Los puertos agrupar por deben estar en el mismo orden que los puertos en el origen de ordenación.
- Cuando una el conjunto de resultados de una transformación de incorporación con otro canal, compruebe que la salida de datos de la primera transformación de incorporación esté ordenada.

## Reglas y directrices para las condiciones de unión

Algunas reglas y directrices se aplican al crear condiciones de unión para una transformación de incorporación ordenada.

Utilice las siguientes directrices cuando cree condiciones de unión:

- Debe definir un tipo de condición simple que utilice el operador de igualdad.
- Si entre el origen de ordenación y la transformación de incorporación usa una transformación de agregación ordenada, trate esta transformación como si fuera el origen de ordenación cuando defina la condición de unión.
- Los puertos que utiliza en la condición de unión deben coincidir con los puertos del origen de ordenación.
- Cuando configure varias condiciones de unión, los puertos de la primera condición de unión deben coincidir con los primeros puertos del origen de ordenación.
- Si configura varias condiciones, el orden de las condiciones debe coincidir con el orden de los puertos del origen de ordenación, sin omitir ningún puerto.
- El número de los puertos ordenados en el origen de ordenación debe ser superior o igual al número de puertos de la condición de unión.
- Si une puertos con tipos de datos decimales, la precisión de cada puerto debe pertenecer al mismo rango de precisión. Los rangos de precisión válidos son:
  - 0-18 decimal
  - 19-28 decimal
  - 29 decimal y superior

Por ejemplo, si define la condición `DecimalA = DecimalB`, donde `DecimalA` tiene una precisión de 15 y `DecimalB` de 25, la condición no será válida.

## Ejemplo de una condición de unión y del orden de clasificación

Este ejemplo muestra una transformación de incorporación que une los canales principal y de detalle con puertos ordenados.

Las transformaciones de incorporación en los canales maestro y de detalle se configuran con los siguientes puertos ordenados:

- ITEM\_NO
- ITEM\_NAME
- PRICE

Cuando se configure la condición de unión, utilice las siguientes directrices para mantener el orden de clasificación:

- Debe utilizar ITEM\_NO en la primera condición de unión.
- Si añade una segunda condición de unión, debe utilizar ITEM\_NAME.
- Si desea utilizar PRICE en una condición de unión, también debe utilizar ITEM\_NAME en la segunda condición de unión.

Si omite ITEM\_NAME y une ITEM\_NO y PRICE se pierde el orden de clasificación y el servicio de integración de datos no ejecuta la asignación.

Cuando se usa la transformación de incorporación para unir los canales principal y de detalle, se puede configurar cualquiera de las siguientes condiciones de unión:

```
ITEM_NO = ITEM_NO
o
ITEM_NO = ITEM_NO1
ITEM_NAME = ITEM_NAME1
o
ITEM_NO = ITEM_NO1
ITEM_NAME = ITEM_NAME1
PRICE = PRICE1
```

## Cómo unir datos de un mismo origen

Se pueden unir datos de un mismo origen si se desea realizar un cálculo en parte de los datos y unir los datos transformados a los datos originales.

Cuando se unen datos de un mismo origen, pueden mantener los datos originales y transformar partes de los mismos dentro de una asignación. Se pueden unir datos de un mismo origen de las siguientes maneras:

- Unir dos ramificaciones del mismo canal.
- Unir dos instancias del mismo origen.

## Cómo unir dos ramificaciones del mismo canal

Cuando se unen datos de un mismo origen, puede crear dos ramificaciones del canal.

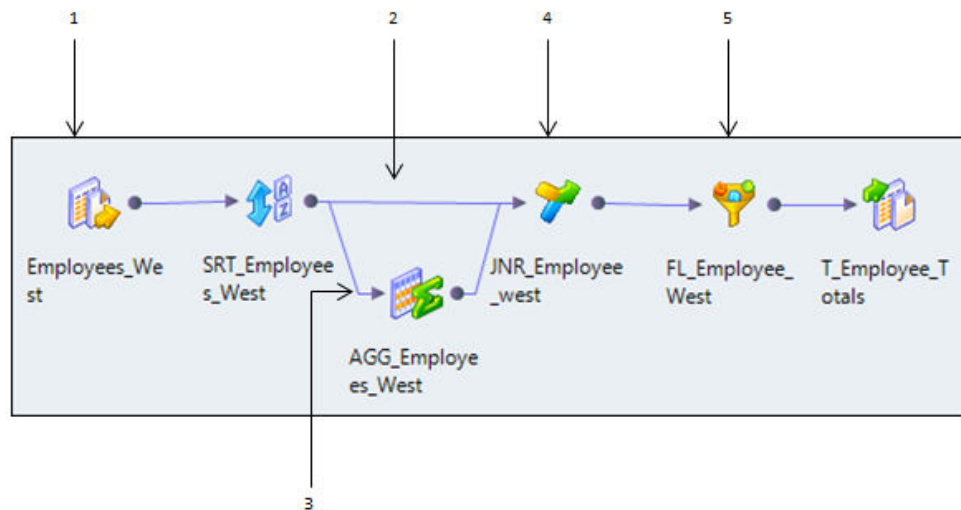
Cuando se ramifica un canal, debe añadirse una transformación entre la entrada de asignación y la transformación de incorporación en, como mínimo, una ramificación del canal. Debe unir datos ordenados y configurar la transformación de incorporación para la entrada ordenada.

Por ejemplo, tiene un origen con los siguientes puertos:

- Empleado
- Departamento
- Total de ventas

En el destino, desea ver los empleados que han generado más ventas que la cifra promedio de ventas de sus departamentos. Para ello, cree una asignación con las siguientes transformaciones:

- Transformación de ordenación. Ordena los datos.
- Transformación de agregación ordenada. Produce un promedio de los datos de ventas y agrupa por departamento. Cuando se realiza esta agregación, se pierden los datos de cada uno de los empleados. Para conservar los datos de los empleados debe pasar una ramificación del canal a la transformación de agregación y pasar una ramificación con los mismos datos a la transformación de incorporación para mantener los datos originales. Cuando se unen ambas ramificaciones del canal se unen los datos agregados con los datos originales.
- Transformación de incorporación ordenada. Une los datos agregados ordenados a los datos originales.
- Transformación de filtro. Compara los datos del promedio de ventas con los datos de ventas de cada empleado y excluye los empleados con una cifra de ventas inferior al promedio de ventas indicado arriba.



1. Origen Employees\_West
2. Ramificación de canal 1
3. Ramificación de canal 2
4. Transformación de incorporación ordenada
5. Filtrar empleados sin un promedio de ventas superior a la media

Unir dos ramificaciones puede reducir el rendimiento si la transformación de incorporación recibe datos de una ramificación mucho después de recibir los datos de la otra ramificación. La transformación de incorporación guarda en memoria caché todos los datos de la primera ramificación y escribe la memoria

caché en disco en el caso de que esté llena. Luego, la transformación de incorporación debe leer los datos del disco cuando recibe los datos de la segunda ramificación.

## Cómo unir dos instancias del mismo origen

Se pueden unir datos de un mismo origen creando una segunda instancia del origen.

Después de crear la segunda instancia del origen, puede unir los canales de las dos instancias del origen. Si desea unir datos no ordenados, debe crear dos instancias del mismo origen y unir los canales.

Cuando se unen dos instancias de un mismo origen, el servicio de integración de datos lee los datos de origen para cada instancia del origen. El rendimiento puede ser más lento que unir dos ramificaciones de un canal.

## Pautas para unir datos de un mismo origen

Se aplican ciertas pautas en el momento de decidir si se deben unir ramificaciones de un canal o unir dos instancias de un origen.

Utilice las siguientes pautas en el momento de decidir si se deben unir ramificaciones de un canal o unir dos instancias de un origen.

- Una dos ramificaciones de un canal cuando el origen sea de gran tamaño o si solamente se pueden leer los datos de origen una sola vez.
- Una dos ramificaciones de un canal cuando utilice datos ordenados. Si los datos de origen no están ordenados y utiliza una transformación de ordenación para ordenar los datos, ramifique el canal después de ordenar los datos.
- Una dos instancias de un origen cuando tenga que añadir una transformación de bloqueo al canal entre el origen y la transformación de incorporación.
- Una dos instancias de un origen en el caso de que un canal se procese más lentamente que el otro canal.
- Una dos instancias de un origen si necesita unir datos no ordenados.

## Cómo bloquear los canales de origen

Cuando se ejecuta una asignación con una transformación de incorporación, el servicio de integración de datos bloquea y desbloquea los datos de origen conforme la configuración de la asignación y si se ha configurado la transformación de incorporación para una entrada ordenada.

## Transformación de incorporación no ordenada

Cuando el servicio de integración de datos procesa una transformación de incorporación no ordenada, lee todas las filas principales antes de leer las filas de detalle. El servicio de integración de datos bloquea el origen de detalle mientras recopila filas del origen principal.

Después de que el servicio de integración de datos lee y guarda en la memoria caché todas las filas principales, desbloquea el origen de detalle y lee las filas de detalle. Algunas asignaciones con transformaciones de incorporación no ordenadas infringen la validación del flujo de datos.

## Transformación de incorporación ordenada

Cuando el servicio de integración de datos procesa una transformación de incorporación ordenada, bloquea datos según la configuración de la asignación. La lógica de bloqueo es posible tanto si la entrada principal como la de detalle de la transformación de incorporación proceden de orígenes diferentes.

El servicio de integración de datos utiliza lógica de bloqueo para procesar la transformación de incorporación, si puede hacerlo, sin bloquear simultáneamente todos los orígenes de un grupo de orden de carga de destino. De lo contrario, no utiliza lógica de bloqueo. En lugar de ello, almacena más filas en la memoria caché.

Cuando el servicio de integración de datos puede utilizar lógica de bloqueo para procesar la transformación de incorporación, almacena menos filas en la memoria caché, con lo que se aumenta el rendimiento.

### Cómo guardar filas maestras en la memoria caché

Cuando el servicio de integración de datos procesa una transformación de incorporación, lee simultáneamente filas de ambos orígenes y construye el índice y la memoria caché de datos según las filas maestras.

Luego, el servicio de integración de datos realiza la unión según los datos de origen de detalle y los datos de la memoria caché. El número de filas que el servicio de integración de datos guarda en la memoria caché depende de los datos de origen y si se ha configurado la transformación de incorporación para una entrada ordenada.

Para aumentar el rendimiento de una transformación de incorporación no ordenada, utilice el origen con menos filas que el origen principal. Para aumentar el rendimiento de una transformación de incorporación ordenada, utilice el origen con menos valores clave duplicados que el origen principal.

## Consejos sobre el rendimiento de la transformación de incorporación

Siga estos consejos para aumentar el rendimiento de la transformación de incorporación.

Las transformaciones de incorporación pueden reducir el rendimiento porque necesitan espacio adicional en tiempo de ejecución para guardar los resultados intermedios. Puede ver la información del contador de rendimiento de la transformación de incorporación para determinar si necesita optimizar las transformaciones de incorporación. Siga los consejos que se dan a continuación para aumentar el rendimiento de la transformación de incorporación:

### **Designa el origen principal como origen con menos valores de clave duplicadas.**

Cuando el servicio de integración de datos procesa una transformación de incorporación ordenada, guarda simultáneamente en la memoria caché cien claves únicas. Si el origen principal contiene muchas filas con el mismo valor de clave, el servicio de integración de datos debe guardar más filas en la memoria caché, con lo cual se reduce el rendimiento.

### **Designa el origen principal como origen con menos filas.**

La transformación de incorporación compara cada fila del origen de detalle con el origen principal. A menor número de filas que haya en el origen principal se producen menos iteraciones de la comparación de uniones, lo que acelera el proceso de unión.

**Realice uniones en una base de datos siempre que sea posible.**

Realizar una unión en una base de datos es más rápido que realizarla durante la ejecución de la asignación. El tipo de unión de base de datos que se utilice puede afectar al rendimiento. Las uniones normales son más rápidas que los outer joins y producen menos filas. En ocasiones no es posible realizar la unión en la base de datos, como unir tablas de dos bases de datos o sistemas de archivos sin formato diferentes.

**Una datos ordenados siempre que sea posible.**

Configure la transformación de incorporación para que utilice la entrada ordenada. El servicio de integración de datos aumenta el rendimiento reduciendo al mínimo la entrada y la salida de disco. Verá un mayor aumento del rendimiento cuando trabaje con conjuntos de datos ordenados. Para una transformación de incorporación no ordenada, designe como origen principal el origen con menos filas.

## Reglas y directrices para una transformación de incorporación

Hay ciertas reglas y directrices que se aplican cuando se usa una transformación de incorporación.

La transformación de incorporación acepta entradas de la mayoría de transformaciones. Sin embargo, no se puede usar una transformación de incorporación cuando uno de los dos canales contiene una transformación de estrategia de actualización.

## CAPÍTULO 19

# Transformación de generador de claves

Este capítulo incluye los siguientes temas:

- [Resumen de la transformación de generador de claves, 226](#)
- [Estrategia Soundex, 227](#)
- [Estrategia de cadena, 228](#)
- [Estrategia NYSIIS, 228](#)
- [Puertos de salida del generador de claves, 229](#)
- [Cómo configurar una estrategia de agrupación, 229](#)
- [Propiedades de la creación de claves, 230](#)
- [Propiedades avanzadas de Transformación de generador de claves, 230](#)

## Resumen de la transformación de generador de claves

La transformación de generador de claves es una transformación activa que organiza registros en grupos según los valores de los datos de una columna seleccionada. Utilice esta transformación para ordenar registros antes de pasarlos a la transformación de coincidencia.

La transformación de generador de claves utiliza una estrategia de agrupación para crear claves de grupo para la columna seleccionada. Las estrategias son String, Soundex y NYSIIS. Los registros con valores comunes en el campo seleccionado tienen un valor de clave de grupo común. La transformación de coincidencia procesa simultáneamente los registros con valores de clave de grupo comunes. Esto permite un análisis de duplicados más rápido en la transformación de coincidencia.

El número de operaciones de comparación que debe realizar la transformación de coincidencia aumenta exponencialmente con el número de registros del conjunto de datos. Este aumento exponencial puede consumir cantidades considerables de recursos informáticos. Mediante la creación de claves de grupo, la transformación de generador de claves permite que la transformación de coincidencia compare registros en grupos más pequeños, con lo cual se reduce el tiempo de procesamiento.

Cuando se realice la coincidencia de campos, seleccione una columna para la generación de claves susceptible de proporcionar grupos útiles para las necesidades de coincidencia. Por ejemplo, una columna Apellido es susceptible de proporcionar datos de clave de grupo más relevantes que una columna Nombre.

Sin embargo, no utilice la columna Apellido si prevé seleccionar esa columna para un análisis de duplicados en la transformación de coincidencia.

La transformación de generador de claves también puede crear un ID único para cada registro. Cada registro que entra en la transformación de coincidencia debe contener un ID único. Utilice la transformación de generador de claves para crear ID para los datos en el caso de que no exista ninguno.

## Estrategia Soundex

La estrategia Soundex analiza palabras y crea claves de grupo mediante códigos alfanuméricos que representan la pronunciación de las palabras.

Los códigos de Soundex comienzan por la primera letra de la palabra, seguida de una serie de números que representan consonantes sucesivas. Utilice la estrategia Soundex para asignar el mismo código a palabras que suenan de forma similar. Configure la profundidad de Soundex para definir el número de caracteres alfanuméricos que devuelve la estrategia.

Esta estrategia se centra en el sonido de las palabras, más que en la ortografía, y puede agrupar pronunciaciones alternativas y variaciones mínimas de la pronunciación. Por ejemplo, los códigos de Soundex para `Smyth` y `Smith` son los mismos.

La estrategia Soundex también puede agrupar palabras mal pronunciadas. Por ejemplo, los códigos de Soundex para `Edmonton` y `Edmonson` son los mismos.

## Propiedades de estrategia Soundex

Configure las propiedades de la estrategia Soundex para determinar los ajustes Soundex que utiliza la transformación de generador de claves para crear una clave de grupo.

En la siguiente tabla se describen las propiedades de la estrategia Soundex:

Propiedad	Descripción
<b>Profundidad de Soundex</b>	Determina el número de caracteres alfanuméricos devueltos por la estrategia Soundex. La profundidad predeterminada es 3. Esta profundidad crea un código Soundex que consta de la primera letra de la cadena y dos números que representan los dos siguientes sonidos consonánticos diferentes.

### TEMAS RELACIONADOS

- [“Propiedades de la estrategia de cadena” en la página 228](#)
- [“Propiedades de la creación de claves” en la página 230](#)
- [“Cómo configurar una estrategia de agrupación” en la página 229](#)

# Estrategia de cadena

La estrategia de cadena crea claves de grupo desde subcadenas en datos de entrada.

Se puede especificar la longitud y la ubicación de una subcadena dentro de la columna de entrada. Por ejemplo, puede configurar esta estrategia para crear una clave a partir de los cuatro primeros caracteres de la cadena de entrada.

## Propiedades de la estrategia de cadena

Configure las propiedades de la estrategia de cadena para determinar las subcadenas que utiliza la transformación de generador de claves para crear una clave de grupo.

En la siguiente tabla se describen las propiedades de la estrategia de cadena:

Propiedad	Descripción
<b>Empezar por la izquierda</b>	Configura la transformación para leer el campo de entrada de izquierda a derecha.
<b>Empezar por la derecha</b>	Configura la transformación para leer el campo de entrada de derecha a izquierda.
<b>Posición de inicio</b>	Especifica el número de caracteres que omitir. Por ejemplo, si especifica 3 para la <b>Posición de inicio</b> , la subcadena comienza por el cuarto carácter del campo de entrada, empezando por el lado especificado.
<b>Longitud</b>	Especifica la longitud de la cadena que utilizar como clave de grupo. Especifique 0 para utilizar el campo de entrada completo.

### TEMAS RELACIONADOS

- [“Propiedades de estrategia Soundex” en la página 227](#)
- [“Propiedades de la creación de claves” en la página 230](#)
- [“Cómo configurar una estrategia de agrupación” en la página 229](#)

# Estrategia NYSIIS

La estrategia NYSIIS analiza palabras y crea claves de grupo mediante letras que representan la pronunciación de las palabras.

Mientras que la estrategia Soundex solamente tiene en cuenta la primera vocal de una cadena, la estrategia NYSIIS analiza todas las vocales incluidas en una cadena. La estrategia NYSIIS convierte todas las letras a uno de seis caracteres y convierte la mayoría de vocales a la letra A.

# Puertos de salida del generador de claves

Los puertos de salida de la transformación de generador de claves crean identificadores y claves de grupo que la transformación de coincidencia utiliza para procesar registros.

En la siguiente tabla se describen los puertos de salida para la transformación del generador de claves:

Propiedad	Descripción
<b>SequenceID</b>	Crea un ID que identifica cada registro en el conjunto de datos de origen.
<b>GroupKey</b>	Crea las claves de grupo que utiliza la transformación de coincidencia para procesar registros.

Cuando cree una transformación reutilizable del generador de claves, utilice la ficha **Resumen** para ver los puertos. Cuando añada una transformación no reutilizable a una asignación, utilice la ficha **Puertos** de la vista **Propiedades** para ver los puertos.

## Cómo configurar una estrategia de agrupación

Para configurar una estrategia de agrupación, edite las propiedades en la vista **Estrategias**.

Antes de configurar una estrategia de generador de claves, añada puertos de entrada a la transformación de generador de claves.

1. Seleccione la vista **Estrategias**.
2. Haga clic en el botón **Nueva**.
3. Seleccione una estrategia de agrupación.
4. Haga clic en **Aceptar**.
5. En la columna **Entradas**, seleccione un puerto de entrada.
6. Configure las propiedades de la estrategia haciendo clic en la flecha de selección del campo de propiedades.
7. Configure las propiedades de la creación de claves.

### TEMAS RELACIONADOS

- [“Propiedades de estrategia Soundex” en la página 227](#)
- [“Propiedades de la estrategia de cadena” en la página 228](#)
- [“Propiedades de la creación de claves” en la página 230](#)

# Propiedades de la creación de claves

Configure las propiedades de creación de claves apropiadas para los datos que se desea analizar.

En la tabla siguiente se describen las propiedades de creación de claves:

Propiedad	Descripción
<b>Ordenar resultados</b>	Ordena la salida de la transformación de generador de claves mediante el campo GroupKey. En el caso de operaciones de coincidencia de campos, debe seleccionar esta opción o comprobar que se proporciona la transformación de coincidencia con datos ordenados. No seleccione esta opción para operaciones de coincidencia de identidad.
<b>Generar clave de secuencia automáticamente</b>	Genera un campo de clave de secuencia según el orden de los datos de entrada.
<b>Usar campo como clave de secuencia</b>	Genera un campo de secuencia para la columna especificada.
<b>Campo de clave de secuencia</b>	Especifica el nombre del campo de la clave de secuencia.

## TEMAS RELACIONADOS

- [“Propiedades de estrategia Soundex” en la página 227](#)
- [“Propiedades de la estrategia de cadena” en la página 228](#)
- [“Cómo configurar una estrategia de agrupación” en la página 229](#)

# Propiedades avanzadas de Transformación de generador de claves

Configure propiedades que ayudan a determinar cómo el servicio de integración de datos procesa los datos para la Transformación de generador de claves.

Puede configurar niveles de seguimiento de registros.

Configure la siguiente propiedad en la ficha **Avanzadas**:

### Nivel de seguimiento

Cantidad de detalles que aparecen en el registro de esta transformación. Puede elegir simplificado, normal, inicialización detallada o datos detallados. El valor predeterminado es Normal.

## CAPÍTULO 20

# Transformación de etiquetador

Este capítulo incluye los siguientes temas:

- [Resumen de la transformación de etiquetador, 231](#)
- [Cuando utilizar una transformación de etiquetador, 232](#)
- [Uso de los datos de referencia en la transformación de etiquetador, 233](#)
- [Estrategias de transformación de etiquetador, 236](#)
- [Puertos de transformación de etiquetador, 237](#)
- [Propiedades de etiquetado de caracteres, 237](#)
- [Propiedades de etiquetado con tokens, 240](#)
- [Cómo configurar una estrategia de etiquetado de caracteres, 243](#)
- [Cómo configurar una estrategia de etiquetado de tokens, 244](#)
- [Propiedades avanzadas de Transformación de etiquetador, 244](#)

## Resumen de la transformación de etiquetador

La transformación de etiquetador es una transformación pasiva que analiza los campos del puerto de entrada campos y escribe etiquetas de texto que describen los datos de cada campo.

Utilice una transformación de etiquetador si desea conocer los tipos de información que un puerto contiene. Utilice una transformación de etiquetador si no conoce los tipos de información en un puerto o si desea para identificar los registros que no contienen los tipos de información esperados en un puerto.

Una etiqueta es una cadena de uno o más caracteres que describe una cadena de entrada. Configure la transformación de etiquetador para asignar etiquetas a cadenas de entrada en función de los datos que contiene cada cadena.

Cuando configure la transformación, especifique los tipos de carácter o de cadena que buscar y especifique la etiqueta que la transformación escribe como salida cuando encuentra el carácter o cadena asociados. Especifique los tipos de carácter y cadena que buscar y las etiquetas que utilizar cuando configure una operación de etiquetado. También puede utilizar los objetos de datos de referencia para especificar los caracteres, las cadenas y las etiquetas.

Configure la transformación para realizar el etiquetado de caracteres o el etiquetado de tokens:

### **Etiquetado de caracteres**

Escribe una etiqueta que describe la estructura de caracteres de la cadena de entrada, incluyendo puntuación y espacios. La transformación escribe una sola etiqueta para cada fila en una columna. Por

ejemplo, la transformación de etiquetador puede etiquetar el código postal 10028 como "nnnnn", donde "n" representa un carácter numérico.

#### Etiquetado de tokens

Escribe una etiqueta que describe el tipo de información en la cadena de entrada. La transformación escribe una etiqueta para cada token identificado en los datos de entrada. Por ejemplo, puede configurar la transformación de etiquetador para que etiquete la cadena "John J. Smith" con los tokens "Word Init Word".

Un token es un valor delimitado en una cadena de entrada.

Cuando el etiquetador encuentra un carácter o una cadena que coincide con una etiqueta que haya especificado, escribe el nombre de la etiqueta en un nuevo puerto de salida.

La transformación de etiquetador utiliza datos de referencia para identificar caracteres y tokens. Seleccione el objeto de datos de referencia al configurar una operación en una estrategia del etiquetador.

## Cuando utilizar una transformación de etiquetador

La transformación de etiquetador escribe una etiqueta descriptiva para cada valor de un puerto.

Los ejemplos siguientes describen algunos de los tipos de análisis que puede realizar con una transformación de etiquetador.

#### Buscar registros con datos de contacto

Configure la transformación con una tabla de referencia que contenga una lista de nombres de pila. Cree una estrategia de etiquetado de tokens para etiquetar cualquier cadena que coincida con un valor de la tabla de referencia. Cuando revise los datos de salida, cualquier registro que contiene la etiqueta es probable que identifique a una persona.

#### Buscar registros de negocio

Configure la transformación con un conjunto de tokens que contenga una lista de sufijos empresariales, tales como Inc, Corp y Ltd. Cree una estrategia de etiquetado de tokens para etiquetar cualquier cadena que coincida con un valor de la tabla de referencia. Cuando revise los datos de salida, cualquier registro que contiene la etiqueta es probable que identifique a una empresa.

**Nota:** Use un conjunto de tokens de sufijos empresariales que desee para identificar cualquier nombre empresarial. Puede usar una tabla de referencia de nombres empresariales si está seguro de que la tabla contiene todos las empresas que desea identificar. Por ejemplo, puede utilizar una tabla de referencia que enumere las corporaciones de la Bolsa de Nueva York.

#### Buscar datos de número de teléfono

Configure la transformación con un conjunto de caracteres que defina la estructura de caracteres de un número de teléfono. Por ejemplo, puede utilizar un conjunto de caracteres que reconozca diferentes patrones de símbolos de puntuación y dígitos como números de teléfono de Estados Unidos. Puede revisar los datos para buscar registros que no contengan los dígitos correctos para un número de teléfono.

Las etiquetas de caracteres pueden utilizar los siguientes caracteres para analizar los datos de columna:

```
c=punctuation character n=digit s=space
```

La siguiente tabla muestra estructuras de números de teléfono de muestra:

Estructura de caracteres	Número de teléfono
cnncsnnncnnncnnnn	(212) 555-1212
nnnnnnnnnn	2125551212
cnncnnncnnnn	+212-555-1212

## Uso de los datos de referencia en la transformación de etiquetador

Informatica Developer se instala con distintos tipos de objetos de datos de referencia que puede utilizar con la transformación de etiquetador. También puede crear objetos de datos de referencia.

Cuando añade un objeto de datos de referencia a la estrategia de transformación de etiquetador, la transformación busca en los datos de entrada de la estrategia los valores del objeto de datos de referencia. La transformación reemplaza cualquier valor que encuentra con un valor válido del objeto de datos de referencia, o con un valor que especifique.

La siguiente tabla describe los tipos de datos de referencia que puede utilizar:

Tipo de datos de referencia	Descripción
Conjuntos de caracteres	Identifica los distintos tipos de caracteres, tales como las letras, números, y símbolos de puntuación. Usar en las operaciones de etiquetado de caracteres.
Modelos probabilísticos	Añade capacidades de coincidencia difusa para las operaciones de etiquetas de tokens. La transformación puede utilizar un modelo probabilístico para inferir el tipo de información de una cadena. Para habilitar las capacidades de coincidencia difusa, puede compilar el modelo probabilístico en Developer Tool. Usar en las operaciones de etiquetado con tokens.
Tablas de referencia	Busca las cadenas que coinciden con las entradas en una tabla de base de datos. Usar en las operaciones de etiquetado de caracteres y de etiquetado con tokens.

Tipo de datos de referencia	Descripción
Expresiones regulares	Identifica las cadenas que coinciden con las condiciones que defina. Puede utilizar una expresión regular para buscar una cadena dentro de una cadena más grande. Usar en las operaciones de etiquetado con tokens.
Conjuntos de tokens	Identifica las cadenas según los tipos de información que contienen. Usar en las operaciones de etiquetado con tokens. Informatica se instala con conjuntos de tokens y distintos tipos de definiciones de tokens, tales como definiciones de palabra, número de teléfono, código postal y de códigos de producto.

## Juegos de caracteres

Un juego de caracteres contiene expresiones que identifican caracteres e intervalos de caracteres específicos. Puede utilizar juegos de caracteres en transformaciones de etiquetador que utilizan el modo de análisis de tokens.

Los rangos de caracteres especifican un rango secuencial de los códigos de carácter. Por ejemplo, el rango de caracteres "[A-C]" coincide con los caracteres en mayúscula "A", "B" y "C". Este rango de caracteres no coincide con los caracteres en minúscula "a", "b", o "c".

Utilice juegos de caracteres para identificar un carácter o un rango de caracteres como parte de las operaciones de análisis o de etiquetado. Por ejemplo, puede etiquetar todos los números de una columna que contiene números de teléfono. Después de etiquetar los números, puede identificar patrones con una transformación de analizador y escribir patrones problemáticos en puertos de salida aparte.

## Modelos probabilísticos

Un modelo probabilístico identifica tokens por los tipos de información que contengan y por sus posiciones en una cadena de entrada.

Un modelo probabilístico contiene las siguientes columnas:

- Una columna de entrada que representa los datos del puerto de entrada. Llene la columna con datos de muestra del puerto de entrada. El modelo utiliza los datos de muestra como datos de referencia en las operaciones de análisis y etiquetado.
- Una o más columnas que identifican los tipos de información de cada cadena de entrada y la posición correcta de los tokens de la cadena. Añada las columnas para el modelo y asigne los tokens de cada cadena a la columna correcta.

Cuando se utiliza un modelo probabilístico en una transformación de etiquetador, esta transformación asigna un valor de etiqueta a cada valor en la fila de entrada. Por ejemplo, la transformación etiqueta la cadena "Franklin Delano Roosevelt" como "FIRSTNAME MIDDLENAME LASTNAME".

La transformación del etiquetador puede inferir una coincidencia entre los valores de los datos del puerto de entrada y los valores de datos del modelo, incluso si los datos de puerto no aparecen en el modelo. Esto significa que un modelo probabilístico no necesita enumerar cada token de un conjunto de datos para etiquetar o analizar correctamente los tokens del conjunto de datos.

La transformación utiliza la lógica probabilística o difusa para identificar tokens que coincidan con los tokens del modelo probabilístico. Puede actualizar las reglas de la lógica difusa al compilar el modelo probabilístico. Debe compilar el modelo probabilístico antes de utilizarlo en una asignación.

## Tablas de referencia

Una tabla de referencia es una tabla de base de datos que contiene al menos dos columnas. Una columna contiene la versión estándar o requerida de un valor de datos y otras columnas contienen versiones alternativas del valor. Cuando añade una tabla de referencia a una transformación, la transformación busca los datos de puertos de entrada para los valores que también aparecen en la tabla. Puede crear tablas con cualquier dato que sea útil para el proyecto de datos en el que esté trabajando.

## Expresiones regulares

En el contexto de las operaciones de etiquetado, una expresión regular es una expresión que se puede usar para identificar una cadena específica en datos de entrada. Puede utilizar expresiones regulares en transformaciones de etiquetador que utilizan el modo de etiquetado de tokens.

Las transformaciones de etiquetador utilizan expresiones regulares para hacer coincidir un patrón de entrada y crear una etiqueta individual. Las expresiones regulares que tienen varias salidas no generan varias etiquetas.

## Conjuntos de tokens

Un conjunto de tokens contiene expresiones que identifican tokens específicos. Puede utilizar conjuntos de tokens en transformaciones de etiquetador que utilizan el modo de etiquetado con tokens.

Emplee conjuntos de tokens para identificar tokens específicos como parte de las operaciones de etiquetado de tokens. Por ejemplo, puede utilizar un conjunto de tokens para etiquetar todas las direcciones de correo electrónico que utilizan un formato "AccountName@DomainName". Después de etiquetar los tokens, puede utilizar la transformación de analizador para escribir direcciones de correo electrónico en los puertos de salida que especifique.

Developer Tool incluye conjuntos de tokens definidos por el sistema que pueden usarse para identificar una amplia gama de patrones. Algunos ejemplos de conjuntos de tokens definidos por el sistema son:

- Palabras
- Números
- Números de teléfono
- Direcciones de correo electrónico
- Códigos postales
- Números de identificación nacional, como números de la seguridad social
- Números de tarjeta de crédito

# Estrategias de transformación de etiquetador

Utilice estrategias de etiquetado para asignar etiquetas a los datos de entrada. Para configurar una estrategia del etiquetador, edite los ajustes en la vista **Estrategias** de una transformación del etiquetador.

Cuando se crea una estrategia de etiquetado, se añade una o más operaciones. Cada operación implementa una tarea de etiquetado específica.

La transformación de etiquetador proporciona un asistente que puede usarse para crear estrategias. Cuando se crea una estrategia de etiquetado, se elige entre el modo de etiquetado de caracteres o el modo de etiquetado de tokens. A continuación se añaden las operaciones específicas a ese modo de etiquetado.

**Importante:** Puede modificar el orden de las operaciones y de las estrategias. El orden de las operaciones dentro de una estrategia puede cambiar la salida de ésta, puesto que cada operación lee los resultados de la operación precedente.

## Operaciones de etiquetado de caracteres

Utilice operaciones de etiquetado de caracteres para crear etiquetas que describan los patrones de carácter en los datos.

Se pueden añadir los siguientes tipos de operación a una estrategia de etiquetado de caracteres:

### Caracteres de etiqueta que usan juegos de caracteres

Caracteres de etiqueta que usan juegos de caracteres predefinidos, como dígitos o caracteres alfabéticos. Se pueden seleccionar juegos de caracteres Unicode y distintos de Unicode.

### Caracteres de etiqueta que usan una tabla de referencia

Caracteres de etiqueta con etiquetas personalizadas de una tabla de referencia.

## Operaciones de etiquetado con tokens

Utilice operaciones de etiquetado con tokens para crear etiquetas que describan las cadenas de los datos.

La transformación de etiquetador puede identificar y etiquetar varios tokens de una cadena de entrada. Por ejemplo, puede configurar la transformación de etiquetador para que utilice los conjuntos de tokens Números de teléfono de Estados Unidos y Direcciones de correo electrónico. Cuando la transformación de etiquetador procesa la cadena de entrada "555-555-1212 someone@somewhere.com", la cadena de salida es "USPHONE EMAIL".

Se pueden añadir los siguientes tipos de operación de etiquetado con tokens a una estrategia de etiquetado:

### Etiqueta con tabla de referencia

Cadenas de etiquetas que coinciden con entradas de tabla de referencia.

### Tokens de etiqueta con conjunto de tokens

Patrones de cadenas de etiqueta que coinciden con datos del conjunto de tokens o con datos de modelos probabilísticos.

# Puertos de transformación de etiquetador

Seleccione los puertos de entrada y salida necesarios para las operaciones de etiquetado que haya configurado en la transformación.

Las transformaciones de etiquetador utilizan el siguiente puerto:

## Puertos de entrada

Lee entradas de cadena desde objetos de un nivel superior.

## Puertos de salida etiquetada

Escribe las etiquetas definidas por las operaciones de transformación.

## Puertos de salida agrupados

Hace pasar las cadenas de entrada que corresponden a cada etiqueta de la salida. Seleccione este puerto si va a añadir una transformación del analizador a un nivel inferior de la transformación de etiquetador en un mapplet o asignación y va a configurar la transformación de analizador para ejecutar en modo de análisis basado en patrones. La transformación de analizador asocia la salida del etiquetado de tokens con los datos en los puertos de salida con tokens.

## Puertos de salida de puntuación

Seleccione escribir los valores de puntuación generados por técnicas de coincidencia probabilística en una operación de etiquetado de tokens.

Al ejecutar una operación de etiquetado de tokens que utiliza un modelo probabilístico, la operación genera una puntuación numérica para cada cadena etiquetada. La puntuación representa el grado de similitud entre la cadena de entrada y los patrones definidos en el modelo probabilístico.

# Propiedades de etiquetado de caracteres

Configure las propiedades para operaciones de etiquetado de caracteres en la vista **Estrategias** de la transformación de etiquetador.

## Propiedades generales

Las propiedades generales se aplican a todas las operaciones de etiquetado de caracteres que haya definido en la estrategia. Utilice las propiedades generales para nombrar la estrategia y especifique los puertos de entrada y salida.

En la tabla siguiente, se describen las propiedades generales:

Propiedad	Descripción
Nombre	Proporciona un nombre para la estrategia.
Entradas	Identifica los puertos de entrada que las operaciones de estrategia pueden leer.

Propiedad	Descripción
Salidas	Identifica los puertos de salida en los que las operaciones de estrategia pueden escribir.
Descripción	Proporciona una descripción de texto de la estrategia. Esta propiedad es opcional.

## Propiedades de la tabla de referencia

Cuando defina una estrategia de etiquetado de caracteres, puede añadir operaciones para etiquetar con conjuntos de caracteres y con tablas de referencia. Utilice las propiedades de la tabla de referencia para especificar cómo la transformación utiliza las tablas de referencia.

La tabla siguiente describe las propiedades de la tabla de referencia:

Propiedad	Descripción
Nombre	Proporciona un nombre para la operación.
Tabla de referencia	Especifica las tablas de referencia que utiliza la transformación para etiquetar caracteres
Etiqueta	Especifica el texto de reemplazo para caracteres de entrada que coinciden con las entradas de la tabla de referencia.
Reemplazar otras etiquetas de la estrategia	Determina si esta operación de etiquetado reemplaza otras operaciones de etiquetado.

## Propiedades de los conjuntos de caracteres

Cuando defina una estrategia de etiquetado de caracteres, puede añadir operaciones para etiquetar con conjuntos de caracteres y con tablas de referencia. Utilice las propiedades del conjunto de caracteres para especificar cómo utiliza la transformación los conjuntos de caracteres.

La siguiente tabla describe las propiedades del conjunto de caracteres:

Propiedad	Descripción
Nombre	Proporciona un nombre para la operación.
Seleccionar conjuntos de caracteres	Especifica los conjuntos de caracteres que utiliza la transformación para etiquetar cadenas. Puede reemplazar el texto de sustitución en las cadenas de entrada que coinciden con el conjunto de caracteres. Haga clic en la flecha de selección situada en la columna <b>Etiqueta</b> para introducir texto de sustitución personalizado.
Filtrar texto	Utiliza los caracteres o comodines que haya especificado para filtrar la lista de conjuntos de caracteres.

Propiedad	Descripción
Añadir conjunto de caracteres	Seleccione esta opción para definir un conjunto de caracteres personalizado.
Editar	Edite el contenido de un conjunto de caracteres personalizado.
Importar	Permite crear copias no reutilizables de conjuntos de caracteres almacenados en los conjuntos de contenido. Los cambios en el conjunto de caracteres original no actualizan las copias almacenadas en la transformación de etiquetador.
Eliminar.	Elimina un conjunto de caracteres personalizado.
Especificar orden de ejecución	Establece el orden en que la operación aplica los conjuntos de tokens a los datos. Utilice las flechas ascendentes y descendentes para cambiar el orden.

## Propiedades de filtro

Puede especificar los valores para omitirlos durante una operación de etiquetado. Utilice las propiedades **Omitir texto** para especificar valores que no se aplicarán a las operaciones de etiquetado.

La siguiente tabla describe las propiedades de filtro:

Propiedad	Descripción
Término de búsqueda	Especifica las cadenas que filtra la transformación antes de realizar el etiquetado. Utilice esta función para especificar las excepciones para la estrategia de etiquetado definida.
Distinguir mayúsculas de minúsculas	Determina si las cadenas filtradas deben coincidir con el formato de mayúsculas y minúsculas del término de búsqueda.
Mayúsculas	Convierte las cadenas filtradas a mayúscula.
Iniciar	Especifica la posición de carácter a partir de la cual comenzar a buscar cadenas filtradas.
Finalización	Especifica la posición de carácter en la que dejar de buscar cadenas filtradas.

# Propiedades de etiquetado con tokens

Configure las propiedades para operaciones de etiquetado con tokens en la vista **Estrategias** de la transformación de etiquetador.

## Propiedades generales

Las propiedades generales se aplican a todas las operaciones de etiquetado con tokens que haya definido en la estrategia. Utilice las propiedades generales para dar un nombre a la estrategia, especificar los puertos de entrada y salida y especificar si la estrategia permite técnicas de coincidencia probabilística.

En la tabla siguiente, se describen las propiedades generales:

Propiedad	Descripción
Nombre	Proporciona un nombre para la estrategia.
Entradas	Identifica los puertos de salida que las operaciones de estrategia pueden leer.
Salidas	Identifica el puerto de salida en los que las operaciones de estrategia pueden escribir.
Descripción	Describe la estrategia. La propiedad es opcional.
Utilizar técnicas de coincidencias probabilísticas	Especifica que la estrategia puede utilizar un modelo probabilístico para identificar los tipos de tokens.
Inversión habilitada	Indica que la estrategia lee datos de entrada de derecha a la izquierda. Esta propiedad está deshabilitada para la coincidencia probabilística.
Delimitadores	Especifica los caracteres que utiliza la transformación para evaluar subcadenas en los datos de entrada. El valor predeterminado es espacio. La propiedad está deshabilitada en el etiquetado probabilístico.
Campo de salida con tokens	Indica que la estrategia escribe varias etiquetas en un puerto de salida. Seleccione este campo para crear datos de entrada para los análisis basados en patrones en la transformación de analizador.
Campo de puntuación de salida	Identifica el campo que contienen los valores de puntuación generados en la coincidencia probabilística. Establezca el campo de salida de la puntuación cuando seleccione la opción para utilizar técnicas de coincidencias probabilísticas.
Delimitador de salida	Especifica un carácter para separar valores de datos del puerto de salida. El valor predeterminado es un signo de dos puntos.

## Propiedades de conjuntos de tokens

Las propiedades del conjunto de tokens se aplican cuando configura una operación de etiquetado para utilizar conjuntos de tokens.

En la tabla siguiente, se describen las propiedades generales:

Propiedad	Descripción
Seleccionar conjuntos de tokens	Identifica los conjuntos de tokens que utiliza la transformación para etiquetar cadenas.
Filtrar texto	Utiliza los caracteres o comodines que especifique para filtrar la lista de conjuntos de tokens o las expresiones regulares.
Añadir conjunto de tokens	Permite definir conjuntos de tokens personalizados.
Añadir expresiones regulares	Permite definir expresiones regulares que coincidan con un patrón de entrada.
Editar	Edite el contenido de un conjunto de tokens personalizados o expresión regular.
Importar	Permite crear copias de conjuntos de tokens o expresiones regulares no reutilizables que están almacenadas en el repositorio. Los cambios realizados en un conjunto de tokens no reutilizables no actualiza la copia que añade a la transformación de etiquetador.
Eliminar	Elimina un conjunto de tokens personalizados o expresión regular.
Especificar orden de ejecución	Establece el orden en que la operación aplica los conjuntos de tokens o las expresiones regulares en los datos. Utilice las flechas ascendentes y descendentes para cambiar el orden.

## Propiedades de etiqueta personalizada

Cuando configura una operación de etiqueta de tokens, puede seleccionar la vista **Etiqueta personalizada** para crear etiquetas para términos de búsqueda específicos.

La tabla siguiente describe las propiedades de etiquetas personalizadas:

Propiedad	Descripción
Término de búsqueda	Identifica la cadena que desea buscar.
Distinguir mayúsculas de minúsculas	Especifica si el formato de mayúsculas y minúsculas de las cadenas de entrada debe coincidir con el del término de búsqueda.
Etiqueta personalizada	Identifica la etiqueta personalizada que aplicar.

## Propiedades de coincidencia probabilística

Cuando selecciona las opciones para utilizar las técnicas de coincidencias probabilísticas, puede añadir un modelo probabilístico a la operación de etiquetado. No puede añadir un modelo probabilístico a una estrategia que utilice un conjunto de tokens o una tabla de referencia.

La siguiente tabla describe las propiedades asociadas con coincidencias probabilísticas:

Propiedad	Descripción
Nombre	Proporciona un nombre para la operación.
Filtrar texto	Utiliza los caracteres o comodines que especifique para filtrar la lista de modelos probabilísticos en el repositorio.
Modelo probabilístico	Identifica el modelo probabilístico que utilizar en la operación.

## Propiedades de la tabla de referencia

Las propiedades de la tabla de referencia se aplican al configurar una operación de etiquetado para usar una tabla de referencia.

La tabla siguiente describe las propiedades de la tabla de referencia:

Propiedad	Descripción
Nombre	Proporciona un nombre para la operación.
Tabla de referencia	Especifica la tabla de referencia que utiliza la operación para etiquetar tokens.
Etiqueta	Especifica el texto que la operación escribe en un nuevo puerto si una cadena de entrada coincide con una tabla de referencia entrada.
Distinguir mayúsculas de minúsculas	Determina si el formato de mayúsculas y minúsculas de las cadenas de entrada debe coincidir con el de las entradas de la tabla de referencia.
Reemplazar coincidencias por valores válidos	Reemplaza cadenas etiquetadas con la entrada de una columna Válida en la tabla de referencia.

Propiedad	Descripción
Modo	Determina el método de etiquetado de tokens. Seleccione Inclusivo para etiquetar cadenas que coinciden con entradas de la tabla de referencia. Seleccione Exclusivo para etiquetar cadenas que no coinciden con entradas de la tabla de referencia.
Establecer prioridad	Determina si las operaciones de etiquetado de la tabla de referencia tienen prioridad sobre las operaciones de etiquetado del conjunto de tokens en una estrategia. Si establece esta propiedad, la transformación lleva a cabo el etiquetado de la tabla de referencia antes del etiquetado del conjunto de tokens y el análisis del conjunto de tokens no puede sobrescribir el análisis de la etiqueta de la tabla de referencia.

## Cómo configurar una estrategia de etiquetado de caracteres

Para configurar una estrategia del etiquetador, edite los ajustes en la vista **Estrategias** de la transformación del etiquetador.

1. Seleccione la vista **Estrategias** y haga clic en **Nuevo** para crear una estrategia.  
Se abre el asistente **Estrategia**.
2. Especifique un nombre para la estrategia.
3. Haga clic en los campos **Entradas** y **Salidas** para definir los puertos de la estrategia.
4. Opcionalmente puede introducir una descripción de la estrategia.
5. Seleccione el modo de etiquetado de los caracteres.
6. Haga clic en **Siguiente**.
7. Seleccione el tipo de operación de etiquetado de caracteres que desee configurar. Puede configurar las siguientes operaciones:
  - Etiquetar caracteres usando tablas de referencia.
  - Etiquetar caracteres usando conjuntos de caracteres.
8. Haga clic en **Siguiente**.
9. Configure las propiedades de la operación y haga clic en **Siguiente**.
10. Opcionalmente puede configurar propiedades **Omitir texto**.
11. Haga clic en **Siguiente** para añadir más operaciones a la estrategia, o haga clic en **Finalizar**.

Puede cambiar el orden en el que la transformación procesa las estrategias y las operaciones. En la vista **Estrategias**, seleccione una estrategia u operación y haga clic en **Mover hacia arriba** o **Mover hacia abajo**.

# Cómo configurar una estrategia de etiquetado de tokens

Para configurar una estrategia del etiquetador, edite los ajustes en la vista **Estrategias** de la transformación del etiquetador.

1. Seleccione la vista **Estrategias** y haga clic en **Nuevo** para crear una estrategia.  
Se abre el asistente **Estrategia**.
2. Especifique un nombre para la estrategia.
3. Haga clic en los campos **Entradas y Salidas** para definir los puertos de la estrategia.
4. Opcionalmente puede introducir una descripción de la estrategia.
5. Seleccione el modo de etiquetado de tokens.  
Compruebe o edite las propiedades para el modo que ha seleccionado.
6. Haga clic en **Siguiente**.
7. Seleccione el tipo de operación de etiquetado de tokens que desee configurar. Puede configurar las siguientes operaciones:
  - Etiquetar tokens con un conjunto de tokens.
  - Etiquetar tokens con una tabla de referencia.
  - Etiquetar tokens usando la coincidencia probabilística.
8. Haga clic en **Siguiente**.
9. Configure las propiedades de la operación y haga clic en **Siguiente**.  
Si configura la estrategia para utilizar la coincidencia probabilística, especifique una etiqueta para utilizar para los tokens identificados como datos de residuos.
10. Opcionalmente puede configurar las propiedades de **Etiqueta personalizada**.
11. Haga clic en **Siguiente** para añadir más operaciones a la estrategia o haga clic en **Finalizar**.  
Puede cambiar el orden en el que la transformación procesa las estrategias y las operaciones. En la vista **Estrategias**, seleccione una estrategia u operación y haga clic en **Mover hacia arriba** o **Mover hacia abajo**.

## Propiedades avanzadas de Transformación de etiquetador

Configure propiedades que ayudan a determinar cómo el servicio de integración de datos procesa los datos para la Transformación de etiquetador.

Puede configurar niveles de seguimiento de registros.

Configure la siguiente propiedad en la ficha **Avanzadas**:

### Nivel de seguimiento

Cantidad de detalles que aparecen en el registro de esta transformación. Puede elegir simplificado, normal, inicialización detallada o datos detallados. El valor predeterminado es Normal.

## CAPÍTULO 21

# Transformación de búsqueda

Este capítulo incluye los siguientes temas:

- [Resumen de la transformación de búsqueda, 245](#)
- [Búsquedas conectadas y no conectadas, 246](#)
- [Cómo desarrollar una transformación de búsqueda, 248](#)
- [Condición de búsqueda, 248](#)
- [Memoria caché de búsqueda, 249](#)
- [Propiedades del tiempo de ejecución para Búsquedas en archivos sin formato, tablas de referencia o datos relacionales, 251](#)
- [Propiedades avanzadas para transformaciones de búsqueda de archivo sin formato, 252](#)
- [Propiedades avanzadas para la tabla de referencia o transformaciones de búsqueda de datos relacionales, 253](#)
- [Propiedades avanzadas de transformaciones de búsqueda de objetos de datos lógicos, 254](#)
- [Cómo crear una transformación de búsqueda reutilizable, 254](#)
- [Cómo crear una transformación de búsqueda no reutilizable, 255](#)
- [Cómo crear una transformación de búsqueda no conectada, 256](#)
- [Ejemplo de búsqueda no conectada, 257](#)

## Resumen de la transformación de búsqueda

La transformación de búsqueda es una transformación pasiva o activa que busca datos en un archivo sin formato, objeto de datos lógicos, tabla de referencia o tabla relacional. La transformación de búsqueda puede devolver una o varias filas desde una búsqueda.

Antes de crear una transformación de búsqueda, cree el origen de búsqueda. Importe un archivo sin formato o una tabla de base de datos relacional como un objeto de datos físicos. También puede crear un objeto de datos lógicos o una tabla de referencia para utilizar como un origen de búsqueda. Cuando se crea una transformación de búsqueda, Developer Tool añade las columnas del objeto de datos o tabla de referencia como puertos de búsqueda en la transformación. Después de crear la transformación, configure uno o más puertos de salida para que devuelvan los resultados de la búsqueda. Configure las condiciones de búsqueda, así como otras propiedades de búsqueda.

Cuando se ejecuta una asignación o se obtiene una vista previa de datos, el servicio de integración de datos consulta el origen de búsqueda. El servicio de integración de datos consulta el origen de búsqueda según los puertos de búsqueda de la transformación, las propiedades de búsqueda y la condición de búsqueda. La transformación de búsqueda devuelve el resultado de la búsqueda al destino o a otra transformación.

Se puede configurar una transformación de búsqueda conectada o no conectada. Una transformación conectada está conectada a otra transformación de la asignación. Una transformación no conectada recibe la entrada desde una expresión :LKP de otra transformación. Si la transformación de búsqueda realiza una búsqueda en un objeto de datos lógicos, debe configurar una transformación de búsqueda conectada. Conecte los puertos de entrada de la transformación de búsqueda a una transformación de nivel superior o a un origen de nivel superior. Conecte los puertos de salida a una transformación de nivel inferior o a un destino de nivel inferior.

Puede usar varias transformaciones de búsqueda en una asignación.

Puede realizar las siguientes tareas con una transformación de búsqueda:

- Obtener un valor relacionado. Recuperar un valor desde el origen de búsqueda conforme a un valor en los datos de entrada. Por ejemplo, los datos de entrada contienen un ID de empleado. Recuperar el nombre del empleado desde la búsqueda sOrigen por ID de empleado.
- Recuperar varias filas desde un origen de búsqueda.
- Realizar un cálculo. Recuperar un valor desde una tabla de búsquedas y usarlo en un cálculo. Por ejemplo, puede buscar un porcentaje de impuesto de venta, calcular un impuesto y devolver el impuesto a un destino.
- Realizar una búsqueda no conectada con una expresión :LKP en una transformación que acepta expresiones. Filtrar los resultados con otra expresión en la transformación.

## Búsquedas conectadas y no conectadas

Puede configurar una transformación de búsqueda conectada o una transformación de búsqueda no conectada. Una transformación de búsqueda conectada es una transformación que tiene puertos de entrada y de salida que se conectan a otras transformaciones en una asignación. Una transformación de búsqueda no conectada aparece en la asignación, pero no está conectada a otras transformaciones.

Una transformación de búsqueda no conectada recibe entradas desde el resultado de una expresión :LKP en una transformación de expresión o de agregación. La expresión :LKP pasa parámetros a la transformación de búsqueda y recibe un resultado desde la búsqueda. La expresión :LKP puede pasar resultados de búsqueda a otra expresión de la transformación a fin de filtrar resultados.

La siguiente tabla enumera las diferencias entre las búsquedas conectadas y las búsquedas desconectadas:

Búsqueda conectadas	Búsqueda no conectadas
Recibe valores de entrada directamente desde la asignación.	Recibe valores de entrada desde el resultado de una expresión :LKP de otra transformación.
La memoria caché incluye las columnas de origen de búsqueda en la condición de búsqueda y las columnas de origen de búsqueda que son puertos de salida.	La memoria caché incluye los puertos de búsqueda y de salida en la condición de búsqueda y el puerto de devolución.
Devuelve varias columnas para una misma fila. Puede devolver varias filas.	Devuelve una columna desde una fila a través de un puerto de devolución.
Si no hay ninguna coincidencia para la condición de búsqueda, el servicio de integración de datos devuelve el valor predeterminado para todos los puertos de salida.	Si no hay ninguna coincidencia para la condición de búsqueda, el servicio de integración de datos devuelve NULL.

Búsqueda conectadas	Búsqueda no conectadas
Si hay una coincidencia para la condición de búsqueda, el servicio de integración de datos devuelve el resultado de la condición de búsqueda para todos los puertos de búsqueda/salida.	Si hay una coincidencia para la condición de búsqueda, el servicio de integración de datos devuelve el resultado de la condición de búsqueda a través del puerto de devolución.
La transformación de búsqueda puede pasar varios valores de salida a otra transformación. Enlace puertos a otra transformación.	La búsqueda devuelve un valor de salida a otra transformación. El puerto de devolución de la transformación de búsqueda pasa el valor al puerto que contiene la expresión :LKP en la otra transformación.
Es compatible con valores predeterminados definidos por el usuario.	No es compatible con valores predeterminados definidos por el usuario.

## Búsquedas conectadas

Una transformación de búsqueda conectada es una transformación de búsqueda que está conectada a un origen o un destino en una asignación.

Cuando se ejecuta una asignación que contiene una transformación de búsqueda conectada, el servicio de integración de datos realiza los siguientes pasos:

1. El servicio de integración de datos pasa valores desde otra transformación a los puertos de entrada de la transformación de búsqueda.
2. Para cada fila de entrada, el servicio de integración de datos consulta el origen de búsqueda o la memoria de caché según los puertos de búsqueda y la condición de búsqueda en la transformación.
3. Si la transformación no está guardada en la memoria caché, o si la transformación tiene una memoria caché estática, el servicio de integración de datos devuelve valores desde la consulta de búsqueda.
4. El servicio de integración de datos devuelve datos desde la consulta y los pasa a la siguiente transformación de la asignación.

## Búsquedas no conectadas

Una transformación de búsqueda no conectada es una transformación de búsqueda que no está conectada a un origen ni a un destino en la asignación. Invoque la búsqueda con una expresión :LKP en una transformación que permita expresiones.

La sintaxis de la expresión de búsqueda es :LKP nombre\_transformación\_búsqueda(argumento, argumento, ...)

El orden en que enumera cada argumento debe coincidir con el orden de las condiciones de búsqueda en la transformación de búsqueda. La transformación de búsqueda devuelve el resultado de la consulta mediante el puerto de devolución de la transformación de búsqueda. La transformación que invoca la búsqueda recibe el valor de resultado de búsqueda en el puerto que contiene la expresión :LKP. Si la consulta no consigue devolver un valor, el puerto recibe valores nulos.

Cuando se realiza una búsqueda no conectada, se puede realizar la misma búsqueda varias veces en una asignación. Puede comprobar los resultados de la búsqueda en otra expresión y filtrar las filas según los resultados.

Cuando se ejecuta una asignación que contiene una transformación de búsqueda no conectada, el servicio de integración de datos realiza los siguientes pasos:

1. El servicio de integración de datos pasa parámetros de búsqueda desde una expresión :LKP a transformación de búsqueda. La expresión se encuentra en otra transformación, como la transformación de expresión o la de agregación.
2. El servicio de integración de datos consulta el origen de búsqueda o la memoria caché según los puertos de búsqueda y la condición establecida en la transformación de búsqueda.
3. El servicio de integración de datos devuelve un valor a través del puerto de devolución de la transformación de búsqueda.
4. El servicio de integración de datos devuelve el valor de retorno a la expresión :LKP de la otra transformación. El puerto que contiene la expresión :LKP recibe el resultado.

## Cómo desarrollar una transformación de búsqueda

Cuando se desarrolla una transformación de búsqueda es necesario tener en cuenta factores como el tipo de origen de búsqueda y la condición de búsqueda.

Tenga en cuenta los siguientes factores cuando desarrolle una transformación de búsqueda:

- Si desea o no crear la transformación a partir de un archivo sin formato, objeto de datos lógicos, tabla de referencia u objeto de datos relacionales. Antes de crear una transformación de búsqueda, cree el origen de búsqueda. Importe un archivo sin formato o una tabla de base de datos relacional como un objeto de datos físicos. También puede crear un objeto de datos lógicos o una tabla de referencia para utilizar como un origen de búsqueda.
- Los puertos de salida para la transformación.
- Las condiciones de búsqueda en la transformación.
- Si desea o no que el servicio de integración de datos guarde en la memoria caché los datos de la búsqueda. El servicio de integración de datos puede guardar datos en la memoria caché de archivos sin formato, tablas de referencia u objetos de datos relacionales.

## Condición de búsqueda

El servicio de integración de datos busca datos en el origen de búsqueda con una condición de búsqueda.

La condición de búsqueda es similar a la cláusula WHERE de una consulta SQL. Cuando se configura una condición de búsqueda en una transformación de búsqueda se compara el valor de una o más columnas en los datos de origen con valores en el origen o la memoria caché de búsqueda.

Por ejemplo, los datos de origen contienen un `employee_number`. La tabla de origen de búsqueda contiene `employee_ID`, `first_name` y `last_name`. Configure la siguiente condición de búsqueda:

```
employee_ID = employee_number
```

Para cada `employee_number`, el servicio de integración de datos devuelve las columnas `employee_ID`, `last_name` y `first_name` del origen de búsqueda.

El servicio de integración de datos puede devolver más de una fila desde el origen de búsqueda. Configure la siguiente condición de búsqueda:

```
employee_ID > employee_number
```

El servicio de integración de datos devuelve filas para todos los números de employee\_ID mayores que el número de empleado de origen.

## Reglas y directrices para las condiciones de una transformación de búsqueda

Hay ciertas reglas y directrices que se aplican cuando se especifica una condición para una transformación de búsqueda.

Aplique las siguientes reglas y directrices cuando especifique una condición para una transformación de búsqueda.

- Los tipos de datos para las columnas de una condición de búsqueda deben coincidir.
- Especifique una condición de búsqueda en todas las transformaciones de búsqueda.
- Utilice un puerto de entrada por cada puerto de búsqueda en la condición de búsqueda. Utilice el mismo puerto de entrada en más de una condición de una transformación.
- Cuando se especifican varias condiciones, el servicio de integración de datos evalúa cada condición como AND y no OR. El servicio de integración de datos devuelve filas que cumplen las condiciones configuradas.
- Si incluye varias condiciones en el siguiente orden para aumentar el rendimiento:
  - Igual a (=)
  - Menor que (<), mayor que (>), menor o igual que (<=), mayor o igual que o (>=)
  - Distinto de (!=)
- Utilice los siguientes operadores cuando cree la condición de búsqueda:
  - =, >, <, >=, <=, !=
- El valor de entrada debe cumplir todas las condiciones de búsqueda para poder devolver un valor.
- El servicio de integración de datos busca coincidencias en valores nulos. Por ejemplo, si una columna de una condición de búsqueda de entrada es NULL, el servicio de integración de datos evalúa el NULL como NULL en la búsqueda.
- Si las columnas de una condición de búsqueda son de tipo decimal, la precisión de cada una de ellas debe pertenecer al mismo rango de precisión. Los rangos de precisión válidos son:
  - 0-18 decimal
  - 19-28 decimal
  - 29 decimal y superior

Por ejemplo, si define la condición `DecimalA = DecimalB`, donde `DecimalA` tiene una precisión de 15 y `DecimalB` de 25, la condición no será válida.

## Memoria caché de búsqueda

Puede configurar la transformación de búsqueda para almacenar en memoria caché el origen de la búsqueda. El servicio de integración de datos compila en la memoria una memoria caché al procesar la

primera fila de datos de una transformación de búsqueda en caché. Cuando se completa la asignación, el servicio de integración de datos libera la memoria caché y elimina los archivos de ésta a menos que configure la transformación de búsqueda para que utilice una memoria caché persistente.

Una memoria caché de búsqueda puede mejorar el rendimiento. El servicio de integración de datos consulta la memoria caché de búsqueda en lugar de consultar el origen de búsqueda para cada fila de entrada. Optimice el rendimiento almacenando en la memoria caché el origen de búsqueda cuando la tabla de origen sea grande.

El servicio de integración de datos asigna memoria para la caché en función de la cantidad que establezca en las propiedades de tiempo de ejecución de la transformación de búsqueda. Si la caché es demasiado grande para la memoria, el servicio de integración de datos crea archivos de caché. De forma predeterminada, el servicio de integración de datos almacena los archivos de la memoria caché en el directorio especificado en el parámetro de sistema CacheDir. El servicio de integración de datos almacena los valores de condición en una memoria caché de índice y los valores de salida en la memoria caché de datos.

Si configura una búsqueda de archivo sin formato para la entrada ordenada, el servicio de integración de datos no puede almacenar en memoria caché la búsqueda si las columnas de la condición no están agrupadas. En caso de que lo estén y de que no estén ordenadas, el servicio de integración de datos procesa la búsqueda como si usted no hubiera configurado la entrada ordenada.

Cuando el servicio de integración de datos procesa una búsqueda que no está almacenada en memoria caché, consulta el origen de la búsqueda en lugar de consultar la memoria caché.

## Memoria caché compartida

De forma predeterminada, el servicio de integración de datos comparte la memoria caché para las transformaciones de búsqueda que tienen estructuras de almacenamiento en memoria caché compatibles en un asignación.

Las transformaciones de búsqueda que utilizan los mismos datos o un subconjunto de datos para crear una memoria caché de disco pueden compartir la caché de disco. Por ejemplo, si configura dos instancias de la misma transformación de búsqueda reutilizable de una asignación y utiliza los mismos puertos de salida para ambas instancias, las transformaciones de búsqueda comparten la memoria caché de búsqueda. Si las propiedades de la transformación o la estructura de la memoria caché no permiten la compartición, el servicio de integración de datos no comparte la memoria caché. Si las claves de búsqueda son diferentes, o bien las transformaciones de búsqueda tienen diferentes condiciones de coincidencias múltiples, el servicio de integración de datos no comparte la memoria caché.

## Memoria caché persistente

Si el origen de la búsqueda no cambia de una asignación a otra, puede configurar la transformación de búsqueda para utilizar una memoria caché de búsqueda persistente. El servicio de integración de datos guarda y reutiliza los archivos de memoria caché entre las ejecuciones de la asignación, además de eliminar el tiempo necesario para leer el origen de la búsqueda.

De forma predeterminada, el servicio de integración de datos utiliza una memoria caché no persistente cuando habilita el guardado en memoria caché en una transformación de búsqueda. El servicio de integración de datos elimina los archivos de la memoria caché al final de la ejecución de una asignación. La próxima vez que ejecute la asignación, el servicio de integración de datos compilará la memoria caché desde la base de datos.

La primera vez que el servicio de integración de datos ejecute una asignación mediante una memoria caché de búsqueda persistente, guarda los archivos de caché en un disco en lugar de eliminarlos. La próxima vez que el servicio de integración de datos ejecute la asignación, compilará la memoria caché a partir de los

archivos de la caché. Si la tabla de búsqueda cambia de forma ocasional, puede anular las propiedades de **Búsqueda** para volver a guardar en la memoria caché la búsqueda desde la base de datos.

Si desea guardar y volver a usar los archivos de la memoria caché, puede configurar la transformación de búsqueda para utilizar una memoria caché persistente. Para configurar una memoria caché persistente, configure la propiedad **Memoria caché de búsqueda persistente** en la vista **Avanzadas** de la transformación de búsqueda.

## Propiedades del tiempo de ejecución para Búsquedas en archivos sin formato, tablas de referencia o datos relacionales

Configure las propiedades del tiempo de ejecución para determinar si el servicio de integración de datos guarda en la memoria caché los datos de la búsqueda.

En la siguiente tabla se describen las propiedades del tiempo de ejecución para las transformaciones de búsqueda que realizan búsquedas en archivos sin formato, tablas de referencia o datos relacionales:

Propiedad	Descripción
Almacenamiento en memoria caché de búsqueda habilitado	<p>Indica si el servicio de integración de datos almacena en memoria caché los valores de búsqueda.</p> <p>Cuando se habilita el guardado de búsquedas en memoria caché, el servicio de integración de datos consulta una vez el origen, almacena los valores en memoria caché y consulta valores en la memoria caché. El almacenamiento de valores de consulta en la memoria caché puede aumentar el rendimiento.</p> <p>Cuando se deshabilita el almacenamiento en memoria caché, cada vez que pasa una fila a la transformación, el servicio de integración de datos emite una instrucción de selección al origen de la búsqueda para los valores de búsqueda.</p> <p>El servicio de integración de datos siempre almacena en memoria caché las búsquedas de archivos sin formato.</p>
Tamaño de la memoria caché de datos de búsqueda	Tamaño máximo que el servicio de integración de datos asigna a la memoria caché de datos en la memoria. El valor predeterminado es Auto.
Tamaño de la memoria caché de índice de búsqueda	Tamaño máximo que el servicio de integración de datos asigna al índice en la memoria. El valor predeterminado es Auto.

Propiedad	Descripción
Generar previamente la memoria caché de búsqueda	<p>Permite que el servicio de integración de datos genere la memoria caché de búsqueda antes de que la transformación de búsqueda reciba los datos. El servicio de integración de datos puede generar simultáneamente varios archivos de memoria caché de búsqueda para aumentar el rendimiento.</p> <p>Configure una de las siguientes opciones:</p> <ul style="list-style-type: none"> <li>- Auto. El servicio de integración de datos determina el valor.</li> <li>- Se permite siempre Permite que el servicio de integración de datos genere la memoria caché de búsqueda antes de que la transformación de búsqueda reciba los datos. El servicio de integración de datos puede generar simultáneamente varios archivos de memoria caché de búsqueda para aumentar el rendimiento.</li> <li>- No se permite nunca. El servicio de integración de datos no puede generar la memoria caché de búsqueda antes de que la transformación de búsqueda reciba la primera fila.</li> </ul>
Nombre del directorio de la memoria caché de búsqueda	<p>Directorio empleado para generar los archivos de memoria caché de búsqueda cuando se configura la transformación de búsqueda para almacenar en memoria caché el origen de búsqueda.</p> <p>El valor predeterminado es el parámetro del sistema CacheDir.</p>

## Propiedades avanzadas para transformaciones de búsqueda de archivo sin formato

Configure las propiedades avanzadas de las transformaciones de búsqueda de archivos sin formato, como la memoria caché de búsquedas persistentes y el nivel de seguimiento.

La siguiente tabla describe las propiedades avanzadas de las transformaciones de búsqueda de archivo sin formato:

Propiedad	Descripción
Memoria caché de búsqueda persistente	Indica si el servicio de integración de datos utilizará una memoria caché de búsqueda persistente, que consta, como mínimo, de dos archivos de memoria caché. Si una transformación de búsqueda está configurada para una memoria caché de búsqueda persistente y no existen archivos de memoria caché de búsqueda persistente, el servicio de integración de datos crea los archivos.
Comparación de cadenas con distinción de mayúsculas y minúsculas	El servicio de integración de datos utiliza comparaciones de cadenas que distinguen mayúsculas de minúsculas al realizar consultas en las columnas de cadenas.
Orden nulo	Determina cómo el servicio de integración de datos ordena los valores nulos. Puede elegir entre ordenar los valores nulos por arriba o por abajo. De forma predeterminada, determina cómo el servicio de integración de datos ordena los valores nulos por arriba. Esto reemplaza la configuración del servicio de integración de datos para tratar los valores nulos en operadores de comparación como altos, bajos o nulos. En el caso de búsquedas relacionales, el orden nulo se basa en el valor predeterminado de la base de datos.
Nivel de seguimiento	Define la cantidad de detalle que se muestra en el registro.
Entrada ordenada	Indica que los datos de entrada se ordenan previamente por grupos.
Formato de fecha y hora	Defina un formato de fecha y hora y el ancho de campo. Los milisegundos, los microsegundos o los nanosegundos tienen un ancho de campo de 29. Si no se selecciona un formato de fecha y hora para un puerto, puede especificar cualquier formato de fecha y hora.  El formato predeterminado es YYYY-MM-DD HH24:MI:SS. El formato de fecha y hora no cambia el tamaño del puerto. Este campo es de solo lectura.
Separador de millar	El valor es Ninguno. Este campo es de solo lectura.
Separador decimal	El valor es una coma. Este campo es de solo lectura.

## Propiedades avanzadas para la tabla de referencia o transformaciones de búsqueda de datos relacionales

Configure la memoria caché de búsqueda persistente y la conexión a una base de datos relacional en las propiedades avanzadas.

La siguiente tabla describe las propiedades avanzadas para la tabla de referencia y las transformaciones de búsqueda de datos relacionales:

Propiedad	Descripción
Memoria caché de búsqueda persistente	Indica si el servicio de integración de datos utilizará una memoria caché de búsqueda persistente, que consta, como mínimo, de dos archivos de memoria caché. Si una transformación de búsqueda está configurada para una memoria caché de búsqueda persistente y no existen archivos de memoria caché de búsqueda persistente, el servicio de integración de datos crea los archivos.
Nivel de seguimiento	Define la cantidad de detalle que se incluye en el registro.
Conexión	Conexión a la base de datos relacional que contiene el origen de búsqueda de datos relacionales. Para búsquedas en la tabla de referencia, este campo es de solo lectura.
Consulta SQL personalizada	Para búsquedas de datos relacionales, reemplaza la consulta SQL predeterminada.
Filtro de origen	Para búsquedas de datos relacionales, reduce el número de filas que consulta el servicio de integración de datos.

## Propiedades avanzadas de transformaciones de búsqueda de objetos de datos lógicos

Configure las propiedades avanzadas de transformaciones de búsqueda de objetos de datos lógicos

Puede configurar niveles de seguimiento de registros.

Configure la siguiente propiedad en la ficha **Avanzadas**:

### Nivel de seguimiento

Cantidad de detalles que aparecen en el registro de esta transformación. Puede elegir simplificado, normal, inicialización detallada o datos detallados. El valor predeterminado es Normal.

## Cómo crear una transformación de búsqueda reutilizable

Cree una transformación de búsqueda para buscar datos en un archivo sin formato, objeto de datos lógicos, tabla de referencia u objeto de datos relacionales.

1. Seleccione un proyecto o una carpeta en la vista **Explorador de objetos**.

2. Haga clic en **Archivo > Nueva > Transformación**.
3. Navegue hasta el asistente Búsqueda.
4. Seleccione **Búsqueda de objetos de datos de archivo sin formato**, **Búsqueda de objetos de datos lógicos**, **Búsqueda de tablas de referencia** o **Búsqueda de objetos de datos relacionales**.
5. Haga clic en **Siguiente**.  
Aparece el cuadro de diálogo **Nueva transformación de búsqueda**.
6. Seleccione un objeto de datos físicos o una tabla de referencia en Developer Tool.
7. Especifique un nombre para la transformación.
8. Para **En varias coincidencias**, determine las filas que devolverá la transformación de búsqueda cuando encuentre varias filas que coincidan con la condición de búsqueda.
9. Haga clic en **Finalizar**.  
La transformación de búsqueda aparece en el editor.
10. En la sección **Puertos** de la vista **Resumen**, añada puertos de salida a la transformación.
11. En la vista **Búsqueda**, añada una o más condiciones de búsqueda.
12. En la vista **Avanzadas**, configure las propiedades avanzadas de la búsqueda.

Al agregar la transformación de búsqueda a una asignación, puede establecer las propiedades de búsqueda del tiempo de ejecución en la vista **Tiempo de ejecución**.

## Cómo crear una transformación de búsqueda no reutilizable

Cree una transformación de búsqueda no reutilizable como parte de una asignación o de un mapplet.

1. En una asignación o en un mapplet, arrastre una transformación de búsqueda desde la paleta Transformación hasta el editor.  
A continuación, aparece el cuadro de diálogo **Nuevo**.
2. Seleccione **Búsqueda de objetos de datos de archivo sin formato**, **Búsqueda de objetos de datos lógicos**, **Búsqueda de tablas de referencia** o **Búsqueda de objetos de datos relacionales**.
3. Haga clic en **Siguiente**.  
Aparece el cuadro de diálogo **Nueva transformación de búsqueda**.
4. Seleccione un objeto de datos físicos o una tabla de referencia en Developer Tool.
5. Especifique un nombre para la transformación.
6. Para **En varias coincidencias**, determine las filas que devolverá la transformación de búsqueda cuando encuentre varias filas que coincidan con la condición de búsqueda.
7. Haga clic en **Finalizar**.  
La transformación de búsqueda aparece en el editor.
8. Seleccione la transformación de búsqueda en el editor.  
La barra de herramientas aparece encima de la transformación.
9. En la ficha **Puertos** de la vista **Propiedades**, añada puertos de salida a la transformación.

10. En la ficha **Tiempo de ejecución** de la vista **Propiedades**, configure las propiedades de tiempo de ejecución.
11. En la ficha **Búsqueda** de la vista **Propiedades**, agregue una o más condiciones de búsqueda.
12. En la ficha **Avanzadas** de la vista **Propiedades**, configure las propiedades avanzadas de la búsqueda.

## Cómo crear una transformación de búsqueda no conectada

Cree una transformación de búsqueda no conectada cuando desee realizar una búsqueda desde una expresión. Puede crear una transformación de búsqueda no conectada reutilizable o no reutilizable.

Antes de crear una transformación de búsqueda no conectada, cree el origen de la búsqueda. Importe un archivo sin formato o una tabla de base de datos relacional como un objeto de datos físicos. También puede crear una tabla de referencia para utilizar como un origen de búsqueda.

1. Seleccione un proyecto o una carpeta en la vista **Explorador de objetos**.
2. Haga clic en **Archivo > Nueva > Transformación**.
3. Navegue hasta el asistente Búsqueda.
4. Seleccione **Búsqueda de objetos de datos de archivos sin formato**, **Búsqueda de tablas de referencia** o **Búsqueda de objetos de datos relacionales**.
5. Haga clic en **Siguiente**.  
Aparece el cuadro de diálogo **Nueva búsqueda**.
6. Seleccione un objeto de datos físicos o una tabla de referencia en Developer Tool.
7. Especifique un nombre para la transformación.
8. Para **En varias coincidencias**, determine qué fila devuelve la transformación de búsqueda cuando encuentra varias filas que coinciden con la condición de búsqueda. No elija Devolver todas para una búsqueda no conectada.
9. Haga clic en **Finalizar**.  
La transformación de búsqueda aparece en el editor.
10. En la sección **Puertos** de la vista **Resumen**, añada puertos a la transformación.  
Cree un puerto de entrada para cada argumento en la expresión :LKP. Cree un puerto de entrada para cada condición de búsqueda que haya creado. Puede usar un puerto de entrada en varias condiciones.
11. En la sección **Puertos** de la vista **Resumen**, configure un puerto como puerto de devolución.
12. En la vista **Búsqueda**, añada una o más condiciones de búsqueda para comparar los valores de entrada de la transformación con valores del origen de la búsqueda o de la memoria caché.  
Cuando la condición es verdadera, la búsqueda devuelve un valor en el puerto de devolución. Si la condición de búsqueda es falsa, la búsqueda devuelve NULL.
13. Cree una expresión :LKP para un puerto en una transformación que permita expresiones, como la transformación de agregación, la de expresión o la de estrategia de actualización.
14. Cuando cree una asignación, añada la transformación de búsqueda no conectada a la asignación en el editor, pero no conecte los puertos a las otras transformaciones de la asignación.

# Ejemplo de búsqueda no conectada

Un comercio minorista de California añade un impuesto de venta estatal a cada precio de los artículos que vende a los clientes del estado. El importe del impuesto se basa en el condado donde reside el cliente. Para recuperar el impuesto de venta, cree una transformación de búsqueda que reciba el nombre del condado y que luego devuelva el importe del impuesto de venta para el condado. Si el condado no carga impuesto de venta, la transformación de búsqueda devuelve NULL. Invoque la búsqueda desde una transformación de expresión.

Realice los siguientes pasos para configurar una búsqueda no conectada del impuesto de venta por condado:

1. Importe un objeto de datos físicos de archivo sin formato que contenga los importes del impuesto de venta por condado.
2. Cree la transformación de búsqueda no conectada.
3. Añada puertos de entrada a la transformación de búsqueda.
4. Defina el puerto de devolución.
5. Cree la condición de búsqueda.
6. Invoque la búsqueda desde una transformación de expresión.

## Paso 1. Importar el origen de la búsqueda del impuesto de venta en el repositorio de modelos.

El archivo de impuestos de venta debe encontrarse en el repositorio de modelos antes de crear la transformación de búsqueda. En este escenario, el archivo de impuestos de venta contiene dos campos: Sales\_County y County\_SalesTax. El condado es una cadena que contiene un nombre de condado. County\_SalesTax es un campo decimal que contiene una tasa de impuesto para el condado. El archivo de impuestos de venta es el origen de la búsqueda.

## Paso 2. Crear la transformación de búsqueda no conectada

Cree una transformación de búsqueda de archivo sin formato reutilizable con el objeto de datos de archivo sin formato de impuestos de venta. En este escenario, el nombre de la transformación es Sales\_Tax\_Lookup. Seleccione **Devolver la primera fila** en varias coincidencias.

## Paso 3. Definir los puertos de la transformación de búsqueda

Defina los puertos de la transformación de búsqueda en la ficha **Puertos** de la vista **Propiedades**.

Tipo de puerto	Nombre	Tipo	Longitud	Escala
Entrada	In_County	String	25	
Salida	SalesTax	Decimal	3	3

## Paso 4. Configurar el puerto de devolución de la transformación de búsqueda

El puerto de devolución es el campo del archivo sin formato que la búsqueda recupera. En la ficha **Columnas**, la columna County\_SalesTax es el puerto de devolución.

Cuando la búsqueda da un resultado verdadero, el servicio de integración de datos encuentra el condado en el origen de archivo sin formato. El servicio de integración de datos devuelve un valor de impuesto de venta en el puerto de devolución. Si el servicio de integración de datos no encuentra el condado, el resultado de la búsqueda será falso y el servicio de integración de datos devolverá NULL en el puerto de devolución.

## Paso 5. Definir la condición de búsqueda

En la vista **Búsqueda**, defina la condición de búsqueda para comparar el valor de entrada con valores del origen de la búsqueda.

Para añadir la condición de búsqueda, haga clic en **Columna de búsqueda**.

La condición de búsqueda tiene la siguiente sintaxis:

```
SALES_COUNTY = IN_COUNTY
```

## Paso 6. Crear una transformación de expresión

Cree una transformación de expresión que reciba los registros de ventas desde un archivo sin formato. La transformación de expresión recibe un número de cliente, un importe de venta y el condado de la venta. Devuelve el número de cliente, el importe de venta y un importe de venta.

La transformación de expresión tiene los siguientes puertos:

Tipo de puerto	Nombre	Tipo	Longitud	Precisión	Valor predeterminado
Entrada	Condado	String	25	10	
Transferencia	Cliente	String	10		
Transferencia	SalesAmt	Decimal	10	2	
Salida	SalesTax	Decimal	10	2	0

El puerto SalesTax contiene una expresión :LKP. La expresión invoca la transformación Sales\_Tax\_Lookup y pasa el nombre del condado como parámetro. La transformación Sales\_Tax\_Lookup devuelve la tasa del impuesto de venta a la expresión. La transformación de expresión multiplica la tasa del impuesto por el importe de la venta.

Especifique la siguiente expresión para el puerto SalesTax:

```
(:LKP.Sales_Tax_Lookup(County) * SalesAmt)
```

El puerto SalesTax contiene el resultado de la expresión. Si la búsqueda es infructuosa, la transformación de búsqueda devuelve NULL y el puerto SalesTax contendrá valores nulos.

Puede añadir una expresión para comprobar si hay valores nulos en el puerto SalesTax. Si SalesTax es NULL puede configurar el puerto SalesTax para que devuelva cero. Añada el siguiente texto a la expresión de búsqueda para comprobar si hay valores nulos y devolver cero:

```
IIF(ISNULL(:LKP.Sales_Tax_Lookup(County) * SalesAmt),0, SalesTax)
```

## CAPÍTULO 22

# Transformación de coincidencia

Este capítulo incluye los siguientes temas:

- [Resumen de la transformación de coincidencia, 259](#)
- [Coincidencia de campos y coincidencia de identidades, 260](#)
- [Flujo del proceso del análisis de coincidencia, 260](#)
- [Cálculos de puntuación de coincidencia, 261](#)
- [Rendimiento de asignación de coincidencias, 264](#)
- [Componentes de transformación de coincidencia, 265](#)
- [Coincidencia de campos, 267](#)
- [Coincidencia de identidades, 272](#)
- [Mapplets de coincidencia, 277](#)
- [Cómo configurar una operación de coincidencia, 278](#)

## Resumen de la transformación de coincidencia

La transformación de coincidencia es una transformación activa que compara los registros de entrada entre ellos y genera una puntuación numérica que representa el grado de similitud entre los registros. Utilice la transformación de coincidencia para buscar registros duplicados en un conjunto de datos o entre dos conjuntos de datos.

Puede utilizar la transformación de coincidencia en los siguientes proyectos de datos:

- Proyectos de Administración de la relación con el cliente. Por ejemplo, un almacén diseña una campaña de correo y debe comprobar la base de datos de clientes para buscar registros de clientes duplicados.
- Fusiones y compras. Por ejemplo, un banco compra a otro banco de la misma zona y los dos bancos tienen clientes común.
- Iniciativas de cumplimiento legal. Por ejemplo, una empresa opera de acuerdo con normas gubernamentales o industriales que insisten en que todos los sistemas de datos deben estar desprovistos de datos duplicados.
- Administración de riesgos financieros. Por ejemplo, un banco puede desear buscar relaciones entre titulares de cuentas.

Cuando configure una transformación de coincidencia, seleccione una columna del conjunto de datos y defina una estrategia para comparar los valores de la columna. La transformación compara cada valor de la columna con otro valor de la misma y devuelve una puntuación de coincidencia para cada comparación. Cuando configure la transformación para analizar dos conjuntos de datos, la transformación compara valores

entre dos columnas que haya seleccionado. Puede definir varias estrategias en la transformación y asignar diferentes columnas para cada estrategia.

Cuando utiliza la transformación de coincidencia con un solo conjunto de datos, la transformación calcula el grado de similitud entre los registros del conjunto de datos. Cuando utiliza la transformación de coincidencia con dos conjuntos de datos, la transformación calcula el grado de similitud entre los dos conjuntos de datos y no compara los registros que hay dentro de los conjuntos de datos.

## Coincidencia de campos y coincidencia de identidades

Puede configurar una transformación de coincidencia para analizar los valores sin formato de una columna de datos, o puede configurar la transformación para que infiera formularios alternativos de los valores en una columna e incluya los formularios alternativos en el análisis de coincidencias.

Cuando desee comparar los valores de datos sin formato, puede realizar coincidencias de campo. Cuando desee incluir alternativas inferidas con los valores de datos sin formato, puede realizar una coincidencia de identidades. Puede definir una operación de coincidencia de identidades si desea buscar nombres duplicados que puedan aparecer en diferentes formularios del conjunto de datos, tales como "John Smith" y "Smith, J". En el análisis de coincidencias, una identidad es un conjunto de valores de datos dentro de un registro que, de forma colectiva, proporciona suficiente información para identificar a una persona o al nombre de una organización.

**Nota:** Las operaciones de coincidencia de identidades leen archivos de datos de referencia llamados poblaciones que definen distintos tipos de identidad. Estos archivos no se instalan junto con las aplicaciones de Informática. Debe descargar los archivos de datos de poblaciones e instalarlos en una ubicación a la que pueda acceder el servicio de administración de contenido. Utilice Informática Administrator para definir la ubicación del servicio de administración de contenido.

## Flujo del proceso del análisis de coincidencia

La complejidad del proceso de análisis de coincidencia depende del tamaño y de la estructura del conjunto de datos y de los objetivos del proyecto. También depende si se han seleccionado campos o identidades individuales.

Puede definir un proceso que solamente utilice la transformación de coincidencia o que utilice varias transformaciones. Puede crear uno o más mapplets o asignaciones para satisfacer las necesidades del proceso.

También puede incrustar mapplets de coincidencia en una transformación de coincidencia. Los mapplets de coincidencia utilizan transformaciones de comparación y de media ponderada en lugar de transformaciones de coincidencia. Es necesario validar un mapplet como regla antes de incrustarlo en la transformación de coincidencia.

Debe incluir todas, o algunas, de las siguientes tareas en el proceso:

- Organice los conjuntos de datos en grupos de registros. De este modo se reduce el tiempo de procesamiento para conjuntos de datos de gran tamaño.  
Utilice una transformación de generador de claves para definir estos grupos. El generador de claves crea un valor de clave de grupo para cada registro.  
Utilice la transformación del generador de claves para crear ID de secuencia en el caso de que ya no existan en los datos.
- Seleccione las columnas que comparar y asigne una o más estrategias de análisis de duplicados a las columnas. Puede realizar este paso en una transformación de coincidencia o en una transformación de comparación. Utilice una transformación de comparación si crea una regla de coincidencia que contenga transformaciones para cada operación de coincidencia. Utilice una transformación de coincidencia para definir operaciones de búsqueda de coincidencias en una transformación individual.  
Seleccione cada columna individualmente en la coincidencia de campos. En la coincidencia de identidades, la estrategia de coincidencia de identidades que seleccione define los pares de columnas que se pueden usar.
- Defina el umbral de coincidencia y calcule una puntuación de coincidencia para los resultados de la estrategia de coincidencia. Puede realizar este paso en una transformación de coincidencia o de media ponderada. Utilice la transformación de media ponderada si va a crear un mapplet de coincidencia.
- Determine el método que va a utilizar la transformación para generar los resultados del análisis de coincidencias. Seleccione el método en la vista **Salida de coincidencia**.

## Cálculos de puntuación de coincidencia

La puntuación de coincidencia es un valor numérico que representa el grado de similitud entre dos valores de columna. Un algoritmo calcula una puntuación de coincidencia como un valor decimal dentro del intervalo de 0 a 1. Un algoritmo asigna una puntuación de 1 cuando dos valores de columna son idénticos.

Cuando seleccione varios pares de columnas para el análisis, la transformación calcula una puntuación media según las puntuaciones de las columnas seleccionadas. De forma predeterminada, la transformación asigna la misma ponderación a las puntuaciones de cada par de columnas. La transformación no infiere en la importancia relativa de los datos de columnas del conjunto de datos.

Puede editar los valores ponderados que utiliza la transformación para calcular la puntuación de coincidencia. Edite los valores ponderados cuando desee asignar una prioridad mayor o menor a las columnas del conjunto de datos.

También puede establecer las puntuaciones que la transformación aplica cuando encuentra un valor nulo en una columna. De forma predeterminada, la transformación trata los valores nulos como errores de datos y asigna una puntuación de coincidencia baja a cualquier par de valores que contengan un valor nulo.

**Nota:** El algoritmo que seleccione determina la puntuación de coincidencia entre dos valores. El algoritmo genera una única puntuación para los dos valores. Las puntuaciones de coincidencia no dependen del tipo de salida de coincidencia o del tipo de método de puntuación que haya seleccionado.

## Puntuaciones ponderadas

Cuando seleccione varias columnas para el análisis de coincidencias, la transformación calcula una puntuación media para cada registro según las puntuaciones de las columnas. La puntuación media incluye cualquier valor ponderado que aplique a los algoritmos de comparación para cada columna.

De forma predeterminada, todos los algoritmos utilizan un valor ponderado de 0,5. Puede aumentar este valor si es más probable que las columnas seleccionadas contengan información duplicada. Puede reducir el valor ponderado si es menos probable que los valores duplicados de las columnas seleccionadas indiquen información duplicada original entre registros. La transformación de coincidencia utiliza la puntuación media como la puntuación de coincidencia única para cada par de registros.

## Puntuaciones de coincidencia nulas

De forma predeterminada, una transformación de coincidencia trata un valor nulo como un error de datos. Un algoritmo de coincidencia aplica una puntuación de coincidencia predefinida cuando uno o ambos de los valores de coincidencia son nulos. Puede editar la puntuación de coincidencia para que un algoritmo de coincidencia de campos se aplique a los valores nulos.

### Puntuaciones de coincidencia nulas y algoritmos de coincidencia de campos

Cuando configura un algoritmo de coincidencia de campos, compruebe los valores de la puntuación de coincidencia que aplica el algoritmo a los datos nulos. Un algoritmo de coincidencia de campos aplica una puntuación predeterminada de 0,5 cuando compara dos valores y uno o ambos son nulos. Una puntuación de 0,5 indica un nivel bajo de similitud entre los valores de datos.

Utilice las siguientes directrices para comprobar las puntuaciones de coincidencia nula:

- Cuando el algoritmo analiza las columnas que contienen una clave principal u otros datos críticos, no edite las puntuaciones predeterminadas. En este caso, un valor nulo representa un error de datos y las puntuaciones predeterminadas son adecuados para los datos.
- Cuando el algoritmo analiza columnas que pueden contener datos de forma opcional, actualice los valores de puntuación de coincidencia nulos con el mismo valor que el umbral de coincidencia. Puede cancelar el efecto de los valores nulos en el análisis de coincidencia cuando configure las puntuaciones de coincidencia nulas con el valor del umbral de coincidencia.

### Puntuaciones de coincidencia nulas y algoritmos de coincidencia de identidades

Un algoritmo de coincidencia de identidades aplica una puntuación de coincidencia de 0 cuando compara dos valores y uno o ambos son nulos. No puede editar la puntuación que un algoritmo de coincidencia de identidades aplica a datos nulos.

## Opciones de puntuación del clúster

Un clúster es un conjunto de registros en el que cada registro coincide al menos con otro registro que tenga una puntuación que cumpla o supere el umbral de coincidencia. Seleccione una opción de salida del clúster cuando desee que los registros que coincidan aparezcan juntos en los datos de salida.

Puede seleccionar opciones de salida del clúster para los análisis de coincidencia de campos y de coincidencia de identidades. La transformación crea clústeres de la misma forma para todos los análisis de coincidencia de campos y de coincidencia de identidades que leen dos orígenes de datos. La transformación proporciona una opción adicional para los análisis de coincidencia de identidades que leen un único origen de datos.

Utilice la ficha **Salida de coincidencia** para seleccionar una opción de clúster.

## Análisis de coincidencia de campos y análisis de coincidencias de identidades de origen dual

La transformación de coincidencia utiliza las siguientes reglas para crear clústeres en los análisis de coincidencia de campos y en los análisis de coincidencia de identidades de origen dual:

- Cuando dos registros tienen una puntuación de coincidencia que cumple o supera el umbral de coincidencia, la transformación de coincidencia añade los registros a un clúster.
- Cuando un registro del conjunto de datos coincide con cualquier registro del clúster, el registro se añade al clúster.
- Si un registro de un clúster coincide con un registro de otro clúster, el proceso fusiona los clústeres.
- El proceso de creación de clústeres realiza un barrido continuo de los resultados de la puntuación de coincidencia hasta que todos los registros se asignan a un clúster.
- Un clúster puede contener un único registro si el registro no coincide con ningún otro registro del conjunto de datos.

**Nota:** Cuando dos registros están enlazados en un clúster, esto no indica que los dos registros representen la mejor coincidencia para cada uno en los datos. Cada uno de los registros se puede parecer más a otro registro del mismo clúster.

## Análisis de coincidencia de identidades de un solo origen

Entre la opciones de salida de un análisis de coincidencia de identidades de un solo origen se incluye la opción **Clústeres - Mejor coincidencia**. Cuando selecciona la opción, la transformación de coincidencia busca la mayor de puntuación de coincidencia entre cada registro y con cualquier otro registro del conjunto de datos.

La transformación utiliza las siguientes reglas para crear los clústeres en el análisis de mejor coincidencia:

- La transformación identifica el registro que tiene la mayor puntuación de coincidencia con el registro actual. Si la puntuación de coincidencia supera el umbral, la transformación añade el par de registros a un clúster.
- Cuando un registro del conjunto de datos tiene la mayor puntuación de coincidencia con un registro de un clúster, el registro se añade al clúster.
- El proceso de creación de clústeres realiza un barrido continuo de los resultados de la puntuación de coincidencia hasta que todos los registros se asignan a un clúster.
- Un clúster puede contener un único registro si el registro no coincide con ningún otro registro del conjunto de datos.

Dado que un registro puede representar la mejor coincidencia con más de un registro del conjunto de datos, la opción **Clústeres - Mejor coincidencia** puede crear clústeres que contengan más de dos registros.

## Puntuaciones del controlador y puntuaciones de enlace

Cuando selecciona una opción de salida del clúster en la transformación de coincidencia, puede añadir datos de puntuaciones de enlace y de puntuaciones del controlador a la salida.

La puntuación de enlace es la puntuación entre dos registros que los identifica como miembros del mismo clúster. Los enlaces entre registros determinan la composición del clúster. Cualquier registro puede enlazarse a cualquier otro registro del mismo clúster.

La puntuación del controlador es la puntuación entre el registro con el valor más alto de ID de secuencia en un clúster y otro registro del mismo clúster. Las puntuaciones del controlador proporcionan un medio para evaluar todos los registros de un clúster frente a un solo registro. Cuando añade puntuaciones del controlador a la salida de coincidencia, se ejecuta la asignación más lentamente, dado que la transformación de coincidencias no puede calcular las puntuaciones del controlador hasta que se hayan completado todos los clústeres.

**Nota:** Las puntuaciones de enlace y del controlador en un clúster pueden depender del orden en el que los registros se han introducido en la transformación. Dado que un registro puede coincidir con un único registro de un clúster, un clúster puede contener pares de registros con puntuaciones del controlador que no cumplan con el umbral de coincidencia.

## Rendimiento de asignación de coincidencias

Puede previsualizar los factores de datos que determinan el rendimiento de una asignación de análisis de coincidencias. Utilice la opción **Análisis de rendimiento de coincidencias** para comprobar que el sistema tiene recursos para ejecutar la asignación. Utilice la opción **Análisis del clúster de coincidencias** para comprobar que la asignación puede medir exactamente los niveles de similitud del conjunto de datos.

Una asignación del análisis de coincidencias puede requerir recursos del sistema y un tiempo de ejecución considerables debido al número de comparaciones que la asignación debe realizar. El número de los cálculos de comparación está relacionado con el número de registros del conjunto de datos. Por ejemplo, una asignación que compara 5.000 de valores de datos realiza más de 12 millones de los cálculos. Una asignación que compara 1.000.000 de valores realiza casi medio trillón de cálculos. Cada cálculo representa una puntuación de coincidencia entre un par de registros del conjunto de datos.

Para reducir el tiempo de procesamiento y los recursos necesarios para una asignación de coincidencia de campos, puede asignar registros de datos a grupos. La asignación realiza los cálculos de coincidencia solo dentro de cada grupo. Cuando selecciona claves de grupo adecuadas para el conjunto de datos, reduce los recursos requeridos sin perder exactitud de coincidencia. No es necesario agrupar los registros antes del análisis de coincidencia de identidades. Los algoritmos de coincidencia de identidades utilizan claves de grupo que no están visibles en la transformación de coincidencia.

Cuando cree grupos, debe comprobar que los grupos tienen un tamaño válido. Si los grupos son demasiado pequeños, el análisis de coincidencias no puede encontrar todos los datos duplicados del conjunto de datos. Si los grupos son demasiado grandes, el análisis de coincidencias puede devolver duplicados falsos y la asignación de coincidencia tarda más en ejecutarse. En un conjunto de datos grande, seleccione claves de grupo que creen un tamaño medio del grupo de 10.000 registros.

**Nota:** Puede ejecutar el análisis de rendimiento de coincidencia y el análisis del clúster de coincidencia en todas las estrategias que lean un solo origen de datos. Asimismo, puede ejecutar el análisis de rendimiento de coincidencia en una estrategia de coincidencia de campos que lea dos orígenes de datos. No ejecute el análisis de rendimiento de coincidencia o el análisis del clúster de coincidencia en una estrategia de coincidencia de identidades que lea dos orígenes de datos.

### Obtener detalles sobre el análisis de rendimiento de coincidencia

Puede obtener detalles sobre los datos del análisis de coincidencias para ver los pares de registros que cumplen o superan el umbral de coincidencia. Haga doble clic en un registro en la vista **Detalles** y utilice el visor de datos para ver los registros que coinciden con el registro seleccionado. El visor de datos muestra los datos para cada par en una única fila. La fila contiene el ID de fila cada registro en el par.

### Obtener detalles sobre el análisis del clúster de coincidencia

Puede obtener detalles sobre los datos del análisis del clúster para ver los registros de cada clúster. Haga doble clic en un clúster en la vista **Detalles** para ver los datos en el visor de datos. El visor de datos muestra un clúster cada vez. Los datos del clúster incluyen las opciones de puntuación seleccionadas, tales como la puntuación del controlador, la puntuación de enlace, el ID del controlador y el ID de enlace.

## Registro de la transformación de coincidencia

Cuando ejecuta una asignación que utiliza una transformación de coincidencia, el registro de las herramientas del desarrollador realiza un seguimiento del número de cálculos de comparación que realiza la asignación. Para ver los datos del registro, seleccione la opción **Mostrar registro** en el visor de datos.

La asignación actualiza el registro cada 100.000 cálculos.

# Componentes de transformación de coincidencia

Los componentes de una transformación de coincidencia incluyen vistas y puertos. Configure las vistas y los puertos para definir una operación de coincidencia.

## Vistas de transformación de coincidencia

Utilice las vistas de transformación de coincidencia para configurar una operación de coincidencia.

Cuando se configura una operación de coincidencia, se pueden configurar las siguientes vistas:

### Tipo de coincidencia

Seleccione el tipo de análisis de duplicados que realizará la transformación. Puede elegir la coincidencia de campos o la coincidencia de identidades para uno o dos conjuntos de datos.

### Estrategias

Seleccione las columnas de datos que se desea hacer coincidir, y configure la estrategia de coincidencia que aplicará a las columnas.

### Salida de coincidencia

Configure el formato para los resultados del análisis de duplicados.

### Parámetros

Asigne parámetros que el servicio de integración de datos pueda aplicar al ejecutar una asignación que contenga la transformación.

### Avanzadas

Configure la cantidad de detalle que la transformación escribirá en el registro de la sesión.

Defina el número de procesos que las operaciones de coincidencia de identidades utilizarán en tiempo de ejecución.

Configure la transformación para incluir o excluir duplicados exactos de los clústeres de salida.

## Puertos de transformación de coincidencia

La transformación de coincidencia contiene puertos de entrada y de salida predefinidos para los datos relacionados con operaciones de búsqueda de coincidencias.

### Puertos de entrada

Los puertos de entrada de una transformación proporcionan los datos que necesita la transformación para las operaciones de búsqueda de coincidencias. Después de crear una transformación de coincidencia, puede configurar los siguientes puertos de entrada:

### SequenceId

Proporciona un ID que identifica de forma única cada registro en el conjunto de datos de origen. Utilice la transformación de generador de claves para crear ID únicos en el caso de que no exista ninguno en el conjunto de datos.

### GroupKey

Proporciona las claves de grupo que utiliza la transformación de coincidencia para procesar registros. La búsqueda de coincidencias de identidades y de campos pueden usar una clave de grupo. Asegúrese de que la clave de grupo y los campos de ID de secuencia que seleccione provienen de la misma transformación.

**Nota:** Para aumentar la velocidad de las búsquedas de coincidencias, configure el puerto de entrada GroupKey y el puerto de salida al que está conectado con el mismo valor de precisión.

### Puertos de salida

Los puertos de salida de transformación proporcionan información sobre el análisis de duplicados que realiza la transformación. Después de crear una transformación de coincidencia, puede configurar los siguientes puertos de salida:

#### ClusterId

ID del clúster al que pertenece el registro. Se usa en las opciones de salida del clúster.

#### Clave de grupo

La clave de grupo del registro.

#### ClusterSize

Número de registros en el clúster al que pertenece un registro. Los registros que no coinciden con otros registros tienen un tamaño de clúster de 1. Se utiliza en la salida del clúster.

#### RowId y RowId1

ID de fila único del registro. La transformación de coincidencia crea este ID. Este ID puede no coincidir con el número de fila en los datos de entrada.

#### DriverId

El ID de fila del registro de controlador en un clúster. Se usa en las opciones de salida del clúster. El registro del controlador es el registro del clúster con el valor de ID de secuencia más alto.

#### DriverScore

La puntuación de coincidencia entre un registro y el registro de controlador en su clúster.

#### LinkId

El ID de fila del registro que coincidió con el registro actual y que se enlazó al clúster. Se usa en las opciones de salida del clúster.

#### LinkScore

La puntuación de coincidencia entre dos registros que da como resultado la creación de un clúster o la adición de un registro a un clúster. Se usa en las opciones de salida del clúster.

**Nota:** El proceso de búsqueda de coincidencias genera un conjunto individual de puntuaciones para cada estrategia definida. Los valores de la puntuación del controlador y de la puntuación de enlace representan un grado de similitud entre pares diferentes de registros y ofrecen diversos tipos de información. La puntuación de enlace define el contenido del clúster. Debe exceder el umbral de coincidencia. La puntuación del controlador puede ser mayor o menor que la puntuación de enlace y puede ser menor que el umbral de coincidencia.

## Selección de puertos de salida y de salida de coincidencia

La configuración de salida de coincidencia que seleccione determinará los puertos de salida en la transformación. Los puertos de ClusterId, ClusterSize, DriverID y DriverScore aparecen cuando selecciona solo un tipo de salida agrupada.

Seleccione el tipo de salida de transformación necesario antes de configurar la transformación. Cuando cambia el tipo de salida de coincidencia, también puede cambiar el número y el tipo de los puertos que la transformación utiliza para escribir los resultados de coincidencia. Si cambia el tipo de salida de coincidencia, compruebe la configuración del puerto de salida en la transformación después de hacerlo. Si utiliza la transformación en una asignación, puede que sea necesario volver a conectar los puertos de salida a objetos de un nivel inferior en la asignación.

## Coincidencia de campos

Utilice la coincidencia de campos para encontrar datos similares o duplicados en dos o más registros.

Las operaciones de coincidencia de campos comparan los valores de dos campos de datos y calculan las similitudes entre ellos. Cuando se configura la transformación de coincidencia para realizar la coincidencia de campos se selecciona uno o más pares de columnas de los datos de entrada.

## Algoritmos de coincidencia de campos

La transformación de coincidencia incluye algoritmos predefinidos que comparan pares de valores de datos.

### Bigram

Utilice el algoritmo Bigram para comparar cadenas de texto largas, como las direcciones postales introducidas en un único campo.

El algoritmo Bigram calcula una puntuación de coincidencia para dos cadenas de datos según la ocurrencia de caracteres consecutivos en ambas cadenas. El algoritmo busca pares de caracteres consecutivos comunes a ambas cadenas y divide el número de pares de caracteres coincidentes por el número total de pares de caracteres.

### Ejemplo de Bigram

Considere las siguientes cadenas:

- larder
- lerder

Estas cadenas producen los siguientes grupos Bigram:

```
l a, a r, r d, d e, e r  
l e, e r, r d, d e, e r
```

Observe que la segunda ocurrencia de la cadena "e r" dentro de la cadena "lerder" no tiene coincidencia, puesto que no hay una segunda ocurrencia correspondiente de "e r" en la cadena "larder".

Para calcular la puntuación de coincidencia Bigram, la transformación divide el número de pares coincidentes (6) por el número total de pares en ambas cadenas (10). En este ejemplo, las cadenas son similares en un 60% y la puntuación de coincidencia es de 0,60.

## Distancia de Hamming

Utilice el algoritmo Distancia de Hamming cuando la posición de los caracteres de datos sea un factor crítico; por ejemplo, en campos numéricos o de códigos como números de teléfono, códigos postales o códigos de producto.

El algoritmo de la distancia de Hamming calcula una puntuación de coincidencia para dos cadenas de datos calculando el número de posiciones en las que los caracteres difieren de una cadena de datos a otra. En el caso de que las cadenas tengan una longitud diferente, cada carácter adicional de la cadena más larga se cuenta como una diferencia de una cadena a otra.

### Ejemplo de distancia de Hamming

Considere las siguientes cadenas:

- Morlow
- Marlowes

Los caracteres resaltados indican las posiciones que el algoritmo de Hamming identifica como diferentes.

Para calcular la puntuación de coincidencia Hamming, la transformación divide el número de caracteres coincidentes (5) por la longitud de la cadena más larga (8). En este ejemplo, las cadenas son similares en un 62,5% y la puntuación de coincidencia es de 0,625.

## Distancia de edición

Utilice el algoritmo Distancia de edición para comparar palabras o cadenas de texto cortas, como un nombre.

El algoritmo Distancia de edición calcula el "coste" mínimo de transformar una cadena en otra por medio de la inserción, eliminación o reemplazo de caracteres.

### Ejemplo de Distancia de edición

Considere las siguientes cadenas:

- Levenston
- Levenshtein

Los caracteres resaltados indican las operaciones necesarias para transformar una cadena en otra.

El algoritmo Distancia de edición divide el número de caracteres no modificados (8) entre la longitud de la cadena más larga (11). En este ejemplo, las cadenas son similares en un 72,7% y la puntuación de coincidencia es de 0,727.

## Distancia de Jaro

Utilice el algoritmo Distancia de Jaro para comparar dos cadenas cuando la similitud entre los caracteres iniciales de las cadenas tenga prioridad.

La puntuación de coincidencia de la distancia de Jaro refleja el grado de similitud entre los primeros cuatro caracteres de ambas cadenas y el número de transposiciones de carácter identificadas. La transformación pondera la importancia de la coincidencia entre los primeros cuatro caracteres utilizando el valor que se introduce en la propiedad Penalización.

### Propiedades de la distancia de Jaro

Cuando se configura un algoritmo Distancia de Jaro, puede configurar las siguientes propiedades:

### Penalización

Determina la penalización para la puntuación de coincidencia si los primeros cuatro caracteres en dos cadenas comparadas no son idénticos. La transformación resta el valor completo de la penalización cuando no hay ninguna coincidencia con el primer carácter. La transformación resta fracciones de la penalización según la posición que ocupan los otros caracteres sin coincidencia. El valor predeterminado de la penalización es 0,20.

### Distinguir mayúsculas de minúsculas

Determina si el algoritmo Distancia de Jaro tiene en cuenta el formato de mayúsculas y minúsculas de los caracteres cuando realiza operaciones de coincidencia.

### Ejemplo de la distancia de Jaro

Considere las siguientes cadenas:

- 391859
- 813995

Si utiliza el valor de Penalización predeterminado de 0,20 para analizar estas cadenas, el algoritmo Distancia de Jaro devuelve una puntuación de coincidencia de 0,513. Esta puntuación de coincidencia indica que las cadenas tienen una similitud del 51,3%.

### Invertir la distancia de Hamming

Utilice el algoritmo Invertir la distancia de Hamming para calcular el porcentaje de posiciones de carácter que difiere entre dos cadenas, leídas de derecha a izquierda.

El algoritmo de la distancia de Hamming calcula una puntuación de coincidencia para dos cadenas de datos calculando el número de posiciones en las que los caracteres difieren de una cadena de datos a otra. En el caso de que las cadenas tengan una longitud diferente, el algoritmo cuenta cada carácter adicional de la cadena más larga se cuenta como una diferencia de una cadena a otra.

### Ejemplo de Invertir la distancia de Hamming

Considere las siguientes cadenas, que utilizan la alineación de derecha a izquierda para imitar el algoritmo Invertir la distancia de Hamming:

- 1-999-9999
- **011-01**-999-9991

Los caracteres resaltados indican las posiciones que el algoritmo Invertir la distancia de Hamming identifica como diferentes.

Para calcular la puntuación de coincidencia de Invertir la distancia de Hamming, la transformación divide el número de caracteres coincidentes (9) entre la longitud de la cadena más larga (15). En este ejemplo, la puntuación de coincidencia es 0,6, lo que indica que las cadenas tienen una similitud del 60%.

## Propiedades de la estrategia para la coincidencia de campos

Abra el asistente de **estrategia** en la ficha **Estrategias** y configure las propiedades para cada estrategia de coincidencia de campos.

Cuando se configura una estrategia para la coincidencia de campos, se pueden configurar las siguientes propiedades:

#### Nombre

Identifica la estrategia por nombre.

### Ponderación

Determina la prioridad de relación asignada a la puntuación de coincidencia cuando se calcula la puntuación general del registro. El valor predeterminado es 0,5.

### Campo único nulo

Define la puntuación de coincidencia que el algoritmo aplica a un par de valores de datos cuando un valor es nulo. El valor predeterminado es 0,5.

### Ambos campos nulos

Define la puntuación de coincidencia que el algoritmo aplica a un par de valores de datos cuando ambos valores son nulos. El valor predeterminado es 0,5.

**Nota:** Un algoritmo de coincidencia no calcula una puntuación de coincidencia cuando uno o ambos valores de columna de coincidencia son nulos. El algoritmo aplica las puntuaciones definidas en las propiedades de coincidencia nula. No puede borrar las propiedades de coincidencia nula.

## Propiedades de salida de coincidencia de campos

Configure las opciones de la **Salida de coincidencia** para definir el formato de salida de las operaciones de coincidencia de campos.

Puede configurar las opciones en las áreas **Tipo de salida de coincidencia** y **Propiedades**.

### Opciones del Tipo de salida de coincidencia

Elija uno de los siguientes tipos de salida de coincidencia:

#### Mejor coincidencia (coincidencia de origen dual)

Realiza la salida de los registros del conjunto de datos principal emparejados con los registros que representan la mejor coincidencia del conjunto de datos secundario.

#### Clústeres

Realiza la salida de los registros coincidentes en los clústeres. Cada registro debe coincidir con al menos otro registro del clúster con una puntuación que cumpla con el umbral de coincidencia.

#### Pares de coincidencia

Realiza la salida de todos los pares de registros con puntuaciones de coincidencia que cumplen con el umbral de coincidencia.

### Opciones de Propiedades

Después de seleccionar un tipo de salida de coincidencia, se pueden configurar las siguientes propiedades:

#### Directorio de la memoria caché

Especifica el directorio que contiene la memoria caché de coincidencia de identidades. El servicio de integración de datos debe poder escribir en este directorio. Este directorio debe formar parte de la estructura de directorios antes de ejecutar la asignación.

#### Tamaño de la memoria caché

Asigna la cantidad de memoria del sistema, en bytes, que la transformación de coincidencia puede usar para procesar operaciones de coincidencia. Siempre que es posible, la transformación de coincidencia utiliza la memoria para realizar el procesamiento. Cuando la memoria caché del sistema está llena, la transformación escribe en el directorio de memoria caché que se especifique.

#### Umbral

Establece la puntuación de coincidencia mínima que usa la transformación de coincidencia para organizar los registros en clústeres. Para que un registro entre en un clúster, debe estar enlazado

con, como mínimo, un registro del clúster con una puntuación de coincidencia que cumpla o exceda el umbral.

#### Método de puntuación

Determina los tipos de puntuación de coincidencia que aparecen en la salida de transformación. Solamente se utiliza para salidas de clústeres de coincidencia. El ajuste predeterminado de esta propiedad es Puntuación de enlace. Si elige Puntuación del controlador o Ambos, es posible que se aumente considerablemente el tiempo de procesamiento.

## Propiedades de coincidencias de campo de origen dual

Para realizar operaciones de coincidencia de campo de origen dual, configure la propiedad **Conjunto de datos principal** en la vista **Tipo de coincidencia**.

Cuando se usan dos orígenes de datos, debe seleccionar uno de ellos como conjunto de datos principal. Si ninguno de los orígenes de datos es un conjunto de datos principal, no importa qué origen de datos se selecciona como conjunto de datos principal.

## Propiedades avanzadas de la coincidencia de campos

Configure las propiedades que determinan cómo el servicio de integración de datos procesa los datos para la transformación de coincidencia.

Configure las siguientes propiedades en la ficha **Avanzadas**:

#### Nivel de seguimiento

Cantidad de detalles que aparecen en el registro de esta transformación. Puede elegir simplificado, normal, inicialización detallada o datos detallados. El valor predeterminado es Normal.

#### Instancias de ejecución

Determina el número de subprocesos que usa la transformación en tiempo de ejecución.

Establezca el número de instancias de ejecución para un valor menor o igual que la cantidad de CPU a las que puede acceder el servicio de integración de datos. El valor que establezca se aplicará a la transformación actual. El valor predeterminado es 1.

**Nota:** Si exporta una transformación de coincidencia a PowerCenter, la Herramienta del desarrollador expande la transformación en varias transformaciones. Cada transformación realiza una tarea diferente, y el número de transformaciones depende de la configuración en la Herramienta del desarrollador. En la exportación, la Herramienta del desarrollador crea una instancia adicional de cada transformación para cada instancia de ejecución. Por ejemplo, si establece el número de instancias de ejecución en 1, el proceso de expansión podría añadir cuatro transformaciones en el repositorio de PowerCenter. Si establece el número de instancias de ejecución en 2 en la misma transformación, el proceso de expansión añade ocho instancias de transformación en el repositorio de PowerCenter.

#### Filtrar coincidencia exacta

Determina si la transformación de coincidencias realiza análisis de coincidencias en registros idénticos. La opción está habilitada cuando selecciona un tipo de salida de clúster. Cuando selecciona la opción **Filtrar coincidencia exacta**, la transformación escribe registros idénticos directamente en la salida del clúster. Seleccione la opción cuando un conjunto de datos contenga muchos registros idénticos.

**Nota:** Los clústeres contienen los mismos datos de registro si selecciona o desactiva la opción. Sin embargo, las puntuaciones de enlace y las puntuaciones del controlador para los registros de un clúster pueden diferir cuando selecciona o desactiva la opción.

# Coincidencia de identidades

Use la coincidencia de identidades para buscar identidades compartidas por varios registros.

Una identidad es un conjunto de valores de datos dentro de un registro que de forma colectiva proporcionan suficiente información para identificar una persona o una organización. Algunas coincidencias de identidades indican la duplicación de registros. Otras indican una conexión entre registros, como una identidad de familia compartida o una identidad de empresario compartida.

Al configurar una transformación de coincidencia para la coincidencia de identidades, debe conectar los puertos de entrada a todos los campos primarios requeridos de una estrategia de identidad. Muchas estrategias de identidad contienen campos primarios requeridos. Las estrategias de identidad Familiar e Individuo también contienen campos secundarios requeridos. Debe conectar puertos de entrada a al menos uno de los campos secundarios requeridos.

## Propiedades de tipo de coincidencia de identidades

Configure las propiedades de tipo de coincidencia cuando seleccione una coincidencia de identidades de un solo origen o origen dual.

Puede configurar las siguientes propiedades en la vista **Tipo de coincidencia**:

### Población

Determina el archivo de datos de referencia que contiene los algoritmos de generación de claves personalizados para las configuraciones regionales e idiomas.

### Nivel de clave

Determina el número de claves generadas por los algoritmos de coincidencia de identidades. La configuración predeterminada es Estándar. El ajuste Limitada produce un número menor de claves y una mayor precisión, y conlleva tiempos de procesamiento más largos. El ajuste Ampliada produce un número mayor de claves y una menor precisión y conlleva tiempos de procesamiento más reducidos.

### Tipo de clave

Describe la categoría de datos que contiene el campo de clave. La búsqueda de coincidencias de identidades puede generar claves para nombres de personas, organizaciones y direcciones. Seleccione el tipo de clave que mejor describa la columna elegida para la propiedad **Campo de clave**.

### Nivel de búsqueda

Indica el equilibrio entre la calidad y la velocidad de búsqueda. La velocidad de búsqueda es inversamente proporcional al número de coincidencias devueltas. Por ejemplo, la opción **Exhaustiva** devuelve menos coincidencias.

### Campo de clave

Especifica la columna que usa la transformación de coincidencia para generar claves de grupo. Compruebe que la columna que haya seleccionado contiene el tipo de datos especificado en la opción **Tipo de clave**.

### Directorio del índice

Especifica la ruta de acceso en el equipo del servicio de integración de datos en la que la transformación de coincidencia crea las carpeta del índice de clave de identidades. El servicio de integración de datos debe poder escribir en esta ruta de acceso.

### Directorio de la memoria caché

El directorio en el equipo del servicio de integración de datos que contiene los datos temporales creados durante el análisis de coincidencia de identidades. El servicio de integración de datos debe poder

escribir en este directorio. Puede usar un parámetro del sistema. El directorio de la memoria caché predeterminado es el parámetro del sistema CacheDir. Si el campo del directorio de la memoria caché está vacío, el servicio de integración de datos escribe la memoria caché en la ubicación de las propiedades de identidad del servicio de administración de contenido.

#### **Conjunto de datos principal**

Especifica el origen de datos que contiene los datos principales. Se utiliza en la búsqueda de coincidencias de orígenes dual.

## **Propiedades del directorio de índice y del directorio de la memoria caché**

El servicio de administración de contenido especifica las ubicaciones predeterminadas para los directorios del índice y de la memoria caché. Puede actualizar las ubicaciones predeterminadas en el servicio de administración de contenido. También puede especificar un directorio de índice o de memoria caché en la transformación de coincidencia.

Cuando especifique un directorio de índice o de memoria caché, introduzca la ruta de acceso local del equipo del servicio de integración de datos. Puede introducir una ruta de acceso totalmente cualificado o una ruta de acceso relativa. Si especifica una ruta de acceso relativa, inicie la ruta de acceso con un punto. La ruta de acceso relativa es relativa al directorio `tomcat/bin` del equipo host del servicio.

**Nota:** No configure el directorio de índice y el directorio de memoria caché en la misma ubicación. La asignación generará un error.

Cuando defina una ubicación del directorio de índice o de memoria caché en la transformación de coincidencia, el servicio de integración de datos utiliza las ubicaciones que ha especificado para los datos solo desde la transformación actual.

### **Propiedades del directorio de índice**

La propiedad del directorio de índice especifica la ruta de acceso al directorio principal para el directorio de índice. Cuando finaliza la asignación, el proceso de coincidencia de identidades crea una carpeta en esta ruta de acceso, escribe los archivos de índice y elimina la carpeta.

### **Propiedades del directorio de la memoria caché**

Para especificar un directorio de la memoria caché en la transformación de coincidencia, cree un parámetro que identifique la ruta de directorio del directorio y asígnelo a la opción de directorio de la memoria caché. Cuando asigne el parámetro, el servicio de integración de datos utilizará la configuración del parámetro para los datos de transformación actuales cuando se ejecute la asignación. El directorio de la memoria caché debe existir en la estructura de directorios antes de que se ejecute la asignación.

De forma predeterminada, el campo del directorio de memoria caché de la transformación de coincidencia contiene el parámetro del sistema CacheDir. Cuando el campo del directorio de memoria caché de la transformación de coincidencia está vacío, la transformación de coincidencia hace referencia al directorio de memoria caché de las propiedades de identidad del servicio de administración de contenido.

## Estrategias de coincidencia de identidades

La transformación de coincidencia incluye estrategias de coincidencia de identidades predefinidas que comparan pares de valores de datos.

La siguiente tabla describe las operaciones de coincidencia y las entradas requeridas para cada estrategia de coincidencia de identidades:

Estrategia de coincidencia de identidades	Operación de coincidencia	Entradas primarias requeridas	Entradas secundarias requeridas (Elija al menos una)
Dirección	Identifica los registros que comparten una dirección.	Dirección	-
Contacto	Identifica los registros que comparten un contacto en una ubicación de una organización.	Person_Name Organization_Name Address_Part1	-
Entidad corporativa	Identifica los registros que comparten datos de una organización. Si lo desea, configure esta estrategia para analizar los datos de direcciones y teléfonos.	Organization_Name	-
División	Identifica los registros que comparte una oficina de una organización con una dirección específica.	Organization_Name Address_Part1	-
Familia	Identifica los individuos que pertenecen a una familia mediante el análisis de los datos de nombre, dirección y número de teléfono.	Person_Name	Address_Part1 Número de teléfono
Campos	Identifica registros que comparten datos para los campos personalizados seleccionados.	-	-
Doméstico	Identifica los individuos que pertenecen a un hogar mediante el análisis de los datos de nombre y dirección.	Person_Name Address_Part1	-
Individuo	Identifica individuos duplicados mediante el análisis de los datos de dirección, fecha de nacimiento e ID. Ejemplos de ID son números de la seguridad social, números de cuenta y números de identificación de vehículos.	Person_Name	Fecha ID
Organización	Identifica los registros que comparten datos de una organización.	Organization_Name	-
Nombre de persona	Identifica individuos duplicados mediante el análisis de datos de nombres.	Person_Name	-

Estrategia de coincidencia de identidades	Operación de coincidencia	Entradas primarias requeridas	Entradas secundarias requeridas (Elija al menos una)
Residente	Identifica individuos duplicados en una dirección. Si lo desea, configure esta estrategia para analizar la información sobre ID.	Person_Name Address_Part1	-
Contacto en general	Identifica los registros que comparten un contacto en una organización.	Person_Name Organization_Name	-
Doméstico en general	Identifica los individuos que pertenecen a un mismo hogar.	Address_Part1	-

## Propiedades de la estrategia para la coincidencia de identidades

Configure las propiedades de cada estrategia para la coincidencia de identidades.

Cuando se configura una estrategia para la coincidencia de identidades, se pueden configurar las siguientes propiedades de estrategia:

### Población

Determina la población que aplicar a una coincidencia de identidad. Las poblaciones pueden contener algoritmos de creación de claves para configuraciones locales e idiomas específicos.

### Nivel de coincidencia

Indica el equilibrio entre la calidad y la velocidad de búsqueda. La velocidad de búsqueda es inversamente proporcional al número de coincidencias devueltas. Las búsquedas que utilizan el ajuste *Aproximada* devuelven menos coincidencias, mientras que las búsquedas que usan el ajuste *Conservadora* devuelven más coincidencias.

## Propiedades de salida de coincidencia de identidades

Configure las opciones de **Salida de coincidencia** para definir el formato de salida de las operaciones de coincidencia de identidades.

Puede configurar las opciones en las áreas **Tipo de salida de coincidencia** y **Propiedades**:

### Opciones del Tipo de salida de coincidencia

Elija uno de los siguientes tipos de salida de coincidencia:

#### Mejor coincidencia (coincidencia de origen dual)

Realiza la salida de cada registro del conjunto de datos principal emparejado con el registro que representa la mejor coincidencia del conjunto de datos secundarios.

#### Clústeres - Mejor coincidencia (coincidencia de origen único)

Realiza la salida de clústeres que representan la mejor coincidencia entre un registro y otro registro del mismo conjunto de datos. La puntuación de coincidencia entre los dos registros debe cumplir o superar el umbral de coincidencia. Los clústeres de mejor coincidencia pueden contener más de dos registros si un registro representa la mejor coincidencia con más de un registro del conjunto de datos.

#### Clústeres- Coincidir con todos

Realiza la salida de los registros coincidentes en los clústeres. Cada registro debe coincidir con al menos otro registro del clúster con una puntuación que cumpla con el umbral de coincidencia.

#### Pares de coincidencia

Realiza la salida de todos los pares de registros con puntuaciones de coincidencia que cumplen con el umbral de coincidencia.

#### Opciones de Propiedades

Después de seleccionar un tipo de salida de coincidencia, puede configurar las siguientes propiedades:

##### Directorio de la memoria caché

Determina el directorio donde la transformación de coincidencia escribe los archivos temporales.

##### Tamaño de la memoria caché

Asigna la cantidad de memoria del sistema, en bytes, que la transformación de coincidencia puede usar para procesar operaciones de coincidencia. Siempre que es posible, la transformación de coincidencia utiliza la memoria para realizar el procesamiento. Cuando la memoria caché del sistema está llena, la transformación escribe en el directorio de memoria caché que se especifique.

##### Umbral

Establece la puntuación de coincidencia mínima que usa la transformación de coincidencia para organizar los registros en clústeres. Para que un registro entre en un clúster, debe estar enlazado con, como mínimo, un registro del clúster con una puntuación de coincidencia que cumpla o exceda el umbral.

## Propiedades avanzadas de la coincidencia de identidades

Configure las propiedades que determinan cómo el servicio de integración de datos procesa los datos para la transformación de coincidencia.

Configure las siguientes propiedades en la ficha **Avanzadas**:

#### Nivel de seguimiento

Cantidad de detalles que aparecen en el registro de esta transformación. Puede elegir simplificado, normal, inicialización detallada o datos detallados. El valor predeterminado es Normal.

#### Instancias de ejecución

Determina el número de subprocesos que usa la transformación en tiempo de ejecución.

Establezca el número de instancias de ejecución para un valor menor o igual que la cantidad de CPU a las que puede acceder el servicio de integración de datos. El valor que establezca se aplicará a la transformación actual. El valor predeterminado es 1.

**Nota:** Si exporta una transformación de coincidencia a PowerCenter, la Herramienta del desarrollador expande la transformación en varias transformaciones. Cada transformación realiza una tarea diferente, y el número de transformaciones depende de la configuración en la Herramienta del desarrollador. En la exportación, la Herramienta del desarrollador crea una instancia adicional de cada transformación para cada instancia de ejecución. Por ejemplo, si establece el número de instancias de ejecución en 1, el proceso de expansión podría añadir cuatro transformaciones en el repositorio de PowerCenter. Si establece el número de instancias de ejecución en 2 en la misma transformación, el proceso de expansión añade ocho instancias de transformación en el repositorio de PowerCenter.

#### Filtrar coincidencia exacta

Determina si la transformación de coincidencias realiza análisis de coincidencias en registros idénticos. La opción está habilitada cuando selecciona un tipo de salida de clúster. Cuando selecciona la opción

**Filtrar coincidencia exacta**, la transformación escribe registros idénticos directamente en la salida del clúster. Seleccione la opción cuando un conjunto de datos contenga muchos registros idénticos.

**Nota:** Los clústeres contienen los mismos datos de registro si selecciona o desactiva la opción. Sin embargo, las puntuaciones de enlace y las puntuaciones del controlador para los registros de un clúster pueden diferir cuando selecciona o desactiva la opción.

## Mapplets de coincidencia

Un mapplet de coincidencia es un tipo de mapplet que se puede crear e incrustar en transformaciones de coincidencia.

Para crear mapplets de coincidencia, guarde la configuración de una transformación de coincidencia como mapplet de coincidencia. Cuando se crea un mapplet de coincidencia se convierten los ajustes de la transformación de coincidencia en transformaciones de comparación y de media ponderada.

Después de crear un mapplet de coincidencia, puede añadir transformaciones a procesos de coincidencia personalizados. Por ejemplo, puede añadir una transformación de expresión para evaluar las puntuaciones de enlace de dos estrategias y elegir la puntuación más alta.

A diferencia de las transformaciones de coincidencia, los mapplets de coincidencia son pasivos, lo que significa que puede usarlos como reglas dentro de Analyst Tool. Utilice mapplets de coincidencia en Analyst Tool para hacer coincidir registros como parte de procesos de creación de perfiles de datos.

La transformación de coincidencia solamente puede leer mapplets creados dentro de una transformación de coincidencia.

## Cómo crear un mapplet de coincidencia

Cree un mapplet de coincidencia para reutilizar operaciones de coincidencia en varias transformaciones de coincidencia y asignaciones.

1. Abra una transformación de coincidencia en el editor y seleccione la vista **Estrategias**.
2. Seleccione **Usar regla de coincidencia**.
3. En el campo **Nombre**, seleccione **Crear nuevo**.  
Se abre la ventana **Nuevo mapplet**.
4. En la ventana **Nuevo mapplet**, escriba un nombre para el mapplet y elija una ubicación donde guardarlo.
5. Si lo desea, seleccione **Reutilizar estrategias desde la transformación de coincidencia** para copiar las entradas, las estrategias de coincidencia y las ponderaciones desde la transformación de coincidencia actual al mapplet de coincidencia.

**Nota:** Informatica recomienda usar esta configuración para crear mapplets de coincidencia rápidamente y replicar las funciones de coincidencia definidas actualmente en las transformaciones de coincidencia.

6. Haga clic en **Finalizar**.  
El mapplet de coincidencia se abre en el editor.
7. Si lo desea, cree operaciones de coincidencia añadiendo y configurando transformaciones de comparación y transformaciones de media ponderada en el mapplet de coincidencia.
8. Haga clic en **Archivo > Guardar** para guardar el mapplet.

9. Cierre el mapplet y seleccione el editor que contiene la transformación de coincidencia. Compruebe que el mapplet que ha creado aparece en el campo **Nombre**.
10. Si lo desea, configure los campos de coincidencia en el mapplet haciendo clic en el botón **Campos coincidentes**.  
Se abre la ventana **Configurar regla de coincidencia**.
11. Haga doble clic en los campos de las columnas **Campos de entrada** y **Entradas disponibles** para asignar puertos de entrada a entradas coincidentes.
12. Haga clic en **Archivo > Guardar** para guardar la transformación.

## Cómo usar un mapplet de coincidencia

Puede seleccionar y configurar un mapplet de coincidencia definido previamente en la transformación de coincidencia.

1. Abra una transformación de coincidencia en el editor y seleccione la vista **Estrategias**.
2. Seleccione **Usar regla de coincidencia**.
3. En el campo **Nombre**, seleccione **Usar existente**.  
Se abre la ventana **Configurar regla de coincidencia**.
4. Haga clic en **Explorar** para localizar un mapplet de coincidencia en el repositorio.  
**Importante:** Solamente se pueden seleccionar mapplets creados por la transformación de coincidencia.  
Se abre la ventana **Seleccionar mapplet coincidente**.
5. Seleccione un mapplet coincidente y haga clic en **Aceptar**.
6. Haga doble clic en los campos de las columnas **Campos de entrada** y **Entradas disponibles** para asignar puertos de entrada a entradas coincidentes.
7. Haga clic en **Aceptar**.  
Se cierra la ventana **Configurar regla de coincidencia**.
8. Haga clic en **Archivo > Guardar** para guardar la transformación de coincidencia.

## Cómo configurar una operación de coincidencia

Para configurar una operación de coincidencia, conecte los datos de origen a la transformación de coincidencia y edite las propiedades en las vistas de la transformación.

1. Cree una transformación de coincidencia y conecte los datos de origen a la transformación.
2. Seleccione la vista **Tipo de coincidencia** y elija un tipo de coincidencia.
3. Configure las propiedades para el tipo de proceso de búsqueda de coincidencias que haya seleccionado.  
Si ha seleccionado un tipo de coincidencia de origen dual, configure la propiedad **Conjunto de datos principal**.
4. Seleccione la vista **Estrategias** y elija **Definir estrategias de coincidencia**.
5. Haga clic en **Nuevo**.  
Se abre el asistente **Nueva estrategia de coincidencia**.
6. Elija una estrategia de coincidencia y haga clic en **Siguiente**.

7. Opcionalmente, edite la configuración de ponderación y de coincidencia nula. Haga clic en **Siguiente**.
8. Haga doble clic en las celdas en la columna Disponibles para seleccionar los puertos de entrada que analizar.

Haga clic en **Siguiente** para configurar otra estrategia o haga clic en **Finalizar** para salir del asistente.

**Nota:** Para editar la configuración de la estrategia, haga clic en la flecha de la celda de esa estrategia en la vista **Estrategias**.

9. Seleccione la vista **Salida de coincidencia**.

Elija un tipo de salida de coincidencia y configure las propiedades.

**Nota:** También puede configurar estrategias de coincidencia seleccionando o creando un mapplet de coincidencia en la vista **Estrategias**. Un mapplet de coincidencia es un tipo de mapplet que se puede incrustar en una transformación de coincidencia.

# Transformación de fusión

Este capítulo incluye los siguientes temas:

- [Resumen de la transformación de fusión, 280](#)
- [Cómo configurar una estrategia de fusión, 280](#)
- [Propiedades avanzadas de Transformación de fusión, 281](#)

## Resumen de la transformación de fusión

La transformación de fusión es una transformación pasiva que lee los valores de datos de varias columnas de entrada y crea una columna de salida individual.

Utilice la transformación de fusión para crear datos en un formato preferido. Por ejemplo, puede combinar los campos Customer\_Firstname y Customer\_Surname para crear un campo llamado Customer\_FullName.

Dentro de una transformación de fusión puede crear varias estrategias de fusión. La transformación de fusión proporciona un asistente que puede usarse para crear estrategias.

## Cómo configurar una estrategia de fusión

Para configurar una estrategia de fusión, edite los ajustes en la vista **Estrategias** de una transformación de fusión.

1. Seleccione la vista **Estrategias**.
2. Haga clic en **Nueva**.  
Se abre el asistente **Nueva estrategia**.
3. Haga clic en el campo **Entradas** para seleccionar los puertos de entrada para la estrategia.
4. Para definir el carácter de fusión que colocar entre los elementos fusionados, haga clic en **Elegir**. Si no se elige un carácter de fusión, la transformación de fusión utiliza de forma predeterminada el carácter de espacio.
5. Si lo desea, seleccione **Incluir cadenas vacías en salida fusionada** para incluir las cadenas de entrada vacías en la salida.
6. Haga clic en **Finalizar**.

# Propiedades avanzadas de Transformación de fusión

Configure propiedades que ayudan a determinar cómo el servicio de integración de datos procesa los datos para la Transformación de fusión.

Puede configurar niveles de seguimiento de registros.

Configure la siguiente propiedad en la ficha **Avanzadas**:

## Nivel de seguimiento

Cantidad de detalles que aparecen en el registro de esta transformación. Puede elegir simplificado, normal, inicialización detallada o datos detallados. El valor predeterminado es Normal.

# Transformación de analizador

Este capítulo incluye los siguientes temas:

- [Resumen de la transformación de analizador, 282](#)
- [Modos de transformación del analizador, 283](#)
- [Cuando utilizar una transformación de analizador, 283](#)
- [Uso de los datos de referencia en la transformación de analizador, 284](#)
- [Operaciones de análisis de tokens, 287](#)
- [Puertos del análisis de tokens, 287](#)
- [Propiedades del análisis de tokens, 288](#)
- [Modo de análisis basado en patrones, 291](#)
- [Configurar una estrategia de análisis de tokens, 292](#)
- [Configurar una estrategia de análisis de patrones, 293](#)
- [Propiedades avanzadas de Transformación de analizador, 293](#)

## Resumen de la transformación de analizador

La transformación de analizador es una transformación pasiva que analiza los valores de los datos de entrada en nuevos puertos. La transformación escribe los valores en nuevos puertos según los tipos de información que contengan los valores y la posición de los valores en la cadena de entrada.

Utilice una transformación de analizador cuando desee cambiar la estructura de un conjunto de datos. La transformación de analizador puede añadir columnas a un conjunto de datos y escribir los valores de datos en nuevas columnas. Utilice una transformación de analizador cuando una columna de datos contenga varios valores en una sola columna y desee escribir los valores de datos en columnas individuales según el tipo de información que contengan.

La transformación de analizador analiza valores de datos en los puertos de salida que haya definido. Si la transformación puede identificar un valor de datos de entrada pero un puerto de salida definido no está disponible, la transformación escribe en un puerto de desbordamiento. Si la transformación no puede identificar un valor de datos de entrada, escribe en un puerto de datos sin analizar.

# Modos de transformación del analizador

Cuando cree una transformación de analizador, seleccione el modo de análisis de tokens o el modo de análisis basado en patrones.

Seleccione uno de los siguientes modos:

- Modo de análisis de tokens. Utilice este modo para analizar los valores de entrada que coinciden con valores de los objetos de datos de referencia como conjuntos de tokens, expresiones regulares, modelos probabilísticos y tablas de referencia. Se pueden usar varias estrategias de análisis de tokens en una transformación.
- Modo de análisis basado en patrones. Utilice este modo para analizar los valores de entrada que coinciden con los valores de los conjuntos de patrones.

## Cuando utilizar una transformación de analizador

Utilice la transformación de analizador cuando los campos de datos de una columna contengan más de un tipo de información y desee mover los valores del campo a nuevas columnas. La transformación de analizador le permite crear una nueva columna para cada tipo de información en un conjunto de datos.

Los ejemplos siguientes describen ciertos tipos de cambios estructurales que puede realizar con una transformación de analizador.

### Crear nuevas columnas para los datos de contacto

Puede crear una estructura de datos que analice los datos de nombres desde una sola columna a varias columnas. Por ejemplo, puede crear columnas para tratamientos, nombres de pila, segundos nombres y apellidos.

Configure la transformación con un modelo probabilístico que represente las estructuras de los nombres de personas del puerto de entrada. Puede utilizar un ejemplo de los datos de los puertos de entrada para definir el modelo.

Cree una estrategia de análisis de tokens que se aplique al modelo probabilístico del puerto de entrada y escriba valores de nombre en nuevas columnas. La transformación escribe los valores de nombre en las nuevas columnas según la posición de cada valor en la cadena de entrada y el tipo de nombre que el valor representa.

**Nota:** también puede utilizar una estrategia de análisis basado en patrones para analizar los datos de contacto. Cuando configure una estrategia de análisis basado en patrones, defina los patrones que representan las estructuras de los nombres del puerto de entrada.

### Crear columnas de dirección

Puede crear una estructura de datos que analice una sola columna de datos de dirección en varias columnas que describen una dirección de entrega.

Configure la transformación con las tablas de referencia que contienen elementos de dirección reconocibles, como códigos postales, nombres de estado y nombres de ciudades. Cree una estrategia de análisis de tokens que escriba cada elemento de dirección en un nuevo puerto.

No puede utilizar una tabla de referencia para analizar datos de dirección de calles desde una cadena de entrada porque el nombre de la calle y los datos del número son demasiado generales para que una tabla de referencia pueda capturarlos. Sin embargo, puede utilizar el puerto de desbordamiento para capturar estos datos. Cuando haya analizado todos los datos de ciudades, estados y códigos postales de una dirección, el resto de datos contienen la información de las calles.

Por ejemplo, utilice una estrategia de análisis de tokens para dividir la siguiente dirección en elementos de dirección:

123 MAIN ST NW STE 12 ANYTOWN NY 12345

La estrategia de análisis puede escribir los elementos de dirección en las columnas siguientes:

Nombre de columna	Datos
Desbordamiento	123 MAIN ST NW STE 12
Ciudad	ANYTOWN
Estado	NY
CÓDIGO POSTAL	12345

#### Crear columnas de datos de producto

Puede crear una estructura de datos que analice una sola columna de datos de productos en varias columnas que describen los detalles del producto e inventario.

Configure la transformación con conjuntos de tokens que contengan elementos de inventario, tales como dimensión, color y peso. Cree una estrategia de análisis de tokens que escriba cada elemento de inventario en un nuevo puerto.

Por ejemplo, utilice una estrategia de análisis de tokens para dividir la siguiente descripción de pintura en elementos de inventario independientes:

500ML Red Matt Exterior

La estrategia de análisis puede escribir los elementos de dirección en las columnas siguientes:

Nombre de columna	Datos
Tamaño	500 ML
Color	Rojo
Estilo	Mate
Exterior	Y

## Uso de los datos de referencia en la transformación de analizador

Informatica Developer se instala con varios objetos de datos de referencia que puede utilizar con la transformación de analizador. También puede crear objetos de datos de referencia en Developer Tool.

Cuando añade un objeto de datos de referencia a una transformación de analizador, la transformación escribe las cadenas que coinciden con un valor del objeto con las nuevas columnas que haya especificado.

La siguiente tabla describe los tipos de datos de referencia que puede utilizar:

Tipo de datos de referencia	Descripción
Conjuntos de patrones	Identifica los valores de datos según la posición relativa de cada valor en la cadena.
Modelos probabilísticos	Añade capacidades de coincidencia difusa para las operaciones de análisis de tokens. La transformación puede utilizar un modelo probabilístico para inferir el tipo de información de una cadena. Para habilitar las capacidades de coincidencia difusa, puede compilar el modelo probabilístico en Developer Tool.
Tablas de referencia	Busca las cadenas que coinciden con las entradas en una tabla de base de datos.
Expresiones regulares	Identifica las cadenas que coinciden con las condiciones que defina. Puede utilizar una expresión regular para buscar una cadena dentro de una cadena más grande.
Conjuntos de tokens	Identifica las cadenas según los tipos de información que contienen.  Informatica se instala con conjuntos de tokens y distintos tipos de definiciones de tokens, tales como definiciones de palabra, número de teléfono, código postal y de códigos de producto.

## Conjuntos de patrones

Un conjunto de patrones contiene expresiones que identifican patrones de datos en la salida de una operación de etiquetado con tokens. Puede emplear conjuntos de patrones para analizar el puerto de salida de datos agrupados y para escribir cadenas coincidentes en uno o más puertos de salida. Utilice conjuntos de patrones en transformaciones de analizador que utilizan el modo de análisis de patrones.

Por ejemplo, puede configurar una transformación de analizador para que utilice conjuntos de patrones que identifiquen nombres e iniciales. Esta transformación utiliza los conjuntos de patrones para analizar la salida de una transformación de etiquetador en el modo de etiquetado de tokens. Puede configurar la transformación de analizador para que escriba nombres e iniciales en la salida, en puertos diferentes.

## Modelos probabilísticos

Un modelo probabilístico identifica tokens por los tipos de información que contengan y por sus posiciones en una cadena de entrada.

Un modelo probabilístico contiene las siguientes columnas:

- Una columna de entrada que representa los datos del puerto de entrada. Llene la columna con datos de muestra del puerto de entrada. El modelo utiliza los datos de muestra como datos de referencia en las operaciones de análisis y etiquetado.
- Una o más columnas de etiquetas que identifican los tipos de información en cada cadena de entrada.  
  
Añada las columnas al modelo y asigne etiquetas a los tokens de cada cadena. Utilice las columnas de etiquetas para indicar la posición correcta de los tokens en la cadena.

Cuando se utiliza un modelo probabilístico en una transformación de analizador, el analizador escribe cada valor de entrada en un puerto de salida según la etiqueta que coincida con el valor. Por ejemplo, el

analizador escribe la cadena "Franklin Delano Roosevelt" en los puertos de salida FIRSTNAME, MIDDLENAME y LASTNAME.

La transformación del analizador puede inferir una coincidencia entre los valores de los datos del puerto de entrada y los valores de datos del modelo, incluso si los datos de puerto no aparecen en el modelo. Esto significa que un modelo probabilístico no necesita enumerar cada token de un conjunto de datos para etiquetar o analizar correctamente los tokens del conjunto de datos.

La transformación utiliza la lógica probabilística o difusa para identificar tokens que coincidan con los tokens del modelo probabilístico. Puede actualizar las reglas de la lógica difusa al compilar el modelo probabilístico.

**Nota:** Si añade un modelo probabilístico a una operación de análisis de tokens y, a continuación, edita la configuración de la etiqueta en el modelo probabilístico, puede invalidar la operación. Cuando actualiza la configuración de la etiqueta en un modelo probabilístico, no recrear ninguna operación de análisis que utilice el modelo.

## Tablas de referencia

Una tabla de referencia es una tabla de base de datos que contiene al menos dos columnas. Una columna contiene la versión estándar o requerida de un valor de datos y otras columnas contienen versiones alternativas del valor. Cuando añade una tabla de referencia a una transformación, la transformación busca los datos de puertos de entrada para los valores que también aparecen en la tabla. Puede crear tablas con cualquier dato que sea útil para el proyecto de datos en el que esté trabajando.

## Expresiones regulares

En el contexto de las operaciones de análisis, una expresión regular es una expresión que se puede usar para identificar una o más cadenas en datos de entrada. La transformación de analizador escribe cadenas identificadas en uno o más puertos de salida. Puede utilizar expresiones regulares en transformaciones de analizador que utilizan el modo de análisis de tokens.

Las transformaciones de analizador utilizan expresiones regulares para hacer coincidir patrones en datos de entrada y analizar todas las cadenas coincidentes con una o más salidas. Por ejemplo, puede utilizar una expresión regular para identificar todas las direcciones de correo electrónico en datos de entrada y analizar cada componente de la dirección de correo electrónico en una salida diferente.

## Conjuntos de tokens

Un conjunto de tokens contiene expresiones que identifican tokens específicos. Puede utilizar conjuntos de tokens en transformaciones de analizador que utilizan el modo de análisis de tokens.

Emplee los conjuntos de tokens para identificar tokens como parte de las operaciones de análisis. Por ejemplo, puede utilizar un conjunto de tokens para analizar todas las direcciones de correo electrónico que utilizan un formato "AccountName@DomainName".

# Operaciones de análisis de tokens

En el modo de análisis de tokens, la transformación de analizador analiza cadenas que coinciden con los datos de los conjuntos de tokens, expresiones regulares, modelos probabilísticos o entradas de la tabla de referencia.

Para realizar el análisis de tokens, añada estrategias en la vista **Estrategias** de la transformación. Se puede añadir una o más operaciones a cada estrategia. La transformación proporciona un asistente que puede usarse para crear estrategias.

Se pueden añadir los siguientes tipos de operación a una estrategia de análisis de tokens:

## Analizar usando conjunto de tokens

Utilice conjuntos de tokens predefinidos o definidos por el usuario para analizar datos de entrada. Las operaciones con conjuntos de tokens pueden utilizar expresiones regulares personalizadas que escriben en una o más salidas.

También puede usar modelos probabilísticos para identificar y analizar tokens.

## Analizar usando tabla de referencia

Utilice las tablas de referencia para analizar datos de entrada.

La transformación lleva a cabo las operaciones en el orden que aparecen en la estrategia.

# Puertos del análisis de tokens

Configure los puertos del análisis de tokens con ajustes apropiados para sus datos.

Una transformación de analizador en el modo de análisis de tokens tiene los siguientes tipos de puerto:

## Entrada

Contiene datos que se pasan a la transformación de analizador. La transformación fusiona todos los puertos de entrada en una cadena de datos combinados mediante el **Carácter de unión de entrada** especificado en la ficha **Estrategias**. Si no se especifica un carácter de unión de entrada, la transformación utiliza de forma predeterminada un carácter de espacio.

## Puertos de salida analizada

Puerto(s) definido(s) por el usuario que contienen cadenas analizadas correctamente. En los casos que varias estrategias de análisis utilicen la misma salida, la transformación fusiona la salida en una cadena de datos combinados mediante el **Carácter de unión de salida** especificado en la ficha **Estrategias**. Si no se especifica un carácter de unión de salida, la transformación utiliza de forma predeterminada un carácter de espacio.

## Desbordamiento

Contiene cadenas analizadas correctamente que no caben en el número de salidas definido en la transformación. Por ejemplo, si la salida solamente tiene dos salidas "WORD", la cadena "John James Smith" da como resultado una salida desbordada de "Smith." La transformación de analizador crea un puerto de desbordamiento para cada estrategia que se añade.

Cuando selecciona la opción de desbordamiento detallado, la transformación crea un puerto de desbordamiento para cada etiqueta del modelo.

### Sin analizar

Contiene cadenas que la transformación no puede analizar correctamente. La transformación de analizador crea un puerto no analizado para cada estrategia que se añade.

### Puertos de salida en la coincidencia probabilística

Cuando configura una estrategia de análisis para utilizar técnicas de coincidencias probabilísticas, la transformación de analizador añade un puerto para almacenar las puntuaciones de coincidencia en cada puerto de salida.

La siguiente tabla describe los tipos de puerto:

Tipo de puerto	Puerto creado en la coincidencia probabilística
Puerto de salida analizado	salida [nombre de etiqueta] salida de puntuación [nombre de etiqueta]
Puerto de datos de desbordamiento	salida [desbordamiento de datos] salida de puntuación [desbordamiento de datos]
Puerto de datos sin analizar	salida [datos sin analizar] salida de puntuación [datos sin analizar]

## Propiedades del análisis de tokens

Configure las propiedades para las operaciones de análisis de tokens la vista **Estrategias** en la transformación de analizador.

### Propiedades generales

Las propiedades generales se aplican a todas las operaciones de análisis de tokens que haya definido en la estrategia. Utilice las propiedades generales para dar un nombre a la estrategia, especificar los puertos de entrada y salida y especificar si la estrategia permite técnicas de coincidencia probabilística.

En la tabla siguiente, se describen las propiedades generales.

Propiedad	Descripción
Nombre	Proporciona un nombre para la estrategia.
Entradas	Identifica los puertos de entrada que las operaciones de estrategia pueden leer.
Salidas	Identifica los puertos de salida en los que las operaciones de estrategia pueden escribir.
Descripción	Describe la estrategia. La propiedad es opcional.
Utilizar técnicas de coincidencias probabilísticas	Especifica que la estrategia puede utilizar un modelo probabilístico para identificar tokens.

Propiedad	Descripción
Carácter de unión de entrada	Especifica el carácter utilizado para unir puertos de datos de entrada. La transformación fusiona todos los puertos de entrada en una cadena de datos combinados y analiza la cadena completa.
Carácter de combinación de salida	Especifica el carácter utilizado para combinar valores de datos de salida en el caso de que varias operaciones utilicen la misma salida.
Inversión habilitada	Configura la estrategia para analizar datos de derecha a izquierda. Esta propiedad está deshabilitada para la coincidencia probabilística.
Permitir desbordamiento inverso	Configura la estrategia para analizar datos de izquierda a derecha. Esta propiedad está deshabilitada para la coincidencia probabilística.
Permitir desbordamiento detallado	Crea un campo de desbordamiento único para cada operación de análisis.
Delimitadores	Especifica el delimitador que separa los datos de entrada en tokens separados. El valor predeterminado es espacio.

## Propiedades del modelo probabilístico

Puede seleccionar un modelo probabilístico en lugar de un conjunto de tokens al configurar una estrategia de análisis de tokens. Seleccione la operación **Analizar usando conjunto de tokens** y seleccione la opción para utilizar técnicas de coincidencias probabilísticas.

La tabla siguiente describe las propiedades de modelos probabilísticos:

Propiedad	Descripción
Nombre	Proporciona un nombre para la operación.
Filtrar texto	Utiliza los caracteres o comodines que especifique para filtrar la lista de conjuntos de tokens, modelos probabilísticos o expresiones regulares.
Modelo probabilístico	Identifica el modelo probabilístico que haya seleccionado.

## Propiedades de la tabla de referencia

Las propiedades de la tabla de referencia se aplican al configurar una operación de etiquetado para usar una tabla de referencia.

La tabla siguiente describe las propiedades de la tabla de referencia:

Propiedad	Descripción
Nombre	Proporciona un nombre para la operación.
Tabla de referencia	Especifica la tabla de referencia que utiliza la operación para analizar los valores de entrada.
Distinguir mayúsculas de minúsculas	Determina si el formato de mayúsculas y minúsculas de las cadenas de entrada debe coincidir con el de las entradas de la tabla de referencia.
Reemplazar coincidencias por valores válidos	Reemplaza los datos analizados con los datos de la columna Válida en la tabla de referencia.
Salidas	Especifica los puertos de salida para los datos analizados.

## Propiedades de conjuntos de tokens

Las propiedades del conjunto de tokens se aplican al configurar una operación de análisis para utilizar conjuntos de tokens.

Seleccione la operación **Analizar usando conjunto de tokens** para analizar la entrada con conjuntos de tokens. Borre la opción para utilizar técnicas de coincidencias probabilísticas.

Las tablas siguientes describen las propiedades del conjunto de tokens:

Propiedad	Descripción
Nombre	Proporciona un nombre para la operación.
Conjuntos de tokens (solo una salida)	Especifica los conjuntos de tokens que utiliza la operación para analizar datos. La operación escribe datos en un solo puerto.
Expresión regular (una o varias salidas)	Especifica la expresión regular que utiliza la operación para analizar datos. La operación escribe datos en varios puertos si encuentra varias cadenas en el campo de entrada.
Salidas	Identifica los puertos de salida en los que escriben las operaciones.

Cuando selecciona un conjunto de tokens o una expresión regular, puede añadir, editar, importar o eliminar un conjunto o expresión. También puede filtrar la lista de nombres de token establecidos. La siguiente tabla describe las propiedades que utiliza para realizar estas tareas:

Propiedad	Descripción
Filtrar texto	Utiliza los caracteres o comodines que especifique para filtrar la lista de conjuntos de tokens, modelos probabilísticos o expresiones regulares.
Añadir conjunto de tokens	Seleccione para definir un conjunto de tokens personalizado. Utilice un conjunto de tokens para escribir en una sola salida.
Añadir expresión regular	Seleccione para definir una expresión regular personalizada. Utilice expresiones regulares para escribir en una o varias salidas.
Editar	Edite el contenido de un conjunto de tokens personalizado.
Importar	Permite crear copias no reutilizables de conjuntos de tokens almacenados en los conjuntos de contenido. Los cambios en los conjuntos de tokens originales no actualizan las copias almacenadas en la transformación de analizador.
Eliminar	Elimina un conjunto de tokens personalizado.

## Modo de análisis basado en patrones

En el modo de análisis basado en patrones, la transformación de analizador analiza patrones formados por varias cadenas.

Puede utilizar los siguientes métodos para definir los patrones en el modo de análisis basado en patrones:

- Analice datos de entrada mediante patrones definidos en tablas de referencia. Puede utilizar una tabla de referencia de patrones desde la salida creada a partir de un perfil de una transformación de etiquetador que utiliza el modo de etiquetado de tokens.
- Analice datos de entrada con los patrones que defina.
- Analizar datos de entrada utilizando patrones que haya importado desde un conjunto de patrones reutilizables en el repositorio de modelos. Los cambios en el conjunto de patrones reutilizables no actualizan los datos que añada a la transformación de analizador.

Puede utilizar los caracteres comodín "+" y "\*" para definir un patrón. Utilice caracteres "\*" para hacer coincidir cualquier cadena y caracteres "+" para hacer coincidir una o más instancias de la cadena precedente. Por ejemplo, utilice "WORD+" para buscar varias instancias consecutivas de un token de palabra y utilice "WORD \*" para buscar un token de palabra seguido de uno o más tokens de cualquier tipo.

Puede utilizar varias instancias de estos métodos dentro de la transformación de analizador. La transformación utiliza las instancias en el orden que aparecen en la lista de la vista **Configuración**.

**Nota:** En el modo de análisis basado en patrones, la transformación de analizador requiere la salida de una transformación de etiquetador que utiliza el modo de etiquetado de tokens. Cree y configure la

transformación de etiquetador antes de crear una transformación de analizador que utiliza el modo de análisis basado en patrones.

## Puertos de análisis basado en patrones

Configure los puertos de análisis basado en patrones con los ajustes apropiados a los datos.

Una transformación de analizador que utiliza el modo de análisis basado en patrones tiene los siguientes tipos de puerto:

### Label\_Data

Conecte este puerto al puerto `Labeled_Output` de una transformación de etiquetador que utilice el modo de etiquetado de tokens.

### Tokenized\_Data

Conecte este puerto al puerto `Tokenized_Data` de una transformación de etiquetador que utilice el modo de etiquetado de tokens.

### Parse\_Status

Si se encuentra una coincidencia para el patrón de entrada, el puerto emite el valor `Con coincidencia`.  
Si no se encuentra ninguna coincidencia, emite `Sin coincidencia`.

### Desbordamiento

Cadenas analizadas correctamente que no encajan en el número de salidas definido en la transformación. Por ejemplo, si solamente se han definido dos salidas "WORD", la cadena "John James Smith" da como resultado un desbordamiento de "Smith" de forma predeterminada.

### Analizada

Cadenas analizadas correctamente en puertos definidos por el usuario.

## Configurar una estrategia de análisis de tokens

Para configurar una estrategia de análisis de tokens, abra una transformación de analizador en el modo de análisis de tokens y seleccione la vista **Estrategias**.

1. Seleccione la vista **Estrategias**.
2. Haga clic en **Nueva**.  
Se abre el asistente **Nueva estrategia**.
3. Haga clic en el campo **Entradas** para seleccionar los puertos para la estrategia.
4. Edite las propiedades de la estrategia y haga clic en **Siguiente**.
5. Elija una operación y haga clic en **Siguiente**.
6. Configure las propiedades de la operación y seleccione puertos de salida para los datos analizados correctamente.
7. Si lo desea, haga clic en **Siguiente** para añadir más operaciones a la estrategia.
8. Después de añadir todas las operaciones a la estrategia, haga clic en **Finalizar**.
9. Si lo desea, añada más estrategias a la transformación.
10. Si lo desea, cambie el orden en que la transformación procesa las estrategias y las operaciones. Seleccione una estrategia u operación y haga clic en **Mover hacia arriba** o **Mover hacia abajo**.

# Configurar una estrategia de análisis de patrones

Para configurar una estrategia de análisis de patrones, abra una transformación de analizador en modo de análisis de patrones y seleccione la vista **Patrones**.

Antes de configurar la transformación para analizar patrones, compruebe que la vista **Patrones** muestra los nombres de los puertos de salida que esperaba. La transformación de analizador analiza tokens en los puertos de salida que seleccione. Cree puertos de salida adicionales si es necesario.

1. Seleccione la vista **Patrones**.
2. Añada uno o más patrones a la estrategia. Puede añadir patrones de las siguientes formas:
  - Introducir valores de datos para crear un patrón. Haga clic en **Nuevo** y seleccione **Nuevo patrón**.  
Si selecciona **Nuevo patrón** haga clic en **Especificar patrones aquí** y especifique uno o más tipos de token. Los tokens que ha introducido debe coincidir con la estructura de tokens de un campo de datos de entrada. Para añadir patrones tiene describir las estructuras de tokens del puerto de entrada.
  - Importar los valores de datos desde una tabla de referencia. Haga clic en **Nuevo** y seleccione **Nueva tabla de referencia**.  
Si selecciona **Nueva tabla de referencia**, busque en el repositorio una tabla de referencia y seleccione una columna que contenga una lista de las estructuras de tokens.
  - Importar valores de datos desde un conjunto de patrones. Haga clic en **Importar** y seleccione un conjunto de patrones reutilizable en el repositorio de modelos.  
Si selecciona **Importar**, busque los conjuntos de contenido definidos en el repositorio y seleccione un conjunto de patrones reutilizable.

**Nota:** puede usar el campo **Filtrar texto** para filtrar la lista de tablas de referencia y los conjuntos de patrones.

Puede mezclar conjuntos de patrones y tablas de referencia en los patrones de columna.
3. Asigne cada token de la columna de patrones a un puerto de salida.
  - Para asignar un token en un puerto de salida, haga doble clic en la columna de puerto y seleccione el nombre del token desde el menú.
  - Para analizar varios tokens en una única salida, haga doble clic en la columna de puerto y seleccione **Personalizado**. Asigne tokens al puerto y seleccione el delimitador que desee utilizar.

Asigne los tokens de cada fila del patrón a uno o más puertos de salida.
4. Guarde la transformación.

## Propiedades avanzadas de Transformación de analizador

Configure las propiedades para ayudarle a determinar cómo el servicio de integración de datos procesa los datos para la transformación de analizador.

Puede configurar niveles de seguimiento de registros.

Configure la siguiente propiedad en la ficha **Avanzadas**:

**Nivel de seguimiento**

Cantidad de detalles que aparecen en el registro de esta transformación. Puede elegir simplificado, normal, inicialización detallada o datos detallados. El valor predeterminado es Normal.

# Transformación de rango

Este capítulo incluye los siguientes temas:

- [Resumen de la transformación de rango, 295](#)
- [Puertos de la transformación de rango, 296](#)
- [Cómo definir grupos, 298](#)
- [Propiedades avanzadas de la transformación de rango, 298](#)

## Resumen de la transformación de rango

La transformación de rango es una transformación activa que limita registros a un rango superior o inferior. Utilice una transformación de rango para devolver el mayor o el menor valor numérico de un puerto o grupo. O bien use una transformación de rango para devolver las cadenas de la parte superior o inferior de un orden de clasificación de una asignación.

Durante la ejecución de una asignación, el servicio de integración de datos guarda en la memoria caché datos de entrada hasta que pueda realizar los cálculos de rango.

La transformación de rango difiere de las funciones de transformación MAX y MIN. La transformación de rango devuelve un grupo de valores de la parte superior o inferior, no solo un único valor. Por ejemplo, utilice una transformación de rango para seleccionar los diez primeros vendedores de un territorio en concreto. O bien, con el fin de generar un informe financiero, puede utilizar una transformación de rango para identificar los tres departamentos que menos gastan en salarios y gastos generales. Mientras que el lenguaje SQL ofrece muchas funciones diseñadas para manejar grupos de datos, no es posible realizar la identificación de estratos superiores o inferiores de un conjunto de filas haciendo uso de las funciones SQL estándar.

Conecte a la transformación todos los puertos que representen el mismo conjunto de filas. Las filas que queden dentro del rango, en función de algunas medidas que establezca al configurar la transformación, pasarán a través de la transformación de rango.

Como transformación activa, la transformación de rango puede modificar el número de filas que pasan a través de ella. Puede pasar 100 filas a la transformación de rango, pero solo puede hacer una selección para crear el rango de las 10 filas principales. Las diez filas superiores pasan de la transformación de rango a otra transformación.

No puede enlazar puertos de una transformación con la transformación de rango. A su vez, también puede crear variables locales y escribir expresiones de no agregado.

## Valores de las cadenas de establecimiento de rango

Puede configurar la transformación de rango para devolver los valores superiores o inferiores de un puerto de cadena. El servicio de integración de datos ordena las cadenas en función del orden de clasificación seleccionado para la asignación utilizada.

Cuando configure la aplicación que contiene la asignación, seleccione el orden de clasificación que el servicio de integración de datos utiliza para ejecutar la asignación. Puede seleccionar un lenguaje binario o un idioma específico, como el francés o el alemán. Si selecciona binario, el servicio de integración de datos calcula el valor binario de cada cadena y ordena las cadenas mediante los valores binarios. Si selecciona un idioma, el servicio de integración de datos ordena las cadenas por orden alfabético mediante el orden de clasificación del idioma.

## Memorias caché de rango

Durante la ejecución de una asignación, el servicio de integración de datos compara una fila de entrada con las filas de la memoria caché de datos. Si la fila de entrada deja fuera de rango una fila de la memoria caché, el servicio de integración de datos sustituye ésta por la fila de entrada. Si configura la transformación de rango a filas del grupo, el servicio de integración de datos establece un rango en las filas de cada grupo.

El servicio de integración de datos guarda la información de grupo en una memoria caché de índice, y los datos de las filas en una memoria caché de datos. Si crea varias particiones en un canal, el servicio de integración de datos crea distintas memorias caché para cada partición.

## Propiedades de la transformación de rango

Cuando crea una transformación de rango, puede configurar las siguientes propiedades:

- Introduzca un directorio de la memoria caché.
- Seleccione el rango superior o inferior.
- Seleccione el puerto de entrada/salida que contiene los valores que se usan para determinar el rango. Puede seleccionar solo un puerto para definir un rango.
- Seleccione el número de filas en las que desee crear un rango.
- Defina los grupos para crear un rango, como por ejemplo los diez productos menos caros de cada fabricante.

## Puertos de la transformación de rango

La transformación de rango incluye puertos de entrada, de entrada/salida o de salida que están conectados a otra transformación de la asignación. La transformación también incluye puertos de transferencia, de variable y de agrupamiento.

Una transformación de rango cuenta con los siguientes tipos de puertos:

### **Entrada**

Recibe los datos de las transformaciones de nivel superior. Usted puede diseñar puertos de entrada como puertos de entrada/salida. La transformación debe tener al menos un puerto de entrada.

### **Salida**

Pasa datos a las transformaciones de nivel inferior. Puede diseñar puertos de salida como puertos de entrada/salida. La transformación debe tener al menos un puerto de salida.

### Transferencia

Pasa datos que no se han modificado.

### Variable

Se utiliza para variables locales. Puede utilizar un puerto de variable para guardar valores o cálculos para utilizar en una expresión. Los puertos de variable no pueden ser ni puertos de entrada ni de salida. Pasan datos dentro de la transformación.

### Agrupar por

Indica cómo crear grupos. El puerto puede ser de entrada, de entrada/salida, de salida o de variable.

## Puerto de rango

El puerto de rango designa la columna en la que desea establecer un rango en sus valores.

Debe designar un puerto de entrada/salida o de salida como el puerto de rango. Por ejemplo, cree una transformación de rango para clasificar los cinco primeros clientes según el número total de ventas. El puerto TOTAL\_SALES contiene la cantidad total de ventas de cada cliente. Especifique el puerto de entrada/salida TOTAL\_SALES como el puerto de rango.

Debe enlazar el puerto de rango con otra transformación.

## Índice de rango

Developer Tool crea un puerto RANKINDEX para cada transformación de rango. El servicio de integración de datos utiliza el puerto de índice de rango para guardar la posición en el rango de cada fila del grupo.

Por ejemplo, puede crear una transformación de rango para identificar los 50 empleados que más cobran de la empresa. Identifique la columna SALARY como el puerto de entrada/salida que se utiliza para medir los rangos y configure la transformación para filtrar todas las filas y quedarnos solo con las 50 primeras.

Una vez que la transformación de rango identifique todas las filas que formen parte de un rango superior o inferior, asigna valores del índice de rango. En el caso de los 50 primeros empleados en función de su salario, el que más cobra recibe el índice de rango 1. El siguiente mejor pagado recibe el índice de rango 2, y así sucesivamente. Al medir un rango inferior, como por ejemplo los diez productos con el precio más bajo del inventario, la transformación de rango asigna un índice de rango en orden ascendente (del más bajo al más alto). Así, el producto más barato recibiría el índice de rango 1.

Si dos valores de rango coinciden, ambos reciben el mismo valor en el índice de rango y la transformación omite el siguiente valor. Por ejemplo, si desea ver las primero cinco tiendas del país y dos de ellas presentan las mismas ventas, los datos que se devuelvan pueden parecerse a:

RANKINDEX	SALES	STORE
1	10000	Orange
1	10000	Brea
3	90000	Los Angeles
4	80000	Ventura

RANKINDEX es solo un puerto de salida. Puede pasar el índice de rango a otra transformación de la asignación o directamente a un destino.

## Cómo definir grupos

Puede configurar la transformación de rango para crear grupos de filas con rango.

Por ejemplo, si desea seleccionar los diez productos más caros por fabricante, en primer lugar debería establecer un grupo para cada fabricante. Al configurar la transformación de rango, puede establecer uno de sus puertos de entrada, de entrada/salida o de salida como grupo por puerto. Para cada valor del puerto de grupo, la transformación crea un grupo de filas dentro de la definición de rango (superior o inferior y un número en concreto en cada rango).

La transformación de rango cambia el número de filas de dos formas diferentes. Al filtrar casi todas las filas de un rango superior o inferior, se reduce el número de filas que se transfieren a la transformación. En la definición de grupos, cree un grupo de filas con rango para cada grupo.

Por ejemplo, si crea una transformación de rango que clasifica los cinco mejores vendedores agrupados por trimestre, el índice de rango enumera los vendedores del 1 al 5 de cada trimestre:

RANKINDEX	SALES_PERSON	SALES	QUARTER
1	Sam	10,000	1
2	Mary	9,000	1
3	Alice	8,000	1
4	Ron	7,000	1
5	Alex	6,000	1

## Propiedades avanzadas de la transformación de rango

Configure propiedades para ayudar a determinar cómo el servicio de integración de datos procesa los datos para la transformación de rango.

Configure las siguientes propiedades en la ficha **Avanzadas**:

### Superior/Inferior

Especifica si desea el rango superior o inferior de una columna.

### Número de rangos

Número de filas para incluir en el rango superior o inferior.

### Comparación de cadenas que distinguen mayúsculas de minúsculas

Especifica si el servicio de integración de datos utiliza comparaciones de cadenas que distinguen mayúsculas de minúsculas cuando crea rangos de cadenas. Desactive esta opción para hacer que el servicio de integración de datos ignore esta distinción en las cadenas. Esta opción está seleccionada de forma predeterminada.

**Directorio de la memoria caché**

Directorio local donde el servicio de integración de datos crea los archivos de memoria caché del índice y los archivos de memoria caché de datos. Compruebe que el directorio existe y que tiene suficiente espacio en disco para los archivos de la memoria caché.

El valor predeterminado es el parámetro del sistema CacheDir.

**Tamaño de la memoria caché de datos de rango**

Tamaño de la memoria caché de datos para la transformación. Puede introducir un valor numérico en bytes o puede seleccionar Auto para hacer que el servicio de integración de datos determine el tamaño de la memoria caché en el tiempo de ejecución. El valor predeterminado es Auto.

**Tamaño de la memoria caché de índice de rango**

Tamaño de la memoria caché de índice para la transformación. Puede introducir un valor numérico en bytes o puede seleccionar Auto para hacer que el servicio de integración de datos determine el tamaño de la memoria caché en el tiempo de ejecución. El valor predeterminado es Auto.

**Nivel de seguimiento**

Cantidad de detalles que aparecen en el registro de esta transformación. Puede elegir simplificado, normal, inicialización detallada o datos detallados. El valor predeterminado es Normal.

## CAPÍTULO 26

# Transformación de enrutador

Este capítulo incluye los siguientes temas:

- [Resumen de la transformación de enrutador, 300](#)
- [Cómo trabajar con grupos, 300](#)
- [Cómo trabajar con puertos, 303](#)
- [Cómo conectar transformaciones de enrutador de una asignación, 303](#)
- [Propiedades avanzadas de la transformación de enrutador, 303](#)

## Resumen de la transformación de enrutador

La transformación de enrutador es una transformación activa que enruta los datos en varias transformaciones en función de una o varias condiciones de grupo.

Una transformación de enrutador es similar a una de filtro, ya que ambas utilizan una condición para comprobar datos. Una transformación de filtro comprueba los datos para aplicar una condición y anula las filas de datos que no cumplen dicha condición. Una transformación de enrutador comprueba los datos para aplicar una o varias condiciones y puede enrutar las filas de datos que no cumplen ninguna de las condiciones en un grupo de salida predeterminado.

Si necesita comprobar los mismos datos de entrada en función de varias condiciones, utilice una transformación de enrutador en una asignación en lugar de crear varias transformaciones de filtro para llevar a cabo la misma tarea. La transformación de enrutador resulta más eficaz. Por ejemplo, para comprobar los datos basándose en tres condiciones, puede usar una transformación de enrutador en lugar de tres transformaciones de filtro. Cuando se utiliza una transformación de enrutador en una asignación, el servicio de integración de datos procesa una vez los datos de entrada. Si utiliza varias transformaciones de filtro en una asignación, el servicio de integración de datos procesa los datos de entrada para cada transformación.

Una transformación de enrutador está formada por grupos de entrada y salida, puertos de entrada y salida, condiciones del filtro de grupo y propiedades avanzadas que usted configura en Developer Tool.

## Cómo trabajar con grupos

Una transformación de enrutador tiene los siguientes tipos de grupos:

- Entrada
- Salida

## Grupo de entrada

La transformación de rango incluye un grupo de entrada simple. El grupo de entrada incluye todos los puertos de entrada que agregue a la transformación.

## Grupos de salida

La transformación de rango incluye los siguientes tipos de grupos de salida:

### Grupos definidos por el usuario

Cree un grupo definido por el usuario para probar una condición en función de los datos de entrada. Un grupo definido por el usuario está formado por puertos de salida y una condición del filtro de grupo. Puede crear y editar grupos definidos por el usuario en la ficha **Grupos** mediante Developer Tool. Cree un grupo definido por el usuario para cada condición que desee especificar.

El servicio de integración de datos utiliza la condición para evaluar cada una de las filas de los datos de entrada. Comprueba las condiciones de cada grupo definido por el usuario antes de procesar el grupo predeterminado. El servicio de integración de datos determina el orden de evaluación de cada condición en función del orden de los grupos de salida conectados. El servicio de integración de datos procesa los grupos definidos por el usuario que están conectados a una transformación o a un destino en una asignación.

Si una fila cumple más de una condición del filtro de grupo, el servicio de integración de datos pasa esta fila varias veces.

### El grupo predeterminado

Tras la creación de un grupo definido por el usuario, Developer Tool crea el grupo predeterminado. Developer Tool no le permite editar ni eliminar el grupo predeterminado. Este grupo no tiene una condición del filtro de grupo asociada a él. Si todas las condiciones de grupo dan como resultado FALSE, el servicio de integración de datos pasa la fila al grupo predeterminado. Si desea que el servicio de integración de datos anule todas las filas del grupo predeterminado, no lo conecte a una transformación ni a un destino de una asignación.

Developer Tool eliminará el grupo predeterminado cuando elimine de la lista el último grupo definido por el usuario.

Developer Tool copia información de la propiedad desde los puertos de entrada del grupo de entrada para crear un grupo de puertos de salida para cada grupo de salida. No puede cambiar ni eliminar los puertos de salida ni sus propiedades.

## Cómo utilizar las condiciones del filtro de grupo

Puede realizar una prueba de los datos en función de una o varias condiciones del filtro de grupo. Puede crear condiciones del filtro de grupo en la ficha **Grupos** mediante el editor de expresiones.

Puede especificar cualquier expresión que devuelva un valor simple. A su vez, también puede introducir una constante para la condición. Una condición del filtro de grupo devuelve para cada fila que pase a través de la transformación los valores TRUE o FALSE, dependiendo de si la fila cumple la condición que se ha especificado. Cero (0) es equivalente a FALSE. Cualquier otro valor que no sea cero equivale a TRUE. El servicio de integración de datos pasa las filas de datos que devuelven el resultado TRUE a cada transformación o destino que está asociado a cada grupo definido por el usuario.

Por ejemplo, tiene cliente en nueve países distintos y desea realizar diferentes cálculos con los datos de tres países. Puede utilizar una transformación de enrutador de una asignación para filtrar dichos datos a tres transformaciones de expresión distintas.

El grupo predeterminado no tiene una condición del filtro de grupo. Sin embargo, puede crear una transformación de expresión para realizar un cálculo en función de los datos de los otros seis países.

Para realizar varios cálculos con los datos de tres países diferentes, cree tres grupos definidos por el usuario y especifique tres condiciones del filtro de grupo en la ficha **Grupos**.

En la siguiente tabla se muestran las condiciones del filtro de grupo que filtran los datos del cliente:

Nombre de grupo	Condición del filtro de grupo
Francia	customer_name='France'=TRUE
Japón	customer_name='Japan'=TRUE
EE. UU.	customer_name='USA'=TRUE

En la asignación, el servicio de integración de datos pasa las filas de datos que devuelven el resultado TRUE en cada transformación o destino asociado con cada grupo definido por el usuario, como por ejemplo Japón, Francia y EE. UU. El servicio de integración de datos pasa la fila al grupo predeterminado si el resultado de todas las condiciones es FALSE. A continuación, el servicio de integración de datos pasa los datos de los otros seis países a la transformación o al destino asociado al grupo predeterminado. Si desea que el servicio de integración de datos anule todas las filas del grupo predeterminado, no lo conecte a una transformación ni a un destino de una asignación.

La transformación de enrutador pasa los datos a través de cada grupo que cumpla la condición. Si los datos cumplen tres condiciones del grupo de salida, la transformación de enrutador pasa los datos a través de tres grupos de salida.

Por ejemplo, puede configurar las siguientes condiciones de grupo en una transformación de enrutador:

Nombre de grupo	Condición del filtro de grupo
Grupo de salida 1	employee_salary > 1000
Grupo de salida 2	employee_salary > 2000

Si la transformación de enrutador procesa datos de una fila de entrada con employee\_salary=3000, enruta los datos a través de los grupos de salida 1 y 2.

## Cómo añadir grupos

Al añadir un grupo, Developer Tool copia la información de las propiedades de los puertos de entrada en los de salida.

1. Haga clic en la ficha **Grupos**.
2. Haga clic en el botón **Nuevo**.
3. Introduzca un nombre para el grupo en la sección **Nombre de grupo**.
4. Haga clic en el campo **Condición del filtro de grupo** para abrir el **Editor de expresiones**.
5. Introduzca la condición del filtro de grupo.
6. Haga clic en **Validar** para comprobar la sintaxis de la condición.
7. Haga clic en **Aceptar**.

## Cómo trabajar con puertos

Una transformación de enrutador tiene puertos de entrada y de salida. Los puertos de entrada se encuentran en el grupo de entrada, mientras que los de salida están en los grupos de salida.

Puede crear puertos de entrada copiándolos de otra transformación o creándolos de forma manual en la ficha **Puertos**.

Developer Tool crea puertos de salida copiando las siguientes propiedades de los puertos de entrada:

- Nombre de puerto
- Tipo de datos
- Precisión
- Escala
- Valor predeterminado

Al realizar modificaciones en los puertos de entrada, Developer Tool actualiza los puertos de salida para reflejar los cambios. No puede editar ni eliminar los puertos de salida.

Developer Tool crea nombres de puertos de salida en función de los nombres de los puertos de entrada. Para cada puerto de entrada, Developer Tool crea un puerto de salida correspondiente en cada grupo de salida.

## Cómo conectar transformaciones de enrutador de una asignación

Cuando conecte transformaciones a una transformación de enrutador en una asignación, tenga en cuenta las siguientes reglas:

- Puede conectar un grupo a una transformación o destino.
- Puede conectar un puerto de salida de un grupo a varias transformaciones o a varios destinos.
- Puede conectar varios puertos de salida de un grupo a varias transformaciones o a varios destinos.
- No puede conectar más de un grupo a un destino o a una transformación de grupo de entrada simple.
- No puede conectar más de un grupo a una transformación de varios grupos de entrada, excepto en el caso de transformaciones de incorporación, en las que puede conectar cada grupo de salida a un grupo de entrada diferente.

## Propiedades avanzadas de la transformación de enrutador

Configure las propiedades para ayudar a determinar cómo el servicio de integración de datos procesa los datos para la transformación de enrutador.

Puede configurar niveles de seguimiento de registros.

Configure la siguiente propiedad en la ficha **Avanzadas**:

**Nivel de seguimiento**

Cantidad de detalles que aparecen en el registro de esta transformación. Puede elegir simplificado, normal, inicialización detallada o datos detallados. El valor predeterminado es Normal.

# Transformación de ordenación

Este capítulo incluye los siguientes temas:

- [Resumen de la transformación de ordenación, 305](#)
- [Cómo desarrollar una transformación de incorporación, 306](#)
- [Propiedades avanzadas de la transformación de ordenación, 306](#)
- [Tamaño de la memoria caché de ordenación, 307](#)
- [Puertos de transformación de ordenación, 308](#)
- [Cómo crear una transformación de ordenación, 308](#)
- [Ejemplo de transformación de ordenación, 309](#)

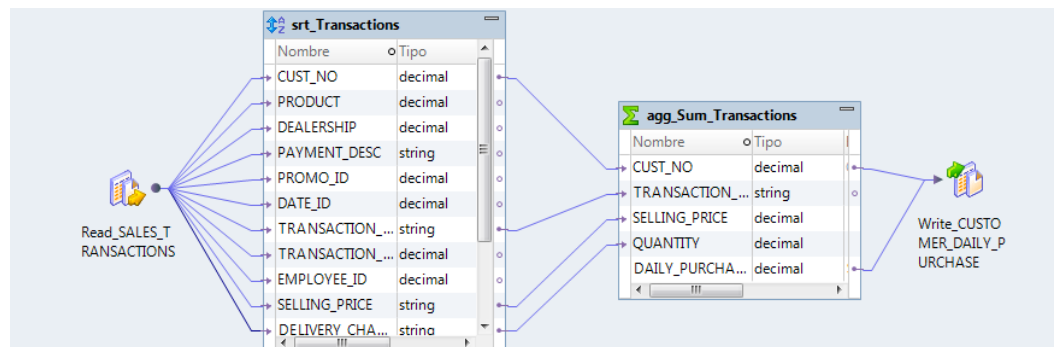
## Resumen de la transformación de ordenación

Utilice una transformación de ordenación para ordenar datos de forma ascendente o descendente de acuerdo con una clave de ordenación especificada. Puede configurar la transformación de ordenación para la ordenación con distinción de mayúsculas y minúsculas y, para la salida distinta. La transformación de ordenación es una transformación activa.

Cuando se crea una transformación de ordenación se especifican puertos como claves de ordenación, y se configura cada uno de los puertos de clave de ordenación para que clasifique en orden ascendente o descendente. El servicio de integración de datos ordena cada puerto secuencialmente cuando se especifican varios puertos para la clave de ordenación.

Por ejemplo, necesita crear una factura para el total de ventas de clientes a partir de una base de datos de clientes. Utilice una transformación de ordenación en la tabla de ventas de clientes para ordenar los datos en orden descendente según el número de cliente. Utilice el resultado de la transformación de ordenación como entrada de la transformación de agregación. Puede aumentar el rendimiento de una transformación de agregación con la opción de entrada ordenada.

La siguiente figura muestra la asignación:



## Cómo desarrollar una transformación de incorporación

Cuando se desarrolla una transformación de incorporación, se deben tener en cuenta factores como los puertos de clave de ordenación, filas de salida distintas y los criterios de ordenación con distinción de mayúsculas y minúsculas.

Tenga en cuenta los siguientes factores cuando desarrolle una transformación de ordenación:

- Los puertos que se desea configurar como claves de ordenación y la dirección de la ordenación.
- Si se desea una ordenación que distinga mayúsculas de minúsculas.
- Si se desea tener en cuenta valores nulos como prioridad de ordenación.
- Si se desean filas de salida distintas.
- El valor del tamaño de la memoria caché que se desea establecer.

## Propiedades avanzadas de la transformación de ordenación

En las propiedades avanzadas de la transformación de ordenación se pueden especificar criterios de clasificación adicionales. El servicio de integración de datos aplica las propiedades a todos los puertos de clave de clasificación. Las propiedades de la transformación de ordenación también determinan los recursos del sistema que asigna el servicio de integración de datos cuando clasifica datos.

La siguiente sección describe las propiedades avanzadas para una transformación de ordenación:

### Distintas

Trata las filas de salida como filas distintas. Si se configura la transformación de ordenación para filas de salida distintas, Developer Tool configura todos los puertos como parte de la clave de clasificación. El servicio de integración de datos descarta las filas duplicadas que se han comparado durante la operación de ordenación.

#### **Distinguir mayúsculas de minúsculas**

Determina si el servicio de integración de datos tiene en cuenta el formato de mayúsculas y minúsculas en la clasificación de los datos. Cuando se habilita la propiedad Distinguir mayúsculas de minúsculas, el servicio de integración de datos ordena antes los caracteres en mayúsculas que los caracteres en minúsculas. De forma predeterminada, Developer Tool define Distinguir mayúsculas de minúsculas.

#### **Valor nulo como inferior**

Trata un valor nulo como inferior a cualquier otro valor. Habilite la propiedad si desea que el servicio de integración de datos trate un valor como inferior respecto a cualquier otro valor cuando realiza la operación de ordenación.

#### **Tamaño de la memoria caché de ordenación**

Determina la cantidad máxima de memoria necesaria para realizar la operación de ordenación. El servicio de integración de datos pasa todos los datos entrantes a la transformación de ordenación antes de realizar la operación de ordenación. El valor predeterminado de la propiedad Tamaño de la memoria caché de ordenación es Auto. Puede configurar un valor numérico para la memoria caché de ordenación.

#### **Directorio de trabajo**

El servicio de integración de datos utiliza el directorio de trabajo para crear archivos temporales cuando ordena datos. Después de que el servicio de integración de datos ordena los datos, elimina los archivos temporales. Puede especificar cualquier directorio en el equipo donde se ejecuta el servicio de integración de datos para usarlo como directorio de trabajo. El valor predeterminado es el parámetro del sistema TempDir.

#### **Nivel de seguimiento**

Cantidad de detalles que aparecen en el registro de esta transformación. Puede elegir simplificado, normal, inicialización detallada o datos detallados. El valor predeterminado es Normal.

## **Tamaño de la memoria caché de ordenación**

El servicio de integración de datos pasa todos los datos entrantes a la transformación de ordenación antes de realizar la operación de ordenación.

Si se establece el Tamaño de memoria caché de ordenación como Auto, el servicio de integración de datos determina el tamaño de la memoria caché en tiempo de ejecución. Antes de iniciar la operación de ordenación, el servicio de integración de datos asigna la cantidad de memoria configurada para el tamaño de memoria caché de ordenación. Si el servicio de integración de datos no puede asignar suficiente memoria, la asignación genera un error.

Para un rendimiento óptimo, configure la memoria caché de ordenación con un valor menor o igual que la cantidad de memoria física RAM del equipo donde reside el servicio de integración de datos. Asigne como mínimo 16 MB (16.777.216 bytes) de memoria física para ordenar datos con una transformación de ordenación. De forma predeterminada, el tamaño de la memoria caché de ordenación se establece como Auto.

Si la cantidad de datos entrantes es mayor que la cantidad de memoria caché de ordenación, el servicio de integración de datos almacena datos temporalmente en el directorio de trabajo de la transformación de ordenación. El servicio de integración de datos necesita un espacio de disco de como mínimo el doble de la cantidad de datos entrantes cuando ordena datos en el directorio de trabajo. Si la cantidad de datos entrantes es considerablemente mayor que el tamaño de memoria de caché de ordenación, el servicio de integración de datos puede requerir más del doble del espacio de disco disponible para el directorio de trabajo.

# Puertos de transformación de ordenación

Todos los puertos de la transformación de ordenación permiten entrar datos, ordenar y enviar datos a otras transformaciones.

La transformación de ordenación tiene los siguientes tipos de puerto:

## Clave

La clave de ordenación es uno o más puertos que se desean utilizar como criterios de ordenación. El servicio de integración de datos ordena los datos según los puertos especificados como clave. Configure la dirección del puerto de clave para que ordene datos en orden ascendente o descendente. Developer Tool establece el valor de dirección predeterminado como ascendente.

## Otros

Los puertos que no se especifican como clave de ordenación.

# Cómo crear una transformación de ordenación

Puede crear una transformación de ordenación reutilizable o no reutilizable.

## Cómo crear una transformación de ordenación reutilizable

Cree una transformación de ordenación reutilizable para utilizarla en varias asignaciones o mapplets.

1. Seleccione un proyecto o una carpeta en la vista **Explorador de objetos**.
2. Haga clic en **Archivo > Nueva > Transformación**.  
A continuación, aparece el cuadro de diálogo **Nueva**.
3. Seleccione la transformación de ordenación.
4. Haga clic en **Siguiente**.
5. Especifique un nombre para la transformación.
6. Haga clic en **Finalizar**.  
La transformación aparece en el editor.
7. Haga clic en **Nueva** para añadir un puerto a la transformación.
8. Edite el puerto para definir el nombre, el tipo de datos y la precisión.
9. Seleccione **Clave** para indicar el puerto como clave de ordenación.
10. Haga clic en la vista **Avanzadas** y edite las propiedades de la transformación.

## Cómo crear una transformación de ordenación no reutilizable

Cree una transformación de ordenación no reutilizable como parte de una asignación o de un mapplet.

1. En una asignación o en un mapplet, arrastre una transformación de ordenación desde la paleta Transformación hasta el editor.  
La transformación aparece en el editor.
2. En la vista **Propiedades**, edite el nombre y la descripción de la transformación.
3. En la ficha **Puertos**, haga clic en **Nuevo** para añadir puertos a la transformación.

4. Edite los puertos para definir el nombre, el tipo de datos y la precisión.
5. Seleccione **Clave** para indicar el puerto como clave de ordenación.
6. Haga clic en la vista **Avanzadas** y edite las propiedades de la transformación.

## Ejemplo de transformación de ordenación

Usted tiene una tabla de base de datos PRODUCT\_ORDERS que contiene información sobre todos los pedidos efectuados por el cliente.

ORDER_ID	ITEM_ID	ITEM	QUANTITY	PRICE
43	123456	ItemA	3	3.04
41	456789	ItemB	2	12.02
43	000246	ItemC	6	34.55
45	000468	ItemD	5	0.56
41	123456	ItemA	4	3.04
45	123456	ItemA	5	3.04
45	456789	ItemB	3	12.02

Utilice la transformación de ordenación en PRODUCT\_ORDERS y especifique ORDER\_ID como clave de ordenación descendente.

Después de ordenar los datos, el servicio de integración de datos pasa las siguientes filas fuera de la transformación de ordenación:

ORDER_ID	ITEM_ID	ITEM	QUANTITY	PRICE
45	000468	ItemD	5	0.56
45	123456	ItemA	5	3.04
45	456789	ItemB	3	12.02
43	123456	ItemA	3	3.04
43	000246	ItemC	6	34.55
41	456789	ItemB	2	12.02
41	123456	ItemA	4	3.04

Necesita averiguar el importe total y la cantidad de artículos de cada pedido. Puede utilizar el resultado de la transformación de ordenación como entrada de la transformación de agregación. Utilice la entrada ordenada en la transformación de agregación para aumentar el rendimiento.

Si no se utiliza la entrada ordenada, el servicio de integración de datos realiza los cálculos de agregados a medida que lee las filas. El servicio de integración de datos almacena datos para cada grupo hasta que lee el origen por completo para asegurarse de que todos los cálculos de agregados se realizan de forma

precisa. Si se utiliza la entrada ordenada y no se ordenan previamente los datos correctamente, se obtienen resultados inesperados.

La transformación de agregación tiene los puertos agrupar por ORDER\_ID e ITEM con la opción de entrada ordenada seleccionada. Cuando se pasan los datos desde la transformación de ordenación, la transformación de agregación agrupa ORDER\_ID para calcular el importe total de cada pedido.

ORDER_ID	SUM
45	54.06
43	216.42
41	36.2

## CAPÍTULO 28

# Transformación de SQL

Este capítulo incluye los siguientes temas:

- [Resumen de la transformación de SQL, 311](#)
- [Propiedades avanzadas de Transformación de SQL, 312](#)
- [Puertos de transformación de SQL, 313](#)
- [Consulta de transformación de SQL, 316](#)
- [Cardinalidad entre fila de entrada y fila de salida, 317](#)
- [Optimización del filtro con la transformación de SQL, 321](#)
- [Ejemplo de transformación de SQL, 323](#)

## Resumen de la transformación de SQL

La transformación de SQL procesa consultas SQL en la parte intermedia de una asignación. Puede insertar, eliminar, actualizar y recuperar filas de una base de datos. Puede ejecutar instrucciones DDL de SQL para crear una tabla o soltar una tabla en la parte intermedia de una asignación.

Una transformación de SQL ejecuta una consulta SQL que se define en el editor de transformaciones de SQL. La transformación de SQL procesa la consulta, devuelve filas y devuelve cualquier error de base de datos que se produce. Puede pasar valores de puerto de entrada a parámetros en la consulta.

La transformación de SQL es una transformación activa. La transformación puede devolver múltiples filas para cada fila de entrada.

Cuando configure una transformación de SQL, realice las siguientes tareas:

1. Defina las propiedades de la transformación, incluido el tipo de base de datos al que conectarse.
2. Defina los puertos.
3. Cree una consulta SQL en el editor de SQL.
4. Configure la transformación de SQL en una asignación conectando puertos situados en un nivel superior.
5. Obtenga una vista previa de los datos para comprobar los resultados.

# Propiedades avanzadas de Transformación de SQL

Es posible modificar las propiedades de una transformación de SQL en cualquier momento. El tipo de base de datos predeterminado es Oracle. Si tiene que conectarse a otro tipo de base de datos, cambie el tipo de base de datos antes de añadir puertos a la transformación.

Configure las siguientes propiedades en la ficha **Avanzadas**:

## Nivel de seguimiento

Cantidad de detalles que aparecen en el registro de esta transformación. Puede elegir simplificado, normal, inicialización detallada o datos detallados. El valor predeterminado es Normal. Cuando se configura el nivel de seguimiento de la transformación de SQL como Datos detallados, el servicio de integración de datos escribe cada consulta SQL que prepara en el registro de la asignación.

## Tipo de conexión.

Describe cómo se conecta el servicio de integración de datos con la base de datos. El tipo de base de datos es estática. El servicio de integración de datos se conecta una vez a la base de datos. Seleccione un objeto de conexión a base de datos en la transformación de SQL. Solo lectura.

## Tipo de base de datos

Tipo de base de datos a la que se conecta la transformación de SQL. Elija un tipo de base de datos de la lista. Puede seleccionar Oracle, Microsoft SQL Server, IBM DB2 o ODBC. El tipo de base de datos influye en los tipos de datos que se pueden asignar en la ficha **Puertos**. Cuando se cambia el tipo de base de datos, Developer Tool cambia los tipos de datos de los puertos de entrada, de salida y de transferencia.

## Continuar si hay errores en una fila

Continúa procesando las instrucciones SQL restantes de una consulta después de que se produzca un error de SQL.

## Incluir estadísticas como salida

Añade un puerto de salida NumRowsAffected. El puerto devuelve el número total de filas de la base de datos que se ven afectadas por las instrucciones de consulta INSERT, DELETE y UPDATE para una fila de entrada.

## Recuento máximo de filas de salida

Define el número máximo de filas que proporciona la transformación de SQL tras una consulta SELECT. Para configurar un número ilimitado de filas, establezca en cero el valor de Recuento de filas de salida máximo.

## Descripción de la consulta

Descripción de la consulta SQL que se define en la transformación.

## Modo SQL

Determina si la consulta SQL es una secuencia de comandos externa o si la consulta se define en la transformación. El modo de SQL es Consulta. La transformación de SQL ejecuta una consulta que se define en el Editor de SQL. Solo lectura.

## Consulta SQL

Muestra la consulta SQL que se configura en el Editor SQL.

## Tiene efectos secundarios

Casilla de verificación que indica que la transformación SQL realiza cualquier función además de devolver filas. La transformación SQL tiene efectos secundarios cuando la consulta SQL actualiza una

base de datos. Cuando la consulta SQL contiene una instrucción como CREATE, DROP, INSERT, UPDATE, GRANT o REVOKE, la consulta SQL contiene efectos secundarios.

La transformación SQL también tiene efectos secundarios si la transformación devuelve filas NULL para las instrucciones SELECT que no devuelven ningún resultado. Las filas pueden contener valores de puerto de transferencia, información de errores de SQL o el campo NUMRowsAffected.

Deshabilite la propiedad **Tiene efectos secundarios** para permitir la optimización de inserción o la optimización de primera selección. Esta propiedad está habilitada de manera predeterminada.

#### Devolver solo salida de base de datos

La transformación SQL no genera filas para las instrucciones SELECT que devuelven 0 resultados, filas para otras instrucciones como INSERT, UPDATE, DELETE o COMMIT o filas nulas.

#### Habilitar la optimización de inserción

Habilita el servicio de integración de datos para insertar la lógica desde una transformación de filtro en la asignación en SQL para la transformación SQL.

## Puertos de transformación de SQL

Cuando se crea una transformación de SQL, Developer Tool crea el puerto SQLError de forma predeterminada. Añada puertos de entrada, puertos de salida y puertos de transferencia en la vista **Puertos**.

La transformación de SQL es compatible con los siguientes tipos de puerto:

#### Entrada

Recibe datos de origen que pueden usarse en una consulta SQL.

#### Salida

Devuelve datos de base de datos desde una consulta SQL SELECT.

#### Transferencia

Los puertos de entrada-salida que pasan datos de origen a través de la transformación sin modificarlos.

#### SQLError

Devuelve errores SQL desde la base de datos. Si no se produce ningún error, se devuelve NULL.

#### NumRowsAffected

El puerto devuelve el número total de filas de la base de datos que se ven afectadas por las instrucciones de consulta INSERT, DELETE y UPDATE para una fila de entrada. Developer Tool crea este puerto cuando se opta por incluir las estadísticas de actualización en la fila de salida.

## Puertos de entrada

Puede crear puertos de entrada en la transformación de SQL para los datos que no se prevé pasar a puertos de salida. Se puede hacer referencia a puertos de entrada en una consulta SQL e incluir nombres de puerto como parámetros en la consulta.

Para añadir puertos de entrada, haga clic en **Entrada**, en la vista **Puertos**. Haga clic en **Nuevo**.

Cuando añada el puerto, especifique el tipo de datos nativo para el puerto. El tipo de datos nativo es el tipo de datos válido para la base de datos a la que se está conectando. Cuando se configura un tipo de datos nativo, aparece un tipo de datos de transformación. Si arrastra filas hasta la transformación de SQL, Developer Tool establece el tipo de datos nativo basándose en los tipos de datos válidos para la base de

datos a la que se está conectando. Compruebe que los tipos de datos de las columnas que se usan en la consulta sean los mismos tipos de datos que las columnas de la base de datos.

**Nota:** Si selecciona **Copiar en salida** para un puerto, el puerto de entrada se convierte en un puerto de transferencia. Los puertos de transferencia aparecen en las secciones **Entrada** y **Salida** de la vista **Puertos**.

## Puertos de salida

Los puertos de salida devuelven valores desde una instrucción SELECT. Cree un puerto de salida para cada columna de la instrucción SELECT.

Cuando configure un puerto de salida, elija el tipo de datos nativo del puerto. El tipo de datos nativo de un puerto de salida debe coincidir con el tipo de datos de la columna correspondiente en la base de datos. Cuando se configura el tipo de datos nativo, Developer Tool define el tipo de datos de la transformación para el puerto.

Por ejemplo, la transformación de SQL contiene la siguiente consulta SQL para una base de datos Oracle:

```
SELECT FirstName, LastName, Age FROM EMPLOYEES
```

Puede configurar los siguientes puertos de salida y los tipos de datos nativos en la transformación de SQL:

Puerto de salida	Tipos de datos nativos	Tipos de datos de transformación
FirstNm	varchar2	string
LastNm	varchar2	string
Age	number	double

El número de puertos de salida y el orden de los mismos debe coincidir con el número y el orden de las columnas que devuelve la instrucción SELECT de la consulta. Cuando el número de puertos de salida es mayor que el número de columnas en la cláusula SELECT, los puertos adicionales devuelven un valor nulo. Cuando el número de puertos de salida es menor que el número de columnas en la cláusula SELECT, el servicio de integración de datos genera un error de fila.

Si se cambia el tipo de base de datos a la que se conecta la transformación, Developer Tool cambia los tipos nativos de los puertos de salida. Developer Tool puede no elegir el tipo de datos correcto para todos los puertos. Si se cambia el tipo de base de datos, compruebe que el tipo de datos nativo para cada puerto de salida sea el mismo tipo de datos que la columna de la base de datos. Por ejemplo, Developer Tool puede elegir nVarchar2 para una columna de base de datos Oracle. Es posible que tenga que cambiar el tipo de datos a varchar2.

Configure puertos de salida en la vista **Puertos** de la transformación de SQL.

## Puertos de transferencia

Los puertos de transferencia son puertos de entrada-salida que pasan datos a través de la transformación sin cambiar los datos. La transformación de SQL devuelve datos en los puertos de transferencia independientemente de que una consulta SQL devuelva filas o no.

Cuando la fila de entrada contiene una instrucción SELECT, la transformación de SQL devuelve los datos en el puerto de transferencia para cada fila que devuelve desde la base de datos. Si el resultado de la consulta contiene varias filas, la transformación de SQL repite los datos de transferencia en cada fila.

Cuando una consulta no devuelve ninguna fila, la transformación de SQL devuelve los datos de columna de transferencia con valores nulos en las columnas de salida. Por ejemplo, las consultas que contienen instrucciones INSERT, UPDATE y DELETE no devuelven ninguna fila. Cuando la consulta genera errores, la transformación de SQL devuelve los datos de columna de transferencia, el mensaje SQLException y valores nulos en los puertos de salida.

No se puede configurar un puerto de transferencia para que devuelva datos desde una consulta SELECT.

Para crear un puerto de transferencia, cree un puerto de entrada y seleccione **Copiar en salida**. Developer Tool crea un puerto de salida y añade un sufijo "\_output" al nombre del puerto. No se puede cambiar el puerto de salida que Developer Tool crea para un puerto de transferencia. No se puede crear un puerto de salida con el sufijo "\_output".

## Puerto SQLException

El puerto SQLException devuelve errores SQL desde la base de datos.

Cuando la consulta SQL contiene errores de sintaxis, el puerto SQLException contiene el texto de error de la base de datos. Por ejemplo, la siguiente consulta SQL genera un error de fila desde una base de datos Oracle:

```
SELECT Product_ID FROM Employees
```

La tabla Employees no contiene Product\_ID. El servicio de integración de datos genera una fila. El puerto SQLException contiene el texto del error en una línea:

```
ORA-0094: "Product_ID": invalid identifier Database driver error... Function Name:
Execute SQL Stmt: SELECT Product_ID from Employees Oracle Fatal Error
```

Puede configurar varias instrucciones de consulta en la consulta SQL. Cuando se configure la transformación de SQL para que continúe ante un error SQL, la transformación de SQL puede devolver filas para una instrucción de consulta, aunque devuelve errores de base de datos para otra instrucción de consulta. La transformación de SQL devuelve cualquier error de base de datos en una fila aparte.

## Número de filas afectadas

Habilite el puerto de salida NumRowsAffected para devolver el número de filas que las instrucciones UPDATE, INSERT o DELETE cambian para cada fila de entrada. El servicio de integración de datos devuelve NumRowsAffected para cada instrucción de la consulta. De forma predeterminada, NumRowsAffected está deshabilitado.

Cuando se habilita NumRowsAffected y la consulta SQL no contiene una instrucción INSERT, UPDATE o DELETE, NumRowsAffected es cero en cada fila de salida.

Cuando la consulta SQL contiene varias instrucciones, el servicio de integración de datos devuelve NumRowsAffected para cada instrucción. NumRowsAffected contiene el número de filas que las instrucciones INSERT, UPDATE y DELETE cambian para una fila de entrada.

Por ejemplo, una consulta contiene las siguientes instrucciones:

```
DELETE from Employees WHERE Employee_ID = '101';
SELECT Employee_ID, LastName from Employees WHERE Employee_ID = '103';
INSERT into Employees (Employee_ID, LastName, Address)VALUES ('102', 'Gein', '38 Beach Rd')
```

La instrucción DELETE afecta una fila. La instrucción SELECT no afecta ninguna fila. La instrucción INSERT afecta una fila.

El servicio de integración de datos devuelve una fila desde la instrucción DELETE. NumRowsAffected es igual a uno. El servicio de integración de datos devuelve una fila desde la instrucción SELECT y NumRowsAffected es cero. El servicio de integración de datos devuelve una fila desde la instrucción INSERT y NumRowsAffected es uno.

# Consulta de transformación de SQL

Cree una consulta SQL en el editor de SQL para recuperar filas de una base de datos, o para actualizar la base de datos.

Para crear una consulta, escriba la instrucción de la consulta en el editor de SQL en la vista SQL. El editor de SQL proporciona una lista de los puertos de transformación a los que se puede hacer referencia en la consulta. Puede hacer doble clic en un nombre de puerto para añadirlo como parámetro de consulta.

Cuando se crea una consulta, el editor de SQL valida los nombres de puerto de la consulta. También comprueba que los puertos utilizados para la sustitución de cadenas son tipos de datos String. El editor de SQL no valida la sintaxis de la consulta SQL.

Se puede crear una consulta SQL estática. La instrucción de la consulta no cambia, aunque puede incluir parámetros para cambiar valores. El servicio de integración de datos ejecuta la consulta para cada fila de entrada.

## Definir la consulta SQL

Define una consulta SQL que ejecuta las mismas instrucciones de consulta para cada fila de entrada. Puede cambiar las columnas de la consulta o de la tabla según los valores de puerto de entrada en la fila. También puede cambiar los valores de la cláusula WHERE según los valores de puerto de entrada.

Para cambiar los valores de los datos en la cláusula WHERE para cada fila de entrada, configure el enlace de parámetros.

Para cambiar las columnas de la consulta o cambiar la tabla según los valores de puerto de entrada, utilice la sustitución de cadenas.

## Enlace de parámetro

Para cambiar los datos de la consulta, configure los parámetros de la consulta y enlace estos con puertos de entrada de la transformación. Cuando enlace un parámetro con un puerto de entrada, identifique el puerto por su nombre en la consulta. El editor de SQL delimita el nombre del puerto mediante signos de interrogación (?). Los datos de la consulta cambian según el valor de los datos en el puerto.

Las siguientes consultas utilizan enlaces de parámetro:

```
DELETE FROM Employee WHERE Dept = ?Dept?
INSERT INTO Employee(Employee_ID, Dept) VALUES (?Employee_ID?, ?Dept?)
UPDATE Employee SET Dept = ?Dept? WHERE Employee_ID > 100
```

La siguiente consulta SQL tiene parámetros de consulta que se enlazan con los puertos de entrada Employee\_ID y Dept de una transformación de SQL:

```
SELECT Name, Address FROM Employees WHERE Employee_Num =?Employee_ID? and Dept = ?Dept?
```

El origen puede tener las siguientes filas:

Employee_ID	Dept
100	Products
123	HR
130	Accounting

El servicio de integración de datos genera las siguientes instrucciones de consulta a partir de las filas:

```
SELECT Name, Address FROM Employees WHERE Employee_ID = '100' and DEPT = 'Products'
SELECT Name, Address FROM Employees WHERE Employee_ID = '123' and DEPT = 'HR'
SELECT Name, Address FROM Employees WHERE Employee_ID = '130' and DEPT = 'Accounting'
```

## Sustitución de cadenas

Utilice variables de cadena para reemplazar componentes de instrucciones de consulta. Por ejemplo, puede utilizar la variable de cadena para reemplazar el nombre de una tabla en una consulta. También puede sustituir los nombres de columna en una instrucción SELECT.

Para sustituir el nombre de una tabla, configure un puerto de entrada para que reciba el nombre de la tabla desde cada fila de entrada. En el editor de SQL, seleccione el puerto desde la lista de puertos **Sustitución de cadenas**. Developer Tool identifica el puerto de entrada por el nombre en la consulta y delimita el nombre con una tilde (~).

La siguiente consulta contiene una variable de cadena , ~Table\_Port~:

```
SELECT Emp_ID, Address from ~Table_Port~ where Dept = 'HR'
```

El origen puede pasar los siguientes valores a la columna **Table\_Port**:

### Table\_Port

Employees\_USA

Employees\_England

Employees\_Australia

El servicio de integración de datos sustituye la variable ~Table\_Port~ con el valor del nombre de tabla del puerto de entrada:

```
SELECT Emp_ID, Address from Employees_USA where Dept = 'HR'
SELECT Emp_ID, Address from Employees_England where Dept = 'HR'
SELECT Emp_ID, Address from Employees_Australia where Dept = 'HR'
```

# Cardinalidad entre fila de entrada y fila de salida

Cuando el servicio de integración de datos ejecuta una consulta SELECT, la transformación de SQL devuelve una fila por cada fila que recupera. Cuando la consulta no recupera ningún dato, la transformación de SQL devuelve cero o una fila por cada fila de entrada.

### Cómo procesar instrucciones de consultas

Cuando la consulta SELECT se ejecuta correctamente, la transformación de SQL puede recuperar varias filas. Cuando la consulta contiene otras instrucciones, el servicio de integración de datos puede generar una fila que contenga errores de SQL o el número de filas afectadas.

### Configuración de puertos

El puerto de salida NumRowsAffected contiene el número de filas que la instrucción UPDATE, INSERT o DELETE cambia para una fila de entrada. La transformación de SQL devuelve el número de filas afectadas para cada instrucción de una consulta. Cuando la transformación de SQL contiene puertos de transferencia, la transformación devuelve los datos de columna al menos una vez para cada fila de origen.

### Configuración del recuento máximo de filas

Recuento máximo de filas de salida limita el número de filas que una transformación de SQL devuelve a partir de consultas SELECT.

### Filas de error

El servicio de integración de datos devuelve errores de fila cuando detecta errores de conexión o errores de sintaxis. La transformación de SQL devuelve errores al puerto SQLError.

### Continuar ante error de SQL

Puede configurar la transformación de SQL para que continúe procesando cuando se produce un error en una instrucción SQL. La transformación de SQL no genera un error de fila.

## Cómo procesar instrucciones de consultas

El tipo de consulta SQL determina cuántas filas devuelve la transformación SQL. La transformación SQL puede devolver ninguna, una o varias filas. Cuando la consulta contiene una instrucción SELECT, la transformación SQL devuelve cada columna de la base de datos en un puerto de salida. La transformación devuelve todas las filas elegibles.

La siguiente tabla enumera las filas de salida que genera la transformación SQL para distintos tipos de instrucciones de consulta cuando no se produce ningún error en el modo de consulta:

Instrucción de consulta	Filas de salida
UPDATE, INSERT, sólo DELETE	Una fila por cada instrucción de la consulta.
Una o más instrucciones SELECT.	Número total de filas recuperadas de la base de datos.
Consultas DDL como CREATE, DROP, TRUNCATE	Una fila por cada instrucción de la consulta.

## Configuración de puertos

Cuando se habilita Incluir estadísticas como salida, Developer Tool crea el puerto NumRowsAffected. El servicio de integración de datos devuelve, como mínimo, una fila con NumRowsAffected según las instrucciones de la consulta SQL.

La siguiente tabla enumera las filas de salida que genera la transformación de SQL si se habilita NumRowsAffected:

Instrucción de consulta	Filas de salida
UPDATE, INSERT, sólo DELETE	Una fila por cada instrucción con NumRowsAffected para la instrucción.
Una o más instrucciones SELECT.	Número total de filas recuperadas de la base de datos. NumRowsAffected es cero en cada fila.
Consultas DDL como CREATE, DROP, TRUNCATE	Una fila con ningún NumRowsAffected.

## Recuento máximo de filas de salida

Se puede limitar el número de filas que la transformación de SQL devuelve para consultas SELECT. Configure la propiedad **Recuento máximo de filas de salida** para limitar el número de filas. Cuando una consulta contiene varias instrucciones SELECT, la transformación de SQL limita el número total de filas de todas las instrucciones SELECT.

Por ejemplo, establezca **Recuento máximo de filas de salida** en 100. La consulta contiene dos instrucciones SELECT:

```
SELECT * FROM table1; SELECT * FROM table2;
```

Si la primera instrucción SELECT devuelve 200 filas, y la segunda instrucción SELECT devuelve 50 filas, la transformación de SQL devuelve 100 filas de la primera instrucción SELECT. La transformación de SQL no devuelve ninguna fila de la segunda instrucción.

**Para configurar un número ilimitado de filas, establezca en cero el valor de Recuento máximo de filas de salida.**

## Filas de error

El servicio de integración de datos devuelve errores de fila cuando detecta un error de conexión o un error de sintaxis. La transformación de SQL devuelve errores de SQL al puerto SQLError.

Cuando se configura un puerto de transferencia o el puerto NumRowsAffected, la transformación de SQL devuelve como mínimo una fila por cada fila de origen. Cuando una consulta no devuelve datos, la transformación de SQL devuelve los datos de transferencia y los valores de NumRowsAffected, pero devuelve valores nulos en los puertos de salida. Se pueden quitar filas con valores nulos haciendo pasar las filas de salida a través de una transformación de filtro.

La siguiente tabla describe las filas que genera la transformación de SQL para las instrucciones de consulta UPDATE, INSERT o DELETE:

Puerto NumRowsAffected o puerto de transferencia configurado	SQLError	Salida de filas
Ningún puerto configurado	No	Una fila con NULL en el puerto SQLError.
Ningún puerto configurado	Sí	Una fila con error en el puerto SQLError.
Cualquiera de los dos puertos configurados	No	Una fila por cada instrucción de consulta con los datos de columna de NumRowsAffected o del puerto de transferencia.
Cualquiera de los dos puertos configurados	Sí	Una fila con el error en el puerto SQLError, el puerto NumRowsAffected o los datos del puerto de transferencia.

La siguiente tabla describe el número de filas de salida que la transformación de SQL genera para las instrucciones SELECT:

Puerto NumRowsAffected o puerto de transferencia configurado	SQLError	Salida de filas
Ningún puerto configurado	No	Una o más filas, según las filas devueltas desde cada instrucción SELECT.
Ningún puerto configurado	Sí	Una fila mayor que la suma de filas de salida para las instrucciones correctas. La última fila contiene el error en el puerto SQLError.
Cualquiera de los dos puertos configurados	No	Una o más filas, según las filas devueltas para cada instrucción SELECT: <ul style="list-style-type: none"> <li>- Si NumRowsAffected está habilitado, cada fila contiene una columna NumRowsAffected con un valor cero.</li> <li>- Si se configura un puerto de transferencia, cada fila contiene los datos de la columna de transferencia. Cuando la consulta devuelve varias filas, los datos de la columna de transferencia se duplican en cada fila.</li> </ul>
Cualquiera de los dos puertos configurados	Sí	Una o más filas, según las filas devueltas para cada instrucción SELECT. La última fila contiene el error en el puerto SQLError: <ul style="list-style-type: none"> <li>- Cuando NumRowsAffected está habilitado, cada fila contiene una columna NumRowsAffected con un valor cero.</li> <li>- Si se configura un puerto de transferencia, cada fila contiene los datos de la columna de transferencia. Cuando la consulta devuelve varias filas, los datos de la columna de transferencia se duplican en cada fila.</li> </ul>

La siguiente tabla describe el número de filas de salida que genera la transformación de SQL para las consultas DDL como CREATE, DROP o TRUNCATE:

Puerto NumRowsAffected o puerto de transferencia configurado	SQLError	Salida de filas
Ningún puerto configurado	No	Una fila con NULL en el puerto SQLError.
Ningún puerto configurado	Sí	Una fila que contiene el error en el puerto SQLError.
Cualquiera de los dos puertos configurados	No	Una fila que incluye la columna NumRowsAffected con valor cero y los datos de columna de transferencia.
Cualquiera de los dos puertos configurados	Sí	Una fila con el error en el puerto SQLError, el puerto NumRowsAffected con valor cero y los datos de la columna de transferencia.

## Continuar ante error de SQL

Puede optar por omitir un error SQL que se produce en una instrucción de consulta. Habilite **Continuar ante error de SQL si hay errores en una fila**. El servicio de integración de datos continúa ejecutando las instrucciones restantes para la fila.

El servicio de integración de datos no genera un error de fila. Sin embargo, el puerto `SQL_Error` contiene la instrucción SQL que falló y mensajes de error.

Por ejemplo, una consulta puede tener las siguientes instrucciones:

```
DELETE FROM Persons WHERE FirstName = 'Ed';  
INSERT INTO Persons (LastName, Address) VALUES ('Gein', '38 Beach Rd')
```

Si la instrucción `DELETE` falla, la transformación de SQL devuelve un mensaje de error desde la base de datos. El servicio de integración de datos continúa procesando la instrucción `INSERT`.

Deshabilite la opción **Continuar ante error de SQL** para solucionar problemas de la base de datos y asociar los errores con las instrucciones de consulta que han generado los errores.

## Optimización del filtro con la transformación de SQL

El servicio de integración de datos puede aplicar una optimización del filtro con una transformación de SQL si la condición de filtro hace referencia solo a los puertos de transferencia y la transformación de SQL no tiene efectos secundarios.

La transformación de SQL tiene efectos secundarios en las siguientes circunstancias:

- La consulta SQL actualiza una base de datos. La consulta SQL contiene una instrucción como `CREATE`, `DROP`, `INSERT`, `UPDATE`, `GRANT` o `REVOKE`.
- La transformación devuelve filas nulas para las instrucciones `SELECT` que no devuelven ningún resultado. Las filas pueden contener valores de puerto de transferencia, información de errores de SQL o el campo `NUMRowsAffected`.

El servicio de integración de datos puede aplicar los métodos de optimización de primera selección e inserción con la transformación de SQL.

## Optimización de primera selección con la transformación de SQL

El servicio de integración de datos puede realizar la optimización de primera selección con una transformación de SQL si la condición de filtro hace referencia solo a los puertos de transferencia y la transformación de SQL no tiene efectos secundarios.

La transformación de SQL tiene efectos secundarios en las siguientes circunstancias:

- La consulta SQL actualiza una base de datos. La consulta SQL contiene una instrucción como `CREATE`, `DROP`, `INSERT`, `UPDATE`, `GRANT` o `REVOKE`.
- La transformación devuelve filas nulas para las instrucciones `SELECT` que no devuelven ningún resultado. Las filas pueden contener valores de puerto de transferencia, información de errores de SQL o el campo `NUMRowsAffected`.

## Habilitar la optimización de primera selección con la transformación de SQL

Habilite la optimización de primera selección en la transformación de SQL si la transformación de SQL no tiene ningún efecto secundario.

1. Habilite la opción **Devolver la base de datos de solo salida** en **Propiedades avanzadas** de la transformación de SQL.
2. Desactive **Tiene efectos secundarios** en **Propiedades avanzadas** de la transformación.
3. Si la transformación tiene un puerto **NumAffectedRows**, quite el puerto.

## Optimización de inserción con la transformación de SQL

Con la optimización de inserción, el servicio de integración de datos inserta la lógica de filtro desde una transformación de filtro de la asignación a la consulta de la transformación de SQL.

Utilice las siguientes reglas y directrices cuando habilite la optimización de inserción con la transformación de SQL:

- La transformación de la consulta SQL solo debe contener instrucciones SELECT.
- La consulta SQL de la transformación debe ser una subconsulta válida.
- La condición de filtro no puede hacer referencia a los campos de Error de SQL o NumRowsAffected.
- Los nombres de los puertos de salida deben coincidir con los nombres de las columnas de la instrucción SQL SELECT. Cuando hace referencia a un puerto de salida en una condición de filtro, el servicio de integración de datos inserta el nombre del puerto correspondiente en la consulta SQL. Puede agregar alias a SQL si las columnas de la consulta no coinciden con los nombres de puertos de salida. Por ejemplo, `SELECT mycolname1 AS portname1, mycolname2 AS portname2`.
- La transformación no puede tener efectos secundarios.

### Ejemplo de la optimización de inserción con la transformación SQL

Una transformación de SQL recupera pedidos por el ID del cliente. Una transformación de filtro que aparece tras la transformación de SQL solamente devuelve las filas en las que el importe del pedido es superior a 1.000.

El servicio de integración de datos inserta el siguiente filtro en una instrucción SELECT en la transformación de SQL:

```
orderAmount > 1000
```

Cada instrucción de la consulta SQL se convierte en otra subconsulta de la instrucción SELECT que contiene el filtro.

La siguiente instrucción de consulta muestra la instrucción de consulta original como una subconsulta de la instrucción SELECT:

```
SELECT <customerID>, <orderAmount>, ... FROM (original query statements) ALIAS WHERE  
<orderAmount> > 1000
```

Si la consulta SQL tiene varias instrucciones, cada instrucción está incluida en otra subconsulta. La subconsulta tiene la misma sintaxis, incluyendo la cláusula WHERE.

Los puertos *customerID* y *orderAmount* son los nombres de los puertos de salida en la transformación de SQL. La subconsulta no incluye puertos de transferencia, el error de SQL o los puertos de estadísticas de SQL. Si inserta varios filtros en la transformación de SQL, la cláusula WHERE contiene todos los filtros.

### Habilitar la optimización de inserción con la transformación de SQL

Habilitar la optimización de inserción mediante las propiedades de configuración en la ficha **Propiedades avanzadas** de la transformación de SQL.

1. Desactive **Tiene efectos secundarios**.
2. Habilite **Devolver la base de datos de solo salida**.
3. Establezca el **Recuento máximo de filas de salida** a cero.
4. Habilitar la optimización de inserción.

# Ejemplo de transformación de SQL

Usted es un desarrollador del departamento de RRHH de la empresa Hypostores. Hypostores mantiene la información de las nóminas de los empleados en una base de datos aparte de los datos de empleados de recursos humanos. El departamento de RRHH necesita consultar una vista individual de los empleados y los salarios en varias regiones.

Usted quiere crear una asignación de objeto de datos lógicos que muestre una vista individual de los datos de empleados y de salarios en un objeto de datos lógicos de empleados.

Cree una asignación de objeto de datos lógicos con el origen de datos de empleados. Incluya una transformación de SQL para recuperar el salario y la fecha de alta laboral desde la base de datos de nóminas.

## Asignación de objeto de datos lógicos

La asignación de objeto de datos lógicos contiene los siguientes objetos:

### Tabla Empleado

Tabla relacional de entrada de la base de datos Recursos Humanos.

### Tabla Salario

Una tabla de la base de datos Nómina que contiene el salario y la fecha de alta laboral del empleado. La base de datos es una base de datos Oracle.

### transformación de SQL

Transformación que recupera la fecha de alta laboral y el salario para cada fila de empleado. La transformación se conecta a una base de datos Nómina y ejecuta una consulta SQL en la tabla Salario de la base de datos.

### Objeto de datos lógicos

Contiene la vista combinada de los datos del empleado y del salario. El objeto de datos lógicos recibe la salida desde la transformación de SQL.

### Archivo SQLErrors

El archivo SQLErrors es un archivo sin formato que contiene cualquier error SQL que se haya producido en la base de datos. El servicio de integración de datos escribe, como mínimo, una fila en el archivo SQLErrors por cada fila de entrada. Si no se produce ningún error SQL, la columna SQLError contiene NULL. Revise el archivo SQLErrors para solucionar los errores.

## Tabla Salario

La tabla Salario es una tabla relacional de la base de datos Nómina. La tabla contiene datos de empleados que mantiene el departamento de nóminas. La transformación de SQL recupera la fecha de alta laboral y el salario del empleado de la tabla Salario.

La siguiente tabla muestra algunas filas de la tabla Salario:

Employee_Num	HireDate	Salary
10	3-May-97	232000
11	11-Sep-01	444000

Employee_Num	HireDate	Salary
12	17-Oct-89	656000
13	13-Ago-07	332100

## Tabla Empleado

El origen es la tabla Empleado de la base de datos Recursos Humanos.

La siguiente tabla muestra algunas filas de la tabla Empleado:

EmpID	LastName	FirstName	DeptId	Teléfono
10	Smith	Martha	FIN	(415) 552-1623
11	Jones	Cynthia	ENG	(415) 552-1744
12	Russell	Cissy	SLS	(415) 552-1656
13	Goyal	Girish	FIN	(415) 552-1656

## Ejemplo de transformación de SQL

La transformación de SQL recupera la fecha de alta laboral y el salario del empleado desde la tabla Salario de la base de datos Nómina. La tabla Salario es una base de datos Oracle.

Utilice los siguientes pasos para configurar la transformación de SQL:

1. Configure las propiedades de la transformación de SQL.
2. Defina los puertos.
3. Cree la consulta SQL.
4. Configure la conexión de base de datos para la transformación de SQL.

### Definir propiedades de transformación de SQL

Configure las propiedades de la transformación de SQL en la vista **Propiedades avanzadas**.

Configure las siguientes propiedades:

#### Tipo de base de datos

El tipo de base de datos es Oracle. En el momento de definir los puertos, puede elegir tipos de datos de puerto aplicables a Oracle.

#### Continuar si hay errores en una fila

Deshabilitar. Detener el proceso si se produce un error SQL en la fila.

#### Incluir estadísticas como salida

Deshabilitar. No cree el puerto de salida NumRowsAffected.

## Definir los puertos

Defina puertos de entrada para cada columna de la tabla de origen de empleados. Seleccione **Copiar a salida** para cambiar los puertos de transferencia de las columnas. Cuando se selecciona **Copiar a salida**, Developer Tool crea el puerto de salida correspondiente para cada puerto que se copia.

Cree los siguientes puertos de transferencia de entrada:

Nombre	Tipo	Tipo nativo	Precisión	Escala	Copiar a salida
EmpID	decimal	número(p,2)	4	0	x
LastName	string	varchar2	30	0	x
FirstName	string	varchar2	20	0	x
DeptID	string	varchar2	4	0	x
Teléfono	string	varchar2	16	0	x

La transformación de expresión tiene los siguientes puertos:

Nombre	Tipo	Tipo nativo	Precisión	Escala
EmpID	decimal	number(p,s)	4	0
LastName	string	varchar2	30	0
FirstName	string	varchar2	20	0
DeptID	string	varchar2	4	0
Teléfono	string	varchar2	16	0
HireDate	date/time	marca de tiempo	29	0
Salary	decimal	number(p,s)	8	2

Developer Tool añade el sufijo "\_output" a cada puerto de salida que crea cuando se selecciona **Copiar a salida**.

Defina manualmente los puertos de salida para las columnas de fecha de alta laboral y de salario. La transformación de SQL devuelve las columnas de fecha de alta laboral y de salario desde la tabla Salario en los puertos.

## Definir la consulta SQL

Cree una consulta SQL para seleccionar la fecha de alta laboral y el salario de cada empleado de la tabla Salario.

Defina la consulta en la vista SQL de la transformación de SQL.

Escriba la consulta siguiente en el editor de SQL:

```
select HIREDATE,SALARY,from Salario where EMPLOYEE_NUM =?EmpID?
```

Hiredate, Salary y Employee\_Num son nombres de columna de la tabla Salario.

?EMPID? es un parámetro que contiene el valor del puerto EmpID.

## Definir la conexión de la base de datos

En la vista **Tiempo de ejecución**, seleccione un objeto de conexión de base de datos para la base de datos a la que se conecta la transformación de SQL. Seleccione un objeto de conexión de base de datos Oracle.

## Salida

Conecte el puerto SQLError y el puerto EmpID\_output al archivo sin formato SQLErrors. El puerto SQLError contiene valores nulos salvo que se produzca un error SQL.

Conecte EmpID y los otros puertos de salida al objeto de datos lógicos.

La transformación SQL devuelve una fila que contiene datos de la tabla Empleado e incluye la fecha de alta y el salario de la tabla Salario.

La siguiente tabla muestra algunas filas del objeto de datos lógicos:

EmpID	LastName	FirstName	DeptId	Teléfono	HireDate	Salary
10	Smith	Martha	FIN	(415) 552-1623	19970303 00:00:00	2320.00
11	Jones	Cynthia	ENG	(415) 552-1744	20010911 00:00:00	4440.00
12	Russell	Cissy	SLS	(415) 552-1656	19891017 00:00:00	6560.00
13	Goyal	Girish	FIN	(415) 552-1660	20070813 00:00:00	3210.00

## CAPÍTULO 29

# Transformación de estandarizador

Este capítulo incluye los siguientes temas:

- [Resumen de la transformación de estandarizador, 327](#)
- [Estrategias de estandarización, 327](#)
- [Propiedades de estandarización, 328](#)
- [Cómo configurar una estrategia de estandarización, 329](#)
- [Propiedades avanzadas de Transformación de estandarizador, 329](#)

## Resumen de la transformación de estandarizador

La transformación de estandarizador es una transformación pasiva que examina cadenas de entrada y crea versiones estandarizadas de las mismas.

La transformación de estandarizador crea columnas que contienen versiones estandarizadas de cadenas de entrada. La transformación puede reemplazar o quitar cadenas en los datos de entrada cuando se crean estas columnas.

Por ejemplo, puede usar la transformación de estandarizador para examinar una columna de datos de direcciones que contenga las cadenas Street, St. y STR. Puede reemplazar todas las instancias de estas cadenas con la cadena St.

Dentro de una transformación de estandarizador puede crear varias estrategias de estandarización. Cada estrategia puede contener varias operaciones de estandarización. La transformación de estandarización proporciona un asistente que puede usarse para crear estrategias.

## Estrategias de estandarización

Utilice estrategias de estandarización para crear columnas con versiones estandarizadas de cadenas de entrada.

Cuando se crea una estrategia de estandarización, se añade una o más operaciones. Cada operación implementa una tarea de estandarización específica.

Se pueden añadir los siguientes tipos de operación a una estrategia de estandarización:

#### Reemplazar coincidencias de tabla de referencia por valores válidos

Reemplaza cadenas que coinciden con valores de tabla de referencia por el valor "Válido" de la tabla de referencia.

#### Reemplazar coincidencias de tabla de referencia por cadenas personalizadas

Reemplaza cadenas que coinciden con valores de tabla de referencia por una cadena de reemplazo definida por el usuario.

#### Quitar coincidencias de tabla de referencia

Quita las cadenas que coinciden con valores de tabla de referencia.

#### Reemplazar cadenas personalizadas

Reemplaza cadenas definidas por el usuario por una cadena de reemplazo definida por el usuario.

#### Quitar cadenas personalizadas

Quita cadenas definidas por el usuario

**Importante:** Se puede modificar el orden de las operaciones. El orden de las operaciones puede cambiar la salida de una estrategia, puesto que cada operación lee los resultados de la operación precedente.

## Propiedades de estandarización

Para configurar las propiedades de estrategias de estandarización y de operaciones, seleccione la vista **Estrategias** en la transformación de estandarizador.

### Propiedades de la estrategia

Las propiedades de la estrategia se aplican a todas las operaciones de la estrategia. Se pueden configurar las siguientes propiedades de estrategia:

#### Quitar varios espacios

Reemplaza varios espacios consecutivos con un espacio.

#### Quitar espacios iniciales y finales

Quita espacios al inicio y final de las cadenas de datos.

#### Delimitadores

Determina los delimitadores que definen los tokens de búsqueda. Por ejemplo, si elige "Punto y coma," la transformación de estandarizador busca en la cadena "naranjas;manzanas" y encuentra las cadenas "naranjas" y "manzanas". La transformación utiliza el delimitador de espacios de forma predeterminada.

### Propiedades de la operación

Puede configurar propiedades para los siguientes tipos de operación de estandarización:

#### Operaciones de tabla de referencia

Las operaciones de tabla de referencia incluyen las siguientes propiedades:

- **Tabla de referencia.** Determina la tabla de referencia que se usa para estandarizar los datos. Haga clic en **Explorar** para seleccionar una tabla de referencia.
- **Distintuir mayúsculas de minúsculas** Determina si el formato de mayúsculas y minúsculas de las cadenas de entrada debe coincidir con el de las entradas de la tabla de referencia.

- **Sustituir por.** Reemplaza las cadenas de entrada que coinciden con entradas de la tabla de referencia con el texto especificado. Solamente se aplica a operaciones de reemplazo.
- **Ámbito.** Especifica la parte de la cadena de entrada que contiene el valor de la tabla de referencia.

#### Operaciones con cadenas personalizadas

Las operaciones con cadenas personalizadas incluyen las siguientes propiedades:

- **Coincidencia de tokens con.** Define las cadenas de búsqueda que encontrar entre los datos de entrada.
- **Sustituir por.** Sustituye las cadenas de entrada que coinciden con las cadenas de búsqueda especificadas. Solamente se aplica a operaciones de reemplazo.
- **Ámbito.** Especifica la parte de la cadena de entrada que buscar.

## Cómo configurar una estrategia de estandarización

Para configurar una estrategia de estandarización, edite los ajustes en la vista **Estrategia** de una transformación de estandarizador.

1. Seleccione la vista **Estrategias**.
2. Haga clic en **Nuevo**.  
Se abre el asistente **Nueva estrategia**.
3. Haga clic en el campo **Entradas** para seleccionar los puertos para la estrategia.
4. Edite las propiedades de la estrategia y haga clic en **Siguiente**.
5. Elija una operación y haga clic en **Siguiente**.
6. Configure las propiedades de la operación.
7. Si lo desea, haga clic en **Siguiente** para añadir más operaciones a la estrategia.
8. Después de añadir todas las operaciones a la estrategia, haga clic en **Finalizar**.
9. Si lo desea, añada más estrategias a la transformación.
10. Si lo desea, cambie el orden en que la transformación procesa las estrategias o las operaciones. Seleccione una estrategia u operación y haga clic en **Mover hacia arriba** o **Mover hacia abajo**.

## Propiedades avanzadas de Transformación de estandarizador

Configure las propiedades para ayudarle a determinar cómo el servicio de integración de datos procesa los datos para la Transformación de estandarizador.

Puede configurar niveles de seguimiento de registros.

Configure la siguiente propiedad en la ficha **Avanzadas**:

#### Nivel de seguimiento

Cantidad de detalles que aparecen en el registro de esta transformación. Puede elegir simplificado, normal, inicialización detallada o datos detallados. El valor predeterminado es Normal.

# Transformación de unión

Este capítulo incluye los siguientes temas:

- [Resumen de la transformación de unión, 330](#)
- [Grupos y puertos, 330](#)
- [Propiedades avanzadas de la transformación de unión, 331](#)
- [Cómo procesar una transformación de unión, 331](#)
- [Cómo crear una transformación de unión, 331](#)

## Resumen de la transformación de unión

Utilice la transformación de incorporación para fusionar datos de varios canales o ramificaciones de canales en una ramificación de canal.

La transformación de unión es una transformación activa con varios grupos de entrada y un grupo de salida. Fusiona orígenes con puertos coincidentes y pasa datos a través de un grupo de salida que tiene la misma estructura de puerto que los grupos de entrada. Utilice una transformación de unión en Developer Tool para fusionar datos de varios orígenes sin quitar las filas duplicadas.

Por ejemplo, puede combinar los datos de cuentas de clientes de los bancos American Bank y California Bank. Puede crear una asignación que contiene una transformación de unión para fusionar datos de los objetos de origen y escribir en un objeto de destino.

## Grupos y puertos

Una transformación de unión tiene varios grupos de entrada y un grupo de salida. Puede crear uno o más grupos de entrada. Developer Tool crea un grupo de salida. No se puede crear, editar o eliminar el grupo de salida. Cada grupo debe tener puertos de coincidencia.

Para crear puertos, puede copiarlos desde una transformación o puede crearlos manualmente. Cuando se crean puertos, Developer Tool crea puertos de entrada en cada grupo de entrada y puertos de salida en el grupo de salida. Developer Tool utiliza los nombres de puerto que se especifican para cada puerto de entrada y de salida. Asimismo, Developer Tool utiliza los mismos metadatos para cada puerto, como el tipo de datos, la precisión y la escala.

Puede conectar los grupos de entrada de distintas ramificaciones en un único canal o desde distintos canales de origen. Cuando se añade una transformación de unión a una asignación, debe comprobar que se conectan los mismos puertos en todos los grupos de entrada. Si se conecta un puerto de un grupo de

entrada pero no se conecta el mismo puerto de otro grupo de entrada, el servicio de integración de datos pasa NULL al puerto no conectado.

## Propiedades avanzadas de la transformación de unión

Configure las propiedades para ayudarle a determinar cómo el servicio de integración de datos muestra los detalles de registro para la transformación de unión.

Puede configurar niveles de seguimiento de registros.

Configure la siguiente propiedad en la ficha **Avanzadas**:

### Nivel de seguimiento

Cantidad de detalles que aparecen en el registro de esta transformación. Puede elegir simplificado, normal, inicialización detallada o datos detallados. El valor predeterminado es Normal.

## Cómo procesar una transformación de unión

Utilice la transformación de unión para fusionar datos de varios canales o ramificaciones de canales en una ramificación de canal. El servicio de integración de datos procesa todos los grupos de entrada en paralelo. Lee simultáneamente los orígenes conectados a la transformación de unión, y pasa bloques de datos a los grupos de entrada de la transformación. La transformación de unión procesa los bloques de datos según el orden en que recibe los bloques desde el servicio de integración de datos. La transformación de unión no bloquea datos de entrada en los grupos de entrada.

## Cómo crear una transformación de unión

Puede crear una transformación de unión reutilizable o no reutilizable.

### Cómo crear una transformación de unión reutilizable

Cree una transformación de unión reutilizable para utilizarla en varias asignaciones o mapplets.

1. Seleccione un proyecto o una carpeta en la vista **Explorador de objetos**.
2. Haga clic en **Archivo > Nueva > Transformación**.  
A continuación, aparece el cuadro de diálogo **Nuevo**.
3. Seleccione la transformación de unión.
4. Haga clic en **Siguiente**.
5. Especifique un nombre para la transformación.
6. Haga clic en **Finalizar**.

La transformación aparece en el editor.

7. Haga clic en el botón **Nuevo** para añadir un puerto a la transformación.
8. Edite el puerto para definir el nombre, el tipo de datos y la precisión.
9. Seleccione la vista **Grupos**.
10. Haga clic en el botón **Nuevo** para añadir un grupo de entrada.
11. Haga clic en la vista **Avanzadas** y edite las propiedades de la transformación.

## Cómo crear una transformación de unión no reutilizable

Cree una transformación de unión no reutilizable como parte de una asignación o de un mapplet.

1. En una asignación o en un mapplet, arrastre una transformación de unión desde la paleta Transformación hasta el editor.  
La transformación aparece en el editor.
2. En la ficha **General**, edite el nombre y la descripción de la transformación.
3. Seleccione todos los puertos de la transformación de nivel superior y arrástrelos a la transformación de unión. Los puertos aparecen como puertos en un grupo de entrada y en un grupo de salida de la transformación de unión.
4. Haga clic en **Nueva**, en la ficha **Grupos** de la vista **Propiedades** para añadir un grupo de entrada.  
Aparece otro grupo de entrada con puertos similares a los del grupo de entrada existente.
5. Seleccione los puertos en el grupo de salida de la transformación de unión y arrástrelos hasta la transformación de nivel inferior en la asignación.

## CAPÍTULO 31

# Transformación de estrategia de actualización

Este capítulo incluye los siguientes temas:

- [Resumen de la transformación de estrategia de actualización, 333](#)
- [Cómo marcar filas dentro de una asignación, 334](#)
- [Cómo especificar opciones de actualización para destinos individuales, 335](#)

## Resumen de la transformación de estrategia de actualización

La transformación de estrategia de actualización es una transformación activa que marca una fila para insertar, actualizar, suprimir o rechazar. Utilice una transformación de estrategia de actualización para controlar las modificaciones que se realicen en las filas existentes de un destino en función de una condición que usted aplique.

Como transformación activa, la transformación de estrategia de actualización puede modificar el número de filas que pasan a través de ella. La transformación de estrategia de actualización comprueba cada fila para ver si cumple una condición en concreto para, a continuación, marcar la fila según proceda. La transformación pasa a la siguiente transformación las filas que marca para insertar, actualizar o suprimir. Puede configurar la transformación para que pase a la siguiente transformación o para que anule las filas marcadas para rechazar.

Por ejemplo, puede usar la transformación de estrategia de actualización para marcar todas las filas de los clientes para su actualización cuando se realice una modificación en la dirección de envío. O bien, puede marcar todas las filas de empleados para rechazarlas en caso de que se trate de personas que ya no trabajan en la empresa.

## Cómo configurar la estrategia de actualización

Para definir una estrategia de actualización, lleve a cabo los siguientes pasos:

1. Para controlar cuántas filas se marcan para insertar, actualizar, suprimir o rechazar en una asignación, añada a la asignación una transformación de estrategia de actualización. Use una transformación de estrategia de actualización para marcar las filas destinadas al mismo fin en diferentes operaciones de bases de datos, o para rechazar filas.

2. Establezca opciones de inserción, actualización y supresión para destinos individuales al configurar la asignación. Destino por destino, puede permitir o prohibir las inserciones y las supresiones de todas las filas marcadas para insertar o suprimir. Puede elegir diferentes formas para gestionar las actualizaciones de todas las filas marcadas para actualizar.

## Cómo marcar filas dentro de una asignación

Añada una transformación de estrategia de actualización a una asignación para marcar filas para su inserción, actualización, supresión o rechazo.

Establezca una expresión de estrategia de actualización para comprobar cada fila y ver si cumple una condición en concreto. A continuación, asigne a cada fila un código numérico para marcar la fila para una operación de base de datos determinada.

En la siguiente tabla se muestran las constantes para cada operación de base de datos y su equivalente numérico:

Operación	Constante	Valor numérico
Insertar	DD_INSERT	0
Actualizar	DD_UPDATE	1
Suprimir	DD_DELETE	2
Rechazar	DD_REJECT	3

El servicio de integración de datos trata cualquier otro valor como una inserción.

## Expresiones de estrategia de actualización

Introduzca una expresión de estrategia de actualización en el editor de expresiones.

La expresión de estrategia de actualización utiliza las funciones IIF o DECODE del lenguaje de transformación para hacer una prueba de cada fila. Por ejemplo, la siguiente instrucción de IIF marca una fila para rechazar si la fecha de la entrada es posterior a la fecha de aplicación. De lo contrario, la instrucción marca la fila para actualizarla:

```
IIF( ( ENTRY_DATE > APPLY_DATE), DD_REJECT, DD_UPDATE)
```

## Propiedades avanzadas de la transformación de estrategia de actualización

Configure las propiedades avanzadas para ayudar a determinar cómo el servicio de integración de datos procesa los datos de la transformación de estrategia de actualización.

Puede definir las siguientes propiedades avanzadas para la transformación de estrategia de actualización en la ficha Avanzadas:

### Reenviar filas rechazadas

Determina si la transformación de estrategia de actualización pasa las filas rechazadas a la siguiente transformación o anula las filas rechazadas. De forma predeterminada, el servicio de integración de datos reenvía las filas rechazadas a la siguiente transformación. El servicio de integración de datos marca las filas para rechazar y las escribe en el archivo de rechazo. Si no selecciona Reenviar filas rechazadas, el servicio de integración de datos anula las filas rechazadas y las escribe en el archivo de registro de asignación.

### Nivel de seguimiento

Cantidad de detalles que aparecen en el registro de esta transformación. Puede elegir simplificado, normal, inicialización detallada o datos detallados. El valor predeterminado es Normal.

## Transformaciones de estrategia de agregación y actualización

Si conecta las transformaciones de estrategia de agregación y actualización como partes de la misma canalización, coloque la transformación de estrategia de agregación antes que la de actualización. Así, el servicio de integración de datos lleva a cabo el cálculo agregado para, después, marcar las filas que contengan los resultados de este cálculo con el fin de insertarlo, actualizarlo, eliminarlo o rechazarlo.

Si coloca la transformación de estrategia de actualización antes que la de agregación, debe tener en cuenta cómo la transformación de agregación trata las filas marcadas para diferentes operaciones. En este orden, el servicio de integración de datos marca las filas para llevar a cabo la inserción, actualización, eliminación o rechazo antes de que se realice el cálculo de agregación. Según cómo marque una fila, así tratará la transformación de agregación los valores que contenga y que se utilicen en el cálculo. Por ejemplo, si marca una fila para eliminarla y, a continuación, la utiliza para calcular la suma, el servicio de integración de datos sustrae el valor en esta fila. Si marca una fila para rechazarla y, a continuación, la utiliza para calcular la suma, el servicio de integración de datos no incluye el valor de esta fila. Si marca una fila para insertarla o actualizarla y, a continuación, la utiliza para calcular la suma, el servicio de integración de datos añade a la suma el valor de esta fila.

## Cómo especificar opciones de actualización para destinos individuales

Después de crear una transformación de estrategia de actualización para marcar cada una de las filas para que realice una operación de base de datos concreta, establezca las opciones de inserción, actualización y supresión para cada destino de la asignación. Puede anular las inserciones o supresiones de las filas marcadas para insertar o eliminar. Puede elegir diferentes formas para gestionar las actualizaciones de todas las filas marcadas para actualizar.

Establezca las opciones de la estrategia de actualización en las propiedades avanzadas de un objeto de datos de destino en una asignación. Puede establecer las siguientes opciones de estrategia de actualización:

### Insertar

Inserta en un destino todas las filas marcadas para insertar. Esta propiedad está habilitada de manera predeterminada.

### Suprimir

Suprime de un destino todas las filas marcadas para suprimir. Esta propiedad está habilitada de manera predeterminada.

### Estrategia de actualización

La estrategia de actualización de las filas existentes. Seleccione una de las siguientes estrategias:

- **Actualizar como actualización.** Actualiza todas las filas marcadas para actualizar. Este es el valor predeterminado.
- **Actualizar como inserción.** Inserta todas las filas marcadas para actualizar.
- **Actualizar o insertar.** Actualiza todas las filas marcadas para actualizar si existen en el destino y, a continuación, inserta las que estén marcadas para insertar.

### Truncar tabla

Trunca el destino antes de cargar los datos. Esta propiedad está deshabilitada de manera predeterminada.

## CAPÍTULO 32

# Transformación de consumidor de servicio web

Este capítulo incluye los siguientes temas:

- [Resumen de la transformación de consumidor de servicio web, 337](#)
- [Selección WSDL, 340](#)
- [Puertos de transformación de consumidor de servicio web, 340](#)
- [Asignación de entrada de transformación de consumidor de servicio web, 342](#)
- [Asignación de salida de transformación de consumidor de servicio web, 344](#)
- [Propiedades avanzadas de la transformación de consumidor de servicio web, 348](#)
- [Optimizaciones de filtro, 352](#)
- [Cómo crear una transformación de consumidor de servicio web, 354](#)
- [Ejemplo de transformación de consumidor de servicio web, 355](#)

## Resumen de la transformación de consumidor de servicio web

La transformación de consumidor de servicio web se conecta a un servicio web como un cliente del servicio web para acceder a los datos o para transformarlos.

Un servicio web utiliza estándares abiertos, como SOAP, WSDL y XML. SOAP es el protocolo de comunicación para los servicios web. La solicitud del cliente del servicio web y la respuesta del servicio web son mensajes SOAP. Un WSDL es un esquema XML que describe los protocolos, los formatos y las firmas de las operaciones del servicio web.

Las operaciones del servicio web incluyen solicitudes de información, de actualización de datos o de ejecución de tareas. Por ejemplo, la transformación de consumidor de servicio web envía una solicitud SOAP para ejecutar una operación del servicio web denominada `getCustomerOrders`. La transformación pasa un ID de cliente en la solicitud. El servicio web recupera la información del cliente y del pedido. El servicio web devuelve la información a la transformación en una respuesta SOAP.

La transformación de consumidor de servicio web se conecta a un servicio web mediante una URL de punto final definida en el WSDL, en una conexión de servicios web o en un puerto de entrada de URL de punto final. La seguridad para los servicios web se habilita en una conexión de servicios web.

## Mensajes SOAP

La transformación de consumidor de servicio web utiliza el Simple Object Access Protocol (SOAP) para intercambiar información con el proveedor de servicios web y para solicitar servicios web. SOAP define el formato de la solicitud del servicio web y de los mensajes de respuesta.

Cuando se transforman datos con una transformación de consumidor de servicio web, la transformación genera una solicitud SOAP y se conecta al servicio web. La transformación se conecta al servicio web mediante una URL de punto final definida en el objeto WSDL, en una conexión de servicios web o en un puerto de entrada de URL de punto final. La solicitud SOAP contiene la información que necesita el servicio web para ejecutar la operación solicitada. La operación del servicio web devuelve datos a la transformación en una respuesta SOAP. La transformación asigna datos desde la respuesta SOAP y devuelve los datos en puertos de salida.

La transformación puede procesar mensajes SOAP con codificación de documento/literal.

Los mensajes SOAP se forman desde el XML. Cuando un mensaje SOAP contiene elementos de ocurrencia múltiple, los grupos de elementos constituyen niveles en la jerarquía XML. Los grupos están relacionados cuando un nivel está anidado dentro de otro.

Un mensaje de solicitud SOAP puede contener datos jerárquicos. Por ejemplo, la transformación de consumidor de servicio web envía una solicitud para añadir pedidos de clientes a una base de datos de ventas. La transformación pasa dos grupos de datos en un mensaje de solicitud SOAP. Un grupo contiene un ID y un nombre de cliente y el otro grupo contiene información de los pedidos. La información de los pedidos se puede generar varias veces.

Un mensaje de respuesta SOAP puede contener datos jerárquicos. Por ejemplo, la transformación de consumidor de servicio web genera una solicitud SOAP para los pedidos de clientes. El servicio web devuelve un encabezado de pedido y elementos de detalle de los pedidos de ocurrencia múltiple en la respuesta SOAP.

## Archivos WSDL

Un archivo WSDL contiene una descripción de los datos que deben pasarse al servicio web para que el emisor y el receptor entiendan los datos que deben intercambiar. Debe importar un archivo WSDL en el repositorio antes de poder crear una transformación de consumidor de servicio web.

El WSDL describe las operaciones que se van a realizar en los datos y en un enlace en un protocolo o transporte, de modo que el consumidor de servicio web pueda enviar el mensaje de solicitud en el formato correcto. El WSDL describe la dirección de red para conectarse al servicio web.

El WSDL incluye información sobre cómo codificar la solicitud SOAP y los mensajes de respuesta. La codificación SOAP determina el formato del cuerpo del mensaje SOAP. Describe el formato de los mensajes de solicitud y de respuesta que utiliza el servicio web para comunicarse con el consumidor de servicio web. Los desarrolladores de servicios web pueden utilizar una variedad de juegos de herramientas para crear servicios web. Los juegos de herramientas son compatibles con diversas maneras de codificar los mensajes SOAP.

La transformación de consumidor de servicio web es compatible con el estilo de codificación SOAP de documento/literal. Puede utilizar WSDL 1.1 con la transformación de consumidor de servicio web. No se pueden usar anexos WDSL como mensajes MIME, DIME y MTOM.

## Operaciones

Un servicio web contiene una operación para cada acción respaldada por el servicio web.

Por ejemplo, un servicio web puede tener una operación llamada `getcustomerid` que recibe un nombre de cliente y que responde con los detalles del cliente. La entrada de operación incluye un elemento para el nombre de cliente. La salida de operación incluye elementos para detalles del cliente según el nombre de cliente.

Cuando se configura una transformación de consumidor de servicio web, se define la manera en la que la transformación asigna datos a la entrada de operación y de la salida de operación. En la transformación se configura la siguiente información:

#### **Asignación de entrada**

Defina cómo deben asignarse los puertos de entrada de transformación a los nodos de entrada de operación del servicio web. La entrada de operación define los elementos en la solicitud SOAP para la operación.

#### **Asignación de salida**

Defina cómo deben asignarse los nodos de salida de operación del servicio web a los puertos de salida de transformación. La salida de operación define los elementos en una respuesta SOAP para la operación.

## Seguridad de servicio web

La seguridad para los servicios web se habilita en una conexión de servicios web. Puede configurar los siguientes tipos de seguridad:

#### **Seguridad de servicio web**

El servicio de integración de datos puede incluir un encabezado de seguridad de servicio web cuando envía una solicitud SOAP al proveedor del servicio web. El encabezado de seguridad del servicio web contiene información de autenticación de modo que el proveedor del servicio web puede autenticar el servicio de integración de datos.

La transformación de consumidor de servicio web proporciona el token de nombre de usuario. El servicio de integración de datos crea un encabezado SOAP de seguridad aparte en la solicitud SOAP y pasa la solicitud al proveedor del servicio web.

Puede utilizar los siguientes tipos de seguridad de servicio web en una conexión de servicios web:

- **PasswordText:** el servicio de integración de datos no cambia la contraseña en el encabezado SOAP de seguridad WS.
- **PasswordDigest:** el servicio de integración de datos combina la contraseña con un valor de seguridad (nonce) y una marca de tiempo. El servicio de integración de datos aplica una dispersión SHA a la contraseña, la codifica como base64 y utiliza la contraseña codificada en el encabezado SOAP.

#### **Seguridad de la capa de transporte**

Seguridad implementada en la parte superior de la capa de transporte (capa TCP) del TCP/IP mediante Secure Sockets Layer (SSL). Los servicios web utilizan Hypertext Transfer Protocol sobre SSL (HTTPS) como dirección web para el transporte de mensajes seguro. Se puede utilizar la siguiente autenticación con la seguridad de la capa de transporte: Autenticación HTTP, autenticación de servidor proxy y certificados SSL.

## Selección WSDL

Antes de crear una transformación de consumidor de servicio web, debe importar un archivo WSDL al repositorio de modelos. El WSDL define la firma de la operación del servicio web que desea ejecutar. Cuando se importa un WSDL, Developer Tool crea un objeto de datos físicos que se puede reutilizar para otras transformaciones.

Un WSDL puede definir múltiples operaciones. Cuando cree una transformación de consumidor de servicio web, seleccione la operación que desea ejecutar. Puede ver las jerarquías de entrada y de salida de la operación en la transformación de consumidor de servicio web. Las jerarquías definen la estructura del mensaje de solicitud SOAP y del mensaje de respuesta SOAP.

## Puertos de transformación de consumidor de servicio web

Cuando visualice los puertos de transformación, muestre los puertos si no necesita ver la jerarquía de la operación. Cuando se muestran los puertos, puede definir grupos y puertos y asignar nodos de la salida de operación a los puertos de salida.

Una transformación de consumidor de servicio web puede poseer varios grupos de entrada y de salida. Cuando cree puertos y grupos y añada los puertos a los grupos. Defina los puertos en una jerarquía de grupo según la estructura de la entrada de operación o la jerarquía de salida de operación. Añada una clave para relacionar un grupo secundario con un grupo principal. Todos los grupos, excepto el grupo situado en el nivel más bajo de la jerarquía, deben tener claves primarias. Todos los grupos de la jerarquía, excepto el grupo raíz, deben tener claves externas.

La transformación posee un grupo de entrada raíz denominado RequestInput. Debe añadir una clave primaria al grupo de entrada raíz. La clave debe contener valores string, bigint o integer.

También puede añadir puertos de transferencia adicionales al grupo de entrada raíz. Los puertos de transferencia pasan datos a través de la transformación sin modificar los datos. El puerto de transferencia puede ocurrir una vez en los datos de entrada. Puede añadir el puerto de transferencia a cualquier grupo de salida. Asocie el puerto de salida al puerto de entrada. El valor de entrada que se transfiere mediante una solicitud SOAP se repite en las filas de salida de la respuesta SOAP.

También puede añadir encabezados HTTP, puertos de cookie, un puerto de URL dinámica y puertos para la autenticación de seguridad de servicio web al grupo de entrada raíz. Los datos en el grupo raíz se generan una vez.

Para asignar un nodo de salida de operación a un puerto de salida, haga clic en el campo de la columna **Ubicación** y expanda la jerarquía del cuadro de diálogo **Seleccionar ubicación**. A continuación, elija un nodo de la jerarquía.

## Puertos de entrada de encabezado HTTP

Un servicio web puede requerir encabezados HTTP adicionales. Puede crear puertos de entrada en el grupo de entrada raíz para pasar la información de los encabezados adicionales al proveedor del servicio web.

Para añadir un encabezado HTTP y un puerto HTTP, seleccione el grupo de entrada raíz y haga clic en la flecha situada junto al botón **Nuevo**. A continuación, haga clic en **Encabezado HTTP**. Especifique un nombre de encabezado y un nombre de puerto.

Se pueden crear varios encabezados HTTP.

## Otros puertos de entrada

Puede añadir puertos de entrada predefinidos a la transformación de consumidor de servicio web.

Puede añadir los siguientes puertos de entrada predefinidos:

### Puerto de cookie

Puede configurar la transformación de consumidor de servicio web para que utilice autenticación por cookies. El servidor remoto del servicio web hace un seguimiento de los usuarios del servicio web mediante cookies. Es posible aumentar el rendimiento cuando una asignación llama un servicio web varias veces.

Cuando se proyecta el puerto de cookie a un mensaje de solicitud de servicio web, el proveedor del servicio web devuelve un valor de cookie en el mensaje de respuesta. Puede pasar el valor de cookie a otra transformación situada en un nivel inferior en la asignación o puede guardar el valor de la cookie en un archivo. Cuando se guarda el valor de la cookie en un archivo, puede configurar la cookie como entrada de la transformación de consumidor de servicio web.

Puede proyectar el puerto de salida de la cookie a cualquiera de los grupos de salida de la transformación de consumidor de servicio web.

### Puerto de URL de punto final

La transformación de consumidor de servicio web se conecta con un servicio web mediante una URL de punto final. Puede definir la URL de punto final en el archivo WSDL, en una conexión de servicios web o en un puerto de entrada de URL de punto final. Cuando la transformación recibe la URL dinámicamente en un puerto, el servicio de integración de datos reemplaza la URL definida en el archivo WSDL o en la conexión de servicios web.

La transformación de consumidor de servicio web puede tener un valor de puerto URL para cada solicitud de servicio web. Añada un puerto de URL de punto final al grupo de entrada raíz.

### Puertos de seguridad WS

Se habilita la seguridad de servicio web en una conexión de servicios web. Cuando se habilita la seguridad de servicio web, debe definir el nombre de usuario y la contraseña en una conexión de servicios web o en puertos de entrada de seguridad WS.

Cuando se añaden puertos de seguridad WS, se pasan el nombre de usuario y la contraseña a través de puertos de entrada en la transformación. Cuando la transformación recibe el nombre de usuario y la contraseña dinámicamente en los puertos, el servicio de integración de datos reemplaza los valores definidos en la conexión de servicios web.

**Nota:** una conexión de servicios web posee un nombre de usuario y una contraseña para la autenticación de HTTP y de seguridad WS.

Para añadir puertos de entrada predefinidos, haga clic en el grupo de entrada raíz en el área **Puertos**. Haga clic en la flecha situada junto al botón **Nuevo** y luego haga clic en **Otros puertos**. Elija los puertos que desea añadir.

# Asignación de entrada de transformación de consumidor de servicio web

Cuando visualice los puertos de transformación, muestre la asignación de entrada para ver la jerarquía de entrada de operación. Cuando se muestra la asignación de entrada, puede definir grupos y puertos de entrada y asignar puertos de entrada a nodos de entrada de operación.

La asignación de entrada incluye las siguientes áreas:

## Puertos

Cree los grupos de entrada de transformación y los puertos de entrada en el área **Puertos**.

## Entrada de operación

El área **Entrada de operación** muestra los nodos del mensaje de solicitud SOAP que la transformación de consumidor de servicio web envía al servicio web. El objetos de datos WSDL que se utiliza para crear la transformación define la jerarquía de entrada de a operación.

Después de crear puertos de entrada, asigne los puertos de entrada del área **Puertos** a los nodos del área **Entrada de operación**. Cuando se asigna un puerto de entrada a un nodo de la entrada de operación, la ubicación del nodo aparece en la columna **Ubicación** del área **Entrada de operación**.

Developer Tool asigna nodos del primer nivel de la entrada de operación a puertos de entrada cuando se opta por asignar el primer nivel de la jerarquía de entrada. Asimismo, Developer Tool crea los puertos que necesita para realizar la asignación. Si el primer nivel de la jerarquía contiene un nodo principal de ocurrencia múltiple con uno o varios nodos secundarios de ocurrencia múltiple, Developer Tool no asigna el primer nivel de la jerarquía.

Puede asignar datos XML de una cadena o un puerto de entrada de texto al mensaje de solicitud SOAP completo. Cuando se asignan datos XML a la solicitud SOAP completa, no se pueden asignar puertos a nodos de la entrada de operación.

Puede optar por ver las líneas que conectan los puertos de entrada a los nodos de la entrada de operación.

## TEMAS RELACIONADOS

- [“Resumen de la generación de mensajes SOAP del servicio web” en la página 367](#)

## Reglas y directrices para asignar puertos de entrada a nodos

Revise las siguientes reglas cuando asigne puertos de entrada a nodos en la jerarquía de entrada de operación:

- Puede asignar un puerto de entrada a un nodo de la jerarquía. Puede asignar el mismo puerto a un número indeterminado de claves de la jerarquía.
- El puerto de entrada y el nodo deben tener tipos de datos compatibles.
- Puede asignar puertos de un grupo de entrada a varios niveles de jerarquía en la entrada de operación.
- Debe asignar puertos de entrada a las claves en la entrada de operación. Cualquier puerto que se asigne a una clave debe ser del tipo de datos string, integer o bigint. Asigne los datos a las claves en todos los niveles de la entrada de operación situados encima del nivel de jerarquía que esté incluyendo en el mensaje SOAP. Incluya las claves externas para todos los niveles situados por encima e incluyendo el nivel que esté asignando.

**Nota:** No es necesario que asigne puertos de entrada a claves si únicamente está asignando el nivel más bajo de la jerarquía de entrada de operación.

- Puede asignar varios puertos de entrada con valores string, integer o bigint a una clave en el área **Entrada de operación** para crear una clave compuesta. Cuando se hace clic en el campo **Ubicación** para una clave compuesta, puede volver a ordenar los puertos de entrada o eliminar uno de los puertos.

## Personalizar opciones de vista

Puede cambiar la jerarquía de entrada de operación para que las claves se muestren en el área **Entrada de operación**. También puede mostrar construcciones de agrupación que definen cómo ordenar nodos.

Haga clic en el botón **Personalizar vista** en el área **Entrada de operación**. Habilite una de las siguientes opciones:

### Secuencia, Selección y Todo

Muestra una línea que indica si una definición de elemento es secuencia, selección o todo.

Los nodos en un grupo todo deben incluirse en el mensaje SOAP.

Los nodos en un grupo secuencia deben estar en el orden especificado en el WSDL.

En el mensaje SOAP debe aparecer, como mínimo, un nodo en un grupo selección.

### Claves

Visualice las claves en el área **Entrada de operación**. El área **Entrada de operación** incluye claves para cada grupo. Puede añadir una clave a un puerto de entrada en el área **Puertos**.

## Cómo asignar puertos de entrada a la entrada de operación

Cuando se muestra la asignación de entrada de transformación, puede definir grupos y puertos de entrada y asignar puertos de entrada a nodos de entrada de operación.

1. Abra una transformación de consumidor de servicio web.
2. Para mostrar la asignación de entrada de la transformación, utilice uno de los siguientes métodos:
  - Para una transformación reutilizable, haga clic en la vista **Resumen**. Elija mostrar la asignación de entrada.
  - Para una transformación no reutilizable, haga clic en la ficha **Puertos** en la vista **Propiedades**. Elija mostrar la asignación de entrada.
3. Defina una clave primaria para el grupo de entrada raíz.
4. Para añadir un grupo o un puerto de entrada al área **Puertos**, utilice uno de los siguientes métodos:

Opción	Descripción
Arrastrar un nodo	Arrastre un nodo de grupo o un nodo secundario en el área <b>Entrada de operación</b> hasta una columna vacía del área <b>Puertos</b> . Si el nodo es un nodo de grupo, Developer Tool agrega un grupo sin puertos.
Añadir un grupo o puerto manualmente	Para añadir un grupo, haga clic en la flecha situada junto al botón <b>Nuevo</b> y luego haga clic en <b>Grupo</b> . Para añadir un puerto, haga clic en la flecha situada junto al botón <b>Nuevo</b> y luego haga clic en <b>Campo</b> .
Arrastrar un puerto desde otra transformación	En el editor, arrastre un puerto desde otra transformación a la transformación de consumidor de servicio web.

Opción	Descripción
Copiar un puerto	Seleccione puertos de otra transformación y cópielos al área <b>Puertos</b> . Para copiar puertos, puede utilizar métodos abreviados de teclado o puede utilizar los botones copiar y pegar de Developer Tool.
Seleccione <b>Asignar primer nivel de la jerarquía</b> .	Seleccione <b>Asignar primer nivel de la jerarquía</b> . Developer Tool asigna nodos que se encuentran en el primer nivel de la entrada de operación a puertos y grupos de entrada. Developer Tool también crea los puertos y grupos de entrada para realizar la asignación.

- Si crea manualmente un puerto o lo copia desde otra transformación, haga clic en la columna **Ubicación** del área **Entrada de operación** y elija un puerto de la lista.
- Para asignar puertos de entrada como una clave compuesta, utilice uno de los siguientes métodos:

Opción	Descripción
Arrastrar puertos de entrada	Seleccione dos o más puertos de entrada y arrástrelos hasta una clave en la jerarquía de entrada de operación.
Seleccione puertos de entrada desde el cuadro de diálogo <b>Seleccionar ubicación</b>	Haga clic en la columna <b>Ubicación</b> de una clave en la jerarquía de entrada de operación y, a continuación, seleccione los puertos de entrada.

- Para borrar las ubicaciones de los nodos, utilice uno de los siguientes métodos:

Opción	Descripción
Haga clic en <b>Borrar</b> .	Seleccione uno o más nodos del área <b>Entrada de operación</b> y haga clic en <b>Borrar</b> .
Eliminar las líneas que conectan puertos con nodos	Seleccione una o más líneas que conectan los puertos de entrada con los nodos en la entrada de operación y pulse <b>Eliminar</b> .

- En el caso de que el objeto de datos WSDL asociado incluya elementos anyType, elementos, atributos anyAttribute, elementos de tipo derivado o grupos de sustitución, elija objetos que se encuentran en el área **Entrada de operación**. En la columna **Tipo** de un nodo, haga clic en **Elegir** y luego elija uno o más tipos, elementos o atributos de la lista.
- Para asignar datos XML de una cadena o un puerto de entrada de texto a la solicitud SOAP completa, haga clic con el botón derecho en el puerto y seleccione **Asignar como XML**.

## Asignación de salida de transformación de consumidor de servicio web

Cuando visualice los puertos de transformación, muestre la asignación de salida para ver la jerarquía de salida de operación. Cuando se muestra la asignación de salida, puede definir grupos y puertos de salida y asignar nodos de salida de operación a puertos de salida.

La asignación de salida incluye las siguientes áreas:

### Salida de operación

El área **Salida de operación** muestra los nodos del mensaje de respuesta SOAP que el servicio web devuelve a la transformación de consumidor de servicio web. El objeto de datos WSDL que se utiliza para crear la transformación define la jerarquía de salida de operación.

### Puertos

Cree los puertos y los grupos de salida de transformación en el área **Puertos**.

Después de crear los puertos de salida, asigne los nodos del área **Salida de operación** a los puertos del área **Puertos**. Cuando se asigna un nodo de la salida de operación a un puerto de salida, la ubicación del nodo aparece en la columna **Ubicación** en el área **Puertos**.

Developer Tool asigna nodos del primer nivel de la salida de operación a puertos de salida cuando se opta por asignar el primer nivel de la jerarquía de salida. Asimismo, Developer Tool crea los puertos que necesita para realizar la asignación. Si el primer nivel de la jerarquía contiene un nodo principal de ocurrencia múltiple con uno o varios nodos secundarios de ocurrencia múltiple, Developer Tool no asigna el primer nivel de la jerarquía.

Puede elegir que se muestren los puertos de salida de una jerarquía. Cada grupo secundario aparece debajo del grupo principal. También puede optar por ver las líneas que conectan los nodos de la salida de operación a los puertos de salida.

Si el objeto de datos WSDL asociado se elimina del repositorio, Developer Tool guarda la ubicación de los nodos de operación en la asignación de salida. Cuando se muestra la asignación de salida, el área **Puertos** sigue mostrando la ubicación de los nodos de operación en la columna **Ubicación** para los puertos de salida. Si asocia otro WSDL con la transformación, Developer Tool comprueba si cada ubicación es válida. Developer Tool borra la ubicación de los nodos de operación del área **Puertos** de la asignación de salida si la ubicación ha dejado de ser válida.

### TEMAS RELACIONADOS

- [“Resumen del análisis de mensajes SOAP del servicio web” en la página 359](#)

## Reglas y directrices para asignar nodos a puertos de salida

Revise las siguientes reglas cuando asigne nodos de la jerarquía de salida de operación a puertos de salida:

- El nodo de salida de operación y el puerto de salida deben tener tipos de datos compatibles.
- No se puede asignar un nodo a más de un puerto de salida en un grupo.
- Cada puerto de salida debe tener una ubicación válida, a excepción de que el puerto sea un puerto de transferencia.
- Si arrastra un nodo secundario de ocurrencia múltiple a un puerto de salida vacío, debe relacionar el grupo con otros grupos de salida. Cuando se selecciona un grupo, Developer Tool crea claves para relacionar los grupos.
- Cuando se arrastra un elemento de ocurrencia múltiple a un grupo que contiene el elemento principal, puede configurar el número de ocurrencias de elemento secundario que se van a incluir. También puede reemplazar el grupo principal con el grupo secundario de ocurrencia múltiple en la salida de transformación.

## Cómo asignar el mensaje SOAP como XML

Puede asignar el mensaje SOAP completo como XML en lugar de devolver los datos a puertos de salida individuales.

Cuando se asigna el mensaje SOAP como XML, el servicio de integración de datos devuelve el mensaje SOAP completo en un puerto. No crea puertos de salida.

Para asignar el mensaje completo, haga clic con el botón derecho en el grupo raíz en el área **Salida de operación**. Seleccione **Asignar como XML**.

Developer Tool crea un puerto de salida de cadenas. La precisión es 65535 bytes.

## Personalizar opciones de vista

Puede cambiar la jerarquía de salida de operación para que los puertos de cookies, los puertos de transferencia y las claves se muestren en el área **Salida de operación**. También puede mostrar construcciones de agrupación que definen cómo ordenar nodos.

Haga clic en el botón **Personalizar vista** en el área **Salida de operación**. Habilite una de las siguientes opciones:

### Secuencia, Selección y Todo

Muestra una línea que indica si una definición de elemento es secuencia, selección o todo.

Los nodos en un grupo todo deben incluirse en el mensaje SOAP.

Los nodos en un grupo secuencia deben estar en el orden especificado en el WSDL.

En el mensaje SOAP debe aparecer, como mínimo, un nodo en un grupo selección.

### Claves

Visualice las claves en el área **Salida de operación**. El área **Salida de operación** incluye claves para cada grupo. Puede añadir una clave a un puerto de salida en el área **Puertos**.

### Puertos de transferencia

El área **Salida de operación** muestra los puertos de transferencia. Los puertos de transferencia son puertos que transfieren datos a través de la transformación sin modificar los datos. Puede proyectar puertos de transferencia desde la salida de operación a cualquiera de los grupos de salida de la transformación de consumidor de servicio web. Un puerto de transferencia recibe los datos sólo una vez, por lo que el puerto se encuentra en el nivel raíz de los mensajes SOAP.

### Puertos de cookies

Muestra el puerto de cookies. Cuando se configura la autenticación por cookies, el servidor remoto del servicio web hace un seguimiento de los usuarios del consumidor de servicio web mediante cookies. Cuando se proyecta una cookie de servicio web en el mensaje de solicitud, el servicio web devuelve una cookie en el mensaje de respuesta. Puede proyectar la cookie desde la salida de operación a cualquiera de los grupos de salida de la transformación de consumidor de servicio web.

## Cómo asignar la salida de operación a puertos de salida

Cuando se muestra la asignación de salida de transformación, puede definir grupos y puertos de salida y asignar nodos de salida de operación a puertos de salida.

1. Abra una transformación de consumidor de servicio web.
2. Para mostrar la asignación de salida de la transformación, utilice uno de los siguientes métodos:

- Para una transformación reutilizable, haga clic en la vista **Resumen**. Elija mostrar la asignación de salida.
- Para una transformación no reutilizable, haga clic en la ficha **Puertos** en la vista **Propiedades**. Elija mostrar la asignación de salida.

3. Para añadir un grupo o un puerto de salida al área **Puertos**, utilice uno de los siguientes métodos:

Opción	Descripción
Arrastrar un nodo	Arrastre un nodo de grupo o un nodo secundario en el área <b>Salida de operación</b> hasta una columna vacía del área <b>Puertos</b> . Si el nodo es un nodo de grupo, Developer Tool agrega un grupo sin puertos.
Añadir un grupo o puerto manualmente	Para añadir un grupo, haga clic en la flecha situada junto al botón <b>Nuevo</b> y luego haga clic en <b>Grupo</b> . Para añadir un puerto, haga clic en la flecha situada junto al botón <b>Nuevo</b> y luego haga clic en <b>Campo</b> .
Arrastrar un puerto desde otra transformación	En el editor, arrastre un puerto desde otra transformación a la transformación de consumidor de servicio web.
Copiar un puerto	Seleccione puertos de otra transformación y cópielos al área <b>Puertos</b> . Para copiar puertos, puede utilizar métodos abreviados de teclado o puede utilizar los botones copiar y pegar de Developer Tool.
Seleccione <b>Asignar primer nivel de la jerarquía</b> .	Seleccione <b>Asignar primer nivel de la jerarquía</b> . Developer Tool asigna nodos que se encuentran en el primer nivel de la salida de operación a puertos y grupos de salida. Developer Tool también crea los grupos y puertos de salida para realizar la asignación.

4. Si crea manualmente un puerto o lo copia desde otra transformación, haga clic en la columna **Ubicación** del área **Puertos** y elija un nodo de la lista.

5. Para borrar las ubicaciones de los puertos, utilice uno de los siguientes métodos:

Opción	Descripción
Haga clic en <b>Borrar</b> .	Seleccione uno o más puertos del área <b>Puertos</b> y haga clic en <b>Borrar</b> .
Eliminar las líneas que conectan los nodos con los puertos	Seleccione una o más líneas que conectan los nodos en la salida de operación con los puertos de salida y pulse <b>Eliminar</b> .

6. En el caso de que el objeto de datos WSDL asociado incluya elementos anyType, elementos, atributos anyAttribute, elementos de tipo derivado o grupos de sustitución, elija objetos que se encuentran en el área **Salida de operación**. En la columna **Tipo** de un nodo, haga clic en **Elegir** y luego elija uno o más tipos, elementos o atributos de la lista.

7. Para asignar el mensaje de respuesta SOAP completo como XML, haga clic con el botón derecho en el grupo raíz en el área **Salida de operación** y seleccione **Asignar como XML**.

# Propiedades avanzadas de la transformación de consumidor de servicio web

Las propiedades avanzadas de la transformación de consumidor de servicio web incluyen el nivel de seguimiento, los puertos de fallos genéricos, la conexión de servicios web y los mensajes de solicitud de servicio web simultáneos.

Puede definir las siguientes propiedades avanzadas para la transformación de consumidor de servicio web en la ficha Avanzadas:

## Nivel de seguimiento

Cantidad de detalles que aparecen en el registro de esta transformación. Puede elegir simplificado, normal, inicialización detallada o datos detallados. El valor predeterminado es Normal.

## Acción de SOAP

Reemplaza el valor de la acción de SOAP definida en el WSDL por un valor constante para la transformación de consumidor de servicio web.

## Habilitar control de fallos SOAP genéricos

Devuelve mensajes de fallo que no se han definido en el WSDL. Crea puertos de salida en un grupo de salida GenericFault para controlar los códigos y los mensajes de fallos.

La siguiente tabla describe los puertos de salida de fallo para SOAP 1.1 y SOAP 1.2:

Puerto de salida de fallo para SOAP 1.1	Puerto de salida de fallo para SOAP 1.2	Descripción
Código de fallo	Código*	Devuelve un código de identificación de fallo.
Cadena de fallo	Motivo*	Devuelve una explicación del error en un mensaje.
Detalles del fallo	Detalle	Devuelve información personalizada que el proveedor del servicio web pasa a la transformación de consumidor de servicio web en un mensaje de fallo genérico.
Actor del fallo	Función	Devuelve información sobre el objeto que ha generado el fallo.
-	Nodo	Devuelve el URI del nodo de SOAP que ha generado el fallo.
* Los puertos de salida de código y motivo son jerárquicos.		

**Nota:** Puede expandir el puerto de salida de fallo de código para extraer el puerto de salida de fallo del subcódigo a un nivel superior.

## Habilitar el control de errores HTTP

Devuelve cualquier error HTTP del servicio web. Crea un puerto de salida de errores HTTP en el grupo de salida GenericFault.

### Tratar fallo como error

Agrega mensajes de fallos al registro de asignación. Cuando se produce un fallo, el servicio de integración de datos incrementa el recuento de errores para la asignación. Deshabilite esta propiedad para permitir la optimización de primera selección y de inserción. Esta propiedad está habilitada de manera predeterminada.

### Conexión

Identifica el objeto de la conexión de servicios web para conectarse con el servicio web. Cree la conexión de servicios web en Developer Tool. Edite la conexión de servicios web en Developer Tool o en Administrator Tool. Cuando configure una conexión de servicios web, configure la URL de punto final, el tipo de seguridad que el servicio web requiera y un tiempo de espera para la conexión.

La transformación de consumidor de servicio web se conecta con un servicio web mediante una URL de punto final. Puede definir la URL de punto final en el archivo WSDL, en una conexión de servicios web o en un puerto de entrada de URL de punto final.

Utilice las siguientes directrices para saber cuándo debe configurar una conexión de servicios web:

- Configure una conexión cuando desee utilizar una URL de punto final que sea diferente de la URL del archivo WSDL y cuando no esté utilizando un puerto de entrada de URL de punto final.
- Configure una conexión cuando el servicio web al que se conecte requiera seguridad de servicio web, autenticación HTTP o certificados SSL.
- Configure una conexión cuando desee cambiar el tiempo de espera predeterminado de la conexión.

**Nota:** Puede asociar un objeto de datos WSDL del repositorio a una conexión de servicios web. La conexión asociada pasa a ser la conexión predeterminada para cada transformación de consumidor de servicio web que cree a partir de ese WSDL.

### Habilitar compresión

Habilita la codificación de solicitudes SOAP con el método de compresión GZIP y habilita la decodificación de respuestas con GZIP o desinflado.

### Validación de esquema XML

Valida el mensaje de respuesta SOAP en tiempo de ejecución. Seleccione **Error en XML no válido** o **Sin validación**.

### Entrada ordenada

Habilita el servicio de integración de datos para generar una salida sin procesar todos los datos de entrada. Habilite la entrada ordenada cuando los datos de entrada se ordenen según las claves en la jerarquía de entradas de operación.

### Optimización de inserción

Habilita la optimización de inserción. Haga clic en el botón **Abrir** en la propiedad **Optimización de inserción** para seleccionar los puertos de filtro que reciben valores de filtro. Para cada puerto de filtro, seleccione el puerto de salida que contiene la columna filtrada en la respuesta de servicio web.

### Tiene efectos secundarios

Casilla de verificación que indica que el servicio web realiza cualquier función además de devolver filas. La transformación de consumidor de servicio web tiene un efecto secundario si el servicio web, además de devolver filas, modifica un objeto o interactúa con otros objetos o funciones. El servicio web puede modificar una base de datos, añadir a un total, elevar una excepción, escribir un correo electrónico o llamar a otros servicios web con efectos secundarios. Deshabilite la propiedad **Tiene efectos secundarios** para permitir la optimización de inserción o la optimización de primera selección. Esta propiedad está habilitada de manera predeterminada.

### Habilitar simultaneidad

Habilita la transformación del consumidor de servicio web para crear varias conexiones simultáneas a un servicio web de modo que pueda enviar varias solicitudes de servicio web en paralelo. Cuando habilita la transformación de consumidor del servicio web para crear varias conexiones simultáneas en el servicio web, puede establecer el límite del consumo total de memoria y el número de límites de conexiones simultáneas.

En la siguiente tabla se describen las opciones:

Opciones	Descripción
Habilitar simultaneidad	Crea varias conexiones simultáneas en un servicio web.
Límite conexión simultánea	El número de conexiones de servicio web simultáneas. El valor predeterminado es 20.
Límite total de memoria simultánea (en MB)	El límite total de asignación de memoria para todas las conexiones simultáneas. El valor predeterminado es 100 MB.

## Control de errores de servicio web

Puede configurar la transformación de consumidor de servicio web para que pase fallos SOAP y errores HTTP hacia abajo en una asignación. Puede incrementar el recuento de errores cuando se produce un fallo. Configure el control de errores de servicio web en las propiedades avanzadas de la transformación.

Un servicio web devuelve un mensaje de respuesta o devuelve un fallo. Un fallo es un error. El servicio web puede generar fallos diferentes según los errores que se producen.

La transformación de consumidor de servicio web puede devolver los siguientes tipos de fallos:

### Fallos SOAP

Errores SOAP que define el WSDL. Configure los puertos de error de salida que devuelven los fallos en el mensaje de respuesta del servicio web. Para un enlace SOAP 1.1, el servicio de integración de datos devuelve el mensaje de fallo, el código de fallo, la cadena de fallo y los elementos del actor del fallo para el fallo. Para un enlace SOAP 1.2, el servicio de integración de datos devuelve el mensaje de fallo, el código, el motivo, el nodo y los elementos de función para el fallo.

### Fallos SOAP genéricos

El servicio web genera fallos SOAP en tiempo de ejecución. Los elementos de fallo son diferentes para un enlace SOAP 1.1 y un enlace SOAP 1.2. El WSDL no define fallos SOAP genéricos. Los fallos SOAP genéricos incluyen fallos de autenticación y errores de solicitud SOAP.

### Errores HTTP

Developer Tool agrega el puerto de salida del fallo HTTP cuando se habilita el control de errores HTTP en la transformación. El servicio de integración de datos devuelve errores HTTP del servicio web en un puerto de cadena individual. Un error HTTP incluye un código de error y un mensaje.

Si la respuesta SOAP del servicio web tiene datos XML no válidos, la transformación de consumidor de servicio web devuelve un error.

Puede especificar si se deben tratar los fallos SOAP como errores. Cuando se habilita Tratar fallo como error y se produce un fallo SOAP, el servicio de integración de datos incrementa el recuento de errores para la asignación. El fallo aparece en el registro de mensajes.

## Compresión de mensajes

Cuando se habilita la compresión de mensajes SOAP, la transformación de consumidor de servicio web comprime los mensajes de solicitud de servicio web y recibe mensajes de respuesta de servicio web comprimidos.

La transformación de consumidor de servicio web codifica la solicitud SOAP mediante compresión GZip. La transformación acepta un mensaje de respuesta codificado con GZip o la compresión por desinflado.

Cuando el servicio de integración de datos recibe la respuesta del servicio web, comprueba el encabezado HTTP de codificación de contenido en el mensaje SOAP y descodifica el mensaje.

El valor predeterminado es no realizar codificación de compresión. El servicio web no comprime la respuesta SOAP.

La siguiente tabla muestra los encabezados en la solicitud los mensajes de respuesta cuando la compresión está activada o desactivada:

Compresión	Encabezado
Activada	Encabezado de codificación de contenido: GZip Encabezado de codificación de aceptación: GZip, desinflado
Desactivada	Encabezado de codificación de contenido vacío Encabezado de codificación de aceptación vacío

En ocasiones, un servicio web codifica un mensaje de respuesta con una compresión predeterminada. La transformación de consumidor de servicio web descodifica el mensaje si está codificado mediante GZip o desinflado. La transformación de consumidor de servicio web registra un mensaje en el registro de asignación en el caso de que el servicio web codifique el mensaje de respuesta inesperadamente.

Habilite la compresión en las propiedades avanzadas de la transformación.

## Simultaneidad

Puede habilitar la transformación de consumidor del servicio web para crear varias conexiones simultáneas en un servicio web de modo que pueda enviar varias solicitudes de servicio web en paralelo.

Por ejemplo, al comprobar la información bancaria, puede configurar la transformación de consumidor de servicio web en simultaneidad para que se envíen varias filas en paralelo. Si hay 20 filas de entrada, puede enviar 20 solicitudes simultáneas para un procesamiento más rápido.

Cuando habilita la simultaneidad en la transformación de consumidor de servicio web, puede configurar el límite total del consumo de memoria.

Cuando habilita la simultaneidad en la transformación de consumidor de servicio web, puede configurar el número de conexiones de servicio web simultáneas.

### Reglas y directrices para la simultaneidad

Utilice las siguientes reglas y directrices cuando utilice la simultaneidad:

- La simultaneidad es compatible con las filas de entrada ordenada como varias conexiones simultáneas para un servicio web. Las filas de salida ordenadas no son compatibles.
- Utilice la simultaneidad del si el conjunto de datos tiene más de 100 filas.

- Se recomienda no aumentar el número de conexiones de servicio web simultáneas. El número de conexiones de servicio web simultáneas está vinculado al número de conexiones utilizadas por el sistema operativo. Aumentar el número de conexiones es caro.
- Utilice sistemas que tengan procesadores de varios núcleos con un mínimo de 100 MB de memoria RAM para un rendimiento óptimo al utilizar la función de simultaneidad.
- El límite de memoria de simultaneidad representa la memoria consumida por los flujos de trabajo simultáneos durante la invocación de servicios web.
- Cuando habilita la simultaneidad en la transformación de consumidor de servicio web, puede configurar el límite de consumo de memoria. Asegúrese de que el consumo de memoria no sea mayor que la memoria RAM física del servidor.

## Mejores prácticas para la simultaneidad

Para un rendimiento óptimo al utilizar la simultaneidad, siga estas mejores prácticas:

- Evitar cambiar los valores predeterminados del límite de memoria simultánea total y el límite de conexión simultánea.
- Evitar utilizar la simultaneidad para conjuntos de datos con menos de 100 filas.
- Evitar puertos de transferencia en la asignación al utilizar la simultaneidad.

# Optimizaciones de filtro

Las optimizaciones de filtro aumentan el rendimiento reduciendo el número de filas que pasan por la asignación. El servicio de integración de datos puede aplicar la optimización de primera selección o la de inserción.

Cuando el servicio de integración de datos aplica un método de optimización del filtro, mueve un filtro tan cerca del origen como sea posible en una asignación. Si el servicio de integración de datos no puede mover un filtro antes de una transformación en una asignación, es posible que pueda insertar la lógica de filtro en una transformación.

## Habilitar la optimización de primera selección con la transformación del consumidor de servicio web

Habilite la optimización de primera selección para la transformación de consumidor de servicio web si la transformación no tiene efectos secundarios y no trata los fallos como errores.

1. Abra la vista **Propiedades avanzadas** de la transformación del consumidor de servicio web.
2. Desactive **Tratar fallo como error**.
3. Desactive **Tiene efectos secundarios**.

## Optimización de inserción con la transformación del consumidor de servicio web

Con la optimización de inserción, la transformación del consumidor de servicio web recibe el valor de filtro en un puerto de filtro. El puerto de filtro es un puerto de entrada inconexo que identifica como un puerto de filtro al configurar la optimización de inserción. El puerto de filtro tiene un valor predeterminado que garantiza que

el servicio web devuelve todas las filas si la consulta del usuario final no contiene ningún filtro. El puerto de filtro no es un puerto de transferencia.

**Nota:** El campo de filtro debe formar parte del grupo raíz en la solicitud de servicio web.

El campo de filtro no puede ser un puerto de transferencia. Cuando configura un puerto de filtro, el valor predeterminado del puerto cambia al valor de la condición de filtro, por lo que el valor del puerto de salida de transferencia cambia. Un filtro basado en el puerto de transferencia de salida devuelve resultados inesperados.

Puede insertar varias expresiones de filtro para la transformación de consumidor de servicio web. Cada condición de filtro debe tener el siguiente formato:

<Field> = <Constant>

Las condiciones de filtro deben estar unidas por AND. No puede combinar las condiciones con un OR.

## Optimización de inserción con el ejemplo de transformación del consumidor de servicio web

Un servicio de datos SQL devuelve pedidos de todos los clientes o devuelve pedidos de un cliente específico según la consulta SQL que reciba del usuario.

El servicio de datos contiene un objeto de datos lógicos con los siguientes componentes:

### Tabla de clientes

Una tabla de base de datos Oracle que contiene la información de los clientes.

### Transformación de consumidor de servicio web

Una transformación que llama a un servicio web para recuperar los últimos pedidos de los clientes. La transformación del consumidor de servicio web tiene puertos de entrada para customerID y orderNum. La transformación tiene puertos de transferencia que contienen datos de los clientes que recibe desde la tabla de clientes. El puerto orderNum es el puerto de filtro y no está conectado. orderNum tiene el valor predeterminado "\*\*\*". Cuando el servicio web recibe este valor en la solicitud de servicio web, devuelve todos los pedidos.

### Tabla virtual de pedidos

Una tabla virtual que recibe los datos de los clientes y los pedidos desde el servicio web. El usuario final consulta esta tabla. Los pedidos contienen una columna de cliente, una columna orderID y datos de clientes y pedidos.

El usuario final pasa la siguiente consulta SQL al servicio de datos SQL:

```
SELECT * de OrdersID donde cliente = 23 y orderID = 56
```

El servicio de integración de datos divide la consulta para optimizar la asignación. El servicio de integración de datos utiliza la optimización de primera selección y mueve la lógica de filtro, cliente = 23, a la lectura de tabla de clientes. El servicio de integración de datos utiliza la optimización de inserción e inserta la lógica de filtro, orderID = 56, en el puerto del filtro de la transformación del consumidor de servicio web. La transformación del consumidor de servicio web recupera ordersID 56 para clientes 23.

## Habilitar la optimización de inserción con la transformación del consumidor de servicio web

Habilite la optimización de inserción para la transformación de consumidor de servicio web si la transformación no tiene efectos secundarios y no trata los fallos como errores.

1. Abra la vista **Propiedades avanzadas** de la transformación del consumidor de servicio web.

2. Desactive **Tratar fallo como error**.
3. Desactive **Tiene efectos secundarios**.
4. Haga clic en el botón **Abrir** de la propiedad **optimización de inserción**.
5. Seleccione el nombre del puerto de filtro en el cuadro de diálogo Entrada optimizada. Puede seleccionar varios puertos de filtro.
6. Haga clic en la columna **Salida**.
7. Para cada puerto de filtro, seleccione el puerto de salida que contiene la columna filtrada en la respuesta de servicio web.
8. Especifique un valor predeterminado para cada puerto de filtro.

**Nota:** No puede configurar un valor predeterminado para un puerto del consumidor de servicio web a menos que sea un puerto de filtro.

## Cómo crear una transformación de consumidor de servicio web

Puede crear una transformación de consumidor de servicio web reutilizable o no reutilizable. Pueden existir transformaciones reutilizables en varias asignaciones. Las transformaciones no reutilizables existen en una única asignación.

Puede crear transformaciones de consumidor del servicio web para un enlace SOAP 1.1 y SOAP 1.2 desde un solo objeto WSDL.

1. Para crear una transformación, utilice uno de los siguientes métodos:

Opción	Descripción
Reutilizable	Seleccione un proyecto o una carpeta en la vista <b>Explorador de objetos</b> . Haga clic en <b>Archivo &gt; Nuevo &gt; Transformación</b> . Seleccione la transformación de consumidor de servicio web y haga clic en <b>Siguiente</b> .
No reutilizable	En una asignación o en un mapplet, arrastre una transformación de consumidor de servicio web desde la paleta Transformación hasta el editor.

A continuación, aparece el cuadro de diálogo **Nueva transformación de consumidor de servicio web**.

2. Explore y seleccione un objeto de datos WSDL para definir la solicitud de servicio web y los mensajes de respuesta.  
Si el WSDL no se encuentra en el repositorio, puede importarlo desde el cuadro de diálogo Nueva transformación de consumidor de servicio web.
3. Explore y seleccione una operación del WSDL.  
Puede elegir una operación que tenga un enlace SOAP 1.1 o 1.2.
4. Haga clic en **Siguiente**.  
Aparece la pantalla **Asignar puertos a la entrada de la operación**. El área **Puertos** muestra los puertos de entrada y los grupos de entrada de transformación. El área **Entrada de operación** muestra la jerarquía de mensajes de solicitud.
5. Defina los grupos y los puertos de entrada y asigne los puertos de entrada a nodos de entrada de operación.

6. Haga clic en **Siguiente**.

Aparece la pantalla **Asignar puertos de la salida de la operación**. El área **Salida de operación** muestra la jerarquía de mensajes de respuesta. El área **Puertos** muestra los puertos de salida y los grupos de salida de transformación.

7. Defina los grupos y los puertos de salida y asigne nodos de salida de operación a los puertos de salida.
8. Haga clic en **Finalizar**.
9. Haga clic en la vista **Avanzadas** para configurar las propiedades de la transformación y la conexión de servicios web.

## TEMAS RELACIONADOS

- [“Cómo asignar puertos de entrada a la entrada de operación” en la página 343](#)
- [“Cómo asignar la salida de operación a puertos de salida” en la página 346](#)
- [“Propiedades avanzadas de la transformación de consumidor de servicio web” en la página 348](#)

# Ejemplo de transformación de consumidor de servicio web

Su organización necesita poner la información sobre pedidos de la línea de productos RT100 a disposición de la organización de ventas. El equipo de ventas necesita consultar a diario el resumen de los pedidos y los detalles de estos.

Cree un objeto de datos lógicos que exponga la información diaria sobre pedidos en tablas virtuales. La asignación de lectura contiene una transformación de consumidor de servicio web que devuelve los pedidos más recientes de RT100. La transformación de consumidor de servicio web consume un servicio web que devuelve el resumen diario de pedidos e información detallada de los pedidos para la línea de productos RT100.

## Archivo de entrada

El archivo de entrada es un archivo sin formato que contiene el número de la línea de producto.

Cree un objeto de datos físicos para definir el archivo de entrada. El archivo tiene un campo, **Product\_Line**. El valor del campo es RT100. Defina la ubicación del objeto de datos físicos en la vista **Propiedades de tiempo de ejecución**.

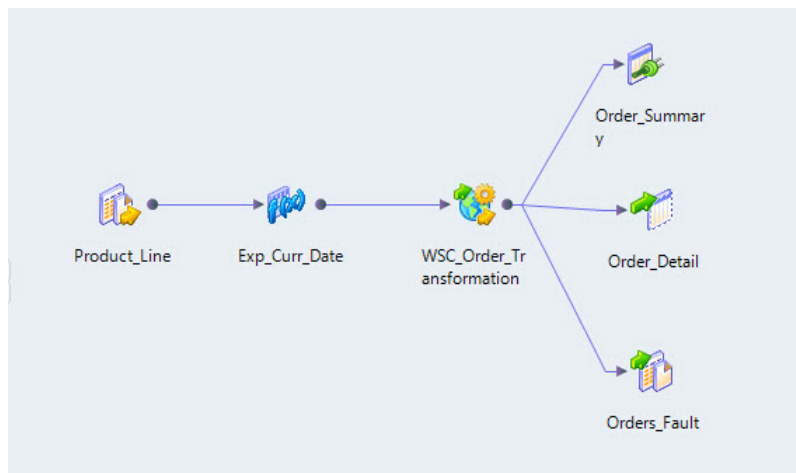
## Modelo de objeto de datos lógicos

Un analista empresarial de su organización crea un modelo de datos lógicos que describe las estructuras del resumen de pedidos y de la tabla de detalles de pedidos. El modelo de datos lógicos contiene los objetos de datos lógicos **Order\_Summary** y **Order\_Detail**.

El analista crea un esquema en una herramienta de modelado que define el modelo de datos lógicos. El modelo de datos lógicos se importa desde el esquema y se crean los objetos de datos **Order\_Summary** y **Order\_Detail**.

## Asignación de objetos de datos lógicos

La asignación de objetos de datos lógicos describe cómo acceder a datos mediante el objeto de datos lógicos.



La asignación de lectura contiene los siguientes objetos:

### **Product\_Line**

Archivo sin formato de entrada que contiene el número de línea de producto.

### **Transformación Exp\_Curr\_Date**

Transformación de expresión que devuelve la fecha actual y una clave primaria para el grupo de entrada del nivel raíz de la transformación de consumidor de servicio web.

### **Transformación WSC\_Order**

Transformación de consumidor de servicio web que consume un servicio web para recuperar información sobre pedidos. La transformación pasa la línea de productos y la fecha actual al servicio web en el mensaje de solicitud. La transformación recibe la información sobre los pedidos desde un servicio web en el mensaje de respuesta.

### **Tabla Order\_Summary**

Un objeto de datos lógicos que contiene información sobre pedidos como Order\_No, Customer\_Id, Qty y Order\_Date.

### **Tabla Order\_Detail**

Un objeto de datos lógicos que contiene información sobre pedidos como Order\_No, Product\_Id, Qty y Status.

### **Orders\_Fault**

Archivo sin formato de salida que recibe mensajes de fallo genéricos.

## Transformación de consumidor de servicio web

La transformación de consumidor de servicio web recibe como entrada una línea de productos, una fecha y un número de secuencia. La transformación consume la operación de servicio web Get\_Order\_Info para recuperar la información del pedido.

Cuando se crea la transformación de consumidor de servicio web, seleccione un objeto de datos WSDL que describa los mensajes de solicitud y de respuesta del servicio web. Un mensaje de servicio web contiene grupos jerárquicos de elementos XML. Un elemento puede contener otros elementos. Algunos elementos

pueden ocurrir varias veces. Cree la transformación desde el objeto WSDL Order\_Info que se encuentra en el repositorio.

Configure los puertos de entrada de transformación y asigne los puertos a la jerarquía de entrada de operación. Asigne nodos de la jerarquía de salida de operación a los puertos de salida. Defina la conexión de servicios web y las propiedades de tiempo de ejecución.

## Asignación de entrada de transformación

Cuando se muestra la asignación de entrada en la vista **Puertos**, se pueden definir puertos de entrada y pueden asignarse a nodos en la entrada de operación.

El área **Puertos** de transformación tiene un grupo raíz y un grupo Pedido. El grupo raíz es el grupo de entrada Solicitud. Añada un puerto al grupo de entrada Solicitud para representar la clave primaria.

El grupo Pedido tiene los puertos de entrada **Select\_Date** y **Select\_Product\_Line**.

Asigne los puertos de entrada a los nodos **Order\_Date** y **Product\_Line** en el área **Entrada de operación**.

El área **Entrada de operación** define el mensaje de solicitud que la transformación de consumidor de servicio web pasa al servicio web. De forma predeterminada, los nodos aparecen en el área **Entrada de operación**.

## Asignación de salida de transformación

Cuando se muestra la asignación de salida en la vista **Puertos**, se pueden definir los puertos de salida asignando nodos de la salida de operación a los grupos de salida de transformación.

El servicio web devuelve la siguiente jerarquía en un mensaje de respuesta de un servicio web:

```
Response
  Orders
    Order
      Key_Order
      Order_ID
      Order_Date
      Customer_ID
      Total_Qty
      Order_Details
        Order_Detail
          Product_ID
          Description
          Qty
          Status
```

El servicio web devuelve múltiples pedidos. El pedido es un nodo de ocurrencia múltiple en el nivel Orders. Para cada pedido, el servicio web puede devolver múltiples detalles del pedido. Order\_Detail es un nodo de ocurrencia múltiple en el nivel Order\_Details.

**Nota:** Developer Tool añade el nodo Key\_Order en la interfaz del usuario. Puede asignar la clave a grupos de salida para definir las relaciones entre los grupos. Para este ejemplo, Order\_ID es la clave primaria en Order y es la clave externa en Order\_Details.

Cree los siguientes grupos de salida en el área **Puertos**:

```
Order
  Order_ID
  Order_Date
  Customer_ID
  Total_Qty

Order_Detail
  Order_ID
  Product_ID
```

Description  
Qty  
Status

El servicio de integración de datos escribe una fila desde el grupo Order siempre que cambia el valor de Order\_ID.

El servicio de integración de datos escribe una fila desde el grupo Order\_Detail siempre que cambian los valores de Order\_ID y Product\_ID.

## Propiedades avanzadas de la transformación

Configure las siguientes propiedades avanzadas para la transformación de consumidor de servicio web:

### Habilitar control de fallos SOAP genéricos

Agrega puertos de salida que reciben mensajes de fallos SOAP.

### Conexión

Seleccione una conexión de servicios web para acceder al servicio web.

### Permitir compresión

La transformación de consumidor de servicio web comprime los mensajes web mediante GZIP.

## CAPÍTULO 33

# Análisis de mensajes SOAP del servicio web

Este capítulo incluye los siguientes temas:

- [Resumen del análisis de mensajes SOAP del servicio web, 359](#)
- [Interfaz de usuario de transformación, 360](#)
- [Configuración de salidas de ocurrencia múltiple, 361](#)
- [Cómo analizar elementos anyType, 363](#)
- [Cómo analizar tipos derivados, 364](#)
- [Análisis de elementos QName, 365](#)
- [Cómo analizar grupos de sustitución, 365](#)
- [Cómo analizar construcciones XML en mensajes SOAP, 366](#)

## Resumen del análisis de mensajes SOAP del servicio web

El servicio de integración de datos genera datos de filas cuando analiza un mensaje SOAP en una transformación de servicio web.

La transformación de entrada de servicios web y la transformación de consumidor de servicio web son transformaciones de servicios web que analizan mensajes SOAP.

Para configurar una transformación para que analice un mensaje SOAP, cree puertos de salida en una estructura similar a la de la jerarquía de mensajes SOAP. Asigne los nodos de la jerarquía de mensajes SOAP a los puertos.

Puede configurar grupos normalizados de puertos de salida, grupos desnormalizados y grupos pivotados de puertos. Cuando el mensaje SOAP contiene tipos derivados, elementos anyType o grupos de sustitución, puede configurar grupos de salida diferentes según los tipos que pueden aparecer en la instancia del mensaje SOAP.

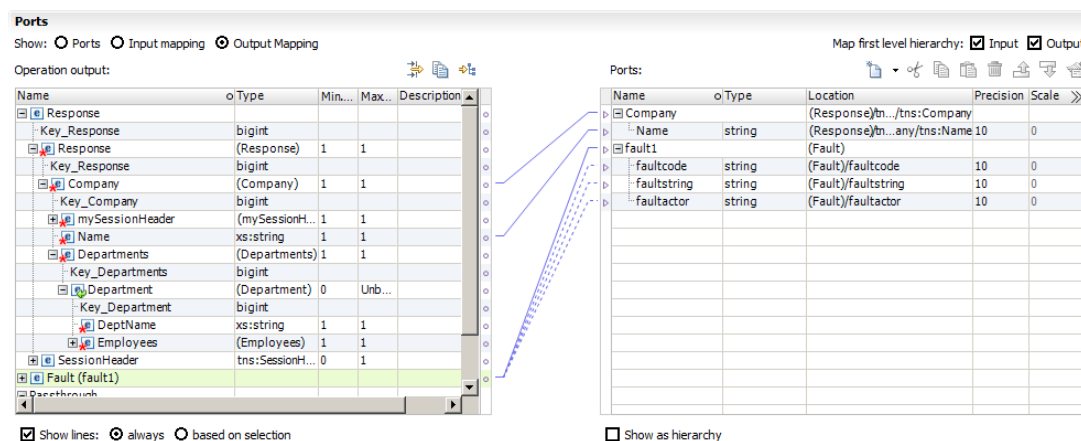
## TEMAS RELACIONADOS

- [“Asignación de salida de transformación de consumidor de servicio web” en la página 344](#)

# Interfaz de usuario de transformación

La transformación de consumidor de servicio web y la transformación de entrada de servicios web ofrecen una interfaz de usuario que puede utilizarse para asignar datos del mensaje SOAP a los puertos de salida de la transformación.

La siguiente figura muestra una asignación entre los nodos de mensaje SOAP 1.1 y los puertos de salida en una transformación de consumidor de servicio web:



## Área Operación

El área Operación contiene la jerarquía de mensajes SOAP. Los nodos complejos o los nodos de ocurrencia múltiple definen los niveles de jerarquía en la estructura. Developer Tool añade claves a los niveles que definen las relaciones principales-secundarias entre ellas.

En la figura anterior, la jerarquía de mensajes SOAP tiene los siguientes niveles:

### Respuesta o Solicitud

Nivel que representa la raíz del mensaje de respuesta o de solicitud.

### Empresa

Nivel superior de los datos de solicitud.

### Departamentos

Departamentos de ocurrencia múltiple en el ámbito de la empresa.

### Empleados

Empleado es un elemento complejo dentro de un departamento.

### Grupo de fallos

Grupo de mensajes de fallo que recibe mensajes de error.

## Área Puertos

Puede asignar datos desde niveles del mensaje SOAP a puertos de salida. Cada grupo de puertos de salida puede relacionarse con otros grupos de salida con relaciones de clave externa primaria.

En la figura anterior, la transformación posee grupos de puertos salida que corresponden a los grupos de nodos del mensaje SOAP.

## Configuración de salidas de ocurrencia múltiple

Cuando una transformación de entrada o una transformación de consumidor de servicio web devuelve datos de ocurrencia múltiple, puede configurar los puertos de salida en configuraciones diferentes.

Puede configurar datos de salida normalizados, datos de salida pivotados o datos de salida desnormalizados.

Por ejemplo, un mensaje SOAP contiene los elementos complejos Departamentos y Empleados. Cada departamento contiene varios empleados. Departamentos es un elemento principal de Empleados.

El mensaje SOAP contiene la siguiente jerarquía de elementos:

```
Departments
  Department_ID
  Department_Name
  Employees
    Employee_ID
    Employee_Name
```

### Salida relacional normalizada

Cuando se crean datos de salida normalizados, los valores de los datos no se repiten en un grupo de salida. Se crea una relación exclusiva entre los niveles de jerarquía del mensaje SOAP y los grupos de salida de los puertos.

Cuando el mensaje SOAP contiene un nivel de jerarquía principal Departamentos y un nivel de jerarquía secundario Empleados, puede crear los siguientes grupos de puertos:

```
Departments
  Department_Key
  Department_ID
  Department_Name

Employees
  Department_Key
  Employee_ID
  Employee_Name
```

Department\_Key es una clave generada que relaciona el grupo de salida Empleados con un grupo Departamento.

### Claves generadas

Cuando se agrega un grupo de salida, Developer Tool relaciona el grupo de salida con otro grupo de salida con una clave generada. Developer Tool añade una clave bigint al grupo principal y al grupo secundario. En tiempo de ejecución, el servicio de integración de datos crea los valores de clave para las claves generadas.

#### Ejemplo

La jerarquía de mensajes SOAP tiene los siguientes nodos:

```
Departments
  Dept_Key
```

```

Dept_Num
Dept_Name

Employees
  Dept_FK
  Employee_Num
  Employee_Name

```

Cuando se crea un grupo de salida de puertos para Departamentos, se asigna el nodo Departamentos a un campo vacío del área Puertos. Developer Tool crea el siguiente grupo de salida:

```

Departments
  Dept_Num
  Dept_Name

```

Cuando se asigna el nodo Empleados a un campo vacío del área Puertos, Developer Tool le pedirá que relacione el grupo Empleados con el grupo Departamentos. Puede relacionar el grupo Empleados con más de un grupo. Developer Tool añade una clave a cada grupo.

Developer Tool crea los siguientes grupos y claves generadas:

```

Departments
  Key_Departments
  Dept_Num
  Dept_Name

Employees
  Key_Departments
  Employee_Num
  Employee_Name

```

**Nota:** No es necesario que asigne nodos a las claves generadas. El servicio de integración de datos crea los valores de clave en tiempo de ejecución.

Developer Tool puede crear claves generadas en varios niveles de un grupo de salida. El grupo Empleados puede contener los siguientes puertos:

```

Employees
  Key_Employees
  Key_Departments
  Key_Managers
  Employee_Num
  Employee_Name

```

Key\_Departments y Key\_Managers son las claves generadas que apuntan a los grupos principales. Key\_Employees es una clave generada para el grupo Empleados. Key\_Employees aparece cuando se relaciona un grupo secundario con el grupo Empleados.

## Salida relacional desnormalizada

Se puede desnormalizar una salida relacional. Cuando se desnormalizan los datos de salida, los valores de elemento del grupo principal se repiten para cada elemento secundario.

Para desnormalizar datos de salida, asigne nodos del nivel de jerarquía principal al grupo secundario de los puertos de salida.

El siguiente ejemplo muestra Department\_ID y Department\_Name en el grupo de salida Empleados:

```

Employees
  Department_ID
  Department_Name
  Employee_ID
  Employee_Name

```

Department\_ID y Department\_Name se repiten para cada empleado del departamento:

Department_ID	Department_Name	Employee_ID	Employee_Name
100	Contabilidad	56500	Kathy Jones
100	Contabilidad	56501	Tom Lyons
100	Contabilidad	56509	Bob Smith

## Salida relacional pivotada

Puede incluir un número específico de elementos de ocurrencia múltiple en un grupo de salida.

Para pivotar elementos de ocurrencia múltiple, asigne el elemento secundario de ocurrencia múltiple al grupo principal de los puertos de salida. Developer Tool le insta a definir el número de elementos secundarios que incluir en el principal.

El siguiente ejemplo muestra dos instancias de Employee\_ID en el grupo principal Departamentos:

```
Departments
  Department_ID
  Department_Name
  Employee_ID1
  Employee_ID2
```

## Cómo analizar elementos anyType

El elemento anyType representa una elección de todos los tipos globales en un WSDL o esquema. Cuando se asignan nodos a puertos en Developer Tool, se eligen los tipos que deben aparecer en el mensaje SOAP para el elemento anyType. Debe reemplazar un elemento anyType en el mensaje SOAP con un tipo complejo o xs:string. Cree grupos de puertos para cada tipo que elija.

Debe elegir un tipo para asignar datos a puertos de salida. Si el WSDL o el esquema no contiene un tipo de global, Developer Tool reemplaza el elemento anyType con xs:string.

Para elegir un tipo de elemento complejo en el área Operación, haga clic en **Elegir** en la columna **Tipo** del elemento anyType. Aparece una lista de tipos complejos disponibles y xs:string.

Cuando se reemplaza un elemento anyType con tipos derivados, el servicio de integración de datos llena uno a uno los elementos para un tipo. El mensaje SOAP no contiene datos para el tipo base y el tipo derivado al mismo tiempo.

### Ejemplo de tipos derivados

El WSDL contiene un elemento anyType. Se reemplaza el elemento con AddressType y un tipo derivado llamado USAddressType. La jerarquía de mensajes SOAP tiene los siguientes grupos:

```
Address:AddressType (base type)
  Address: AddressType
    Street
    City

Address:USAddressType (derived type)
  Street
  City
  State
  ZipCode
```

El mensaje SOAP contiene los siguientes datos:

```
<address xsi: type ="AddressType">
<street>1002 Mission St.</street>
<city>san jose</city>
</address>

<address xsi:type="USAddressType">
<street>234 Fremont Blvd</street>
<city>Fremont</city>
<zip>94556</zip>
<state>CA</state>
</address>
```

El servicio de integración de datos devuelve una fila para xsi: AddressType:

Calle	Ciudad
1002 Mission St.	San José

El servicio de integración de datos devuelve una fila para el tipo derivado xsi: USAddressType:

Calle	Ciudad	Estado	Código postal
234 Fremont Blvd.	Sunnyvale	CA	94556

El servicio de integración de datos no llena AddressType si el tipo es xsi: USAddressType.

## Cómo analizar tipos derivados

Puede analizar mensajes SOAP que contienen tipos derivados. Cuando se definen los puertos que reciben los datos desde el mensaje SOAP, elija los tipos que pueden aparecer en el mismo. Los elementos en los tipos que se eligen determinarán los puertos que necesita crear.

Por ejemplo, el WSDL puede contener un AddressType y un tipo derivado llamado USAddressType. Puede crear los siguientes grupos en el área Operación de Developer Tool:

```
Address
  Address: AddressType
    Street
    City

  Address:USAddressType
    Street
    City
    State
    ZipCode
```

El mensaje SOAP puede contener los siguientes datos:

```
<address>
<street>1002 Mission St.</street>
<city>san jose</city>
</address>

<address xsi:type="USAddressType">
<street>234 Fremont Blvd</street>
<city>Fremont</city>
<zip>94556</zip>
<state>CA</state>
</address>

<address xsi:type="USAddressType">
```

```

<street>100 Cardinal Way</street>
<city>Redwood City</city>
<zip>94536</zip>
<state>CA</state>
</address>

<address>
<street>100 El Camino Real</street>
<city>Sunnyvale</city>
</address>

```

El servicio de integración de datos devuelve las siguientes filas para el tipo base, Address:

Calle	Ciudad
1002 Mission St.	San José
234 Fremont Blvd	Sunnyvale
100 Cardinal Way	Redwood City
100 El Camino Real	Sunnyvale

El servicio de integración de datos devuelve las siguientes filas para el tipo base, USAddress:

Calle	Ciudad	Estado	Código postal
234 Fremont Blvd.	Sunnyvale	CA	94556
100 Cardinal Way	Redwood City	CA	94536

El servicio de integración de datos devuelve todas las direcciones en el tipo base. El servicio de integración de datos devuelve direcciones de EE.UU. en el tipo derivado. El tipo derivado incluye los elementos Calle y Ciudad que USAddressType hereda del tipo base.

## Análisis de elementos QName

Cuando el servicio de integración de datos analiza elementos QName en el mensaje SOAP, actualiza valores QName que pertenecen al espacio de nombres del esquema para utilizar el prefijo de espacio de nombres definido en el esquema. De lo contrario, el servicio de integración de datos no actualiza el valor del elemento.

Por ejemplo, el esquema tiene el prefijo de espacio de nombres `tns` definido para el espacio de nombres `"http://usuario/prueba"`. El mensaje SOAP tiene el prefijo de espacio de nombres `mytns` definido para el mismo espacio de nombres. Cuando el servicio de integración de datos analiza el valor QName `mytns:myelement`, el valor cambia a `tns:myElement`.

Cuando el servicio de integración de datos genera elementos QName en el mensaje SOAP, no actualiza el valor del elemento.

## Cómo analizar grupos de sustitución

Un grupo de sustitución reemplaza un elemento por otro del mismo grupo. Los grupos de sustitución son similares a los tipos derivados, excepto que cada definición de elemento incluye un nombre de grupo de sustitución.

Puede configurar un grupo de salida de puertos que recibe elementos de un tipo específico en un grupo de sustitución. Puede crear un grupo de salida de puertos diferente que recibe un elemento desde otro tipo en el grupo de sustitución.

# Cómo analizar construcciones XML en mensajes SOAP

Un mensaje SOAP puede contener construcciones XML como elementos de selección, de lista y de unión.

Con ciertas limitaciones, las transformaciones de servicio web pueden analizar mensajes SOAP con estas construcciones.

## Elemento de selección

Un elemento de selección restringe un elemento secundario a uno de los elementos en la declaración <selección>.

El siguiente texto muestra un elemento persona que es un empleado o un contratista:

```
<xs:element name="person">
  <xs:complexType>
    <xs:choice>
      <xs:element name="employee" type="employee"/>
      <xs:element name="contractor" type="contractor"/>
    </xs:choice>
  </xs:complexType>
</xs:element>
```

Puede asignar elementos de selección mediante los siguientes métodos:

- Crear puertos de salida para cada elemento de selección en un grupo de salida. Algunos elementos tendrán valores nulos en la fila de salida.
- Crear un grupo de salida para cada selección. Para el ejemplo anterior, cree un grupo de empleados y un grupo de contratistas. El servicio de integración de datos genera una fila basada en el elemento que aparece en el mensaje SOAP.

## Elemento de lista

Una lista es un elemento XML que puede contener varios valores de tipo simple, como "Lunes Martes Miércoles".

El servicio de integración de datos puede devolver una lista como un valor de cadena. Cuando el mensaje SOAP contiene una lista, no se pueden asignar elementos de la lista a filas de salida individuales. Puede configurar una transformación de expresión para separar los elementos de la lista si necesita separarlas en una asignación.

## Elemento de unión

El elemento de unión es un tipo simple que es una unión de más de un tipo.

El siguiente texto muestra un elemento Tamaño que es la unión de dos tipos simples, size\_no y size\_string:

```
<xs:element name="Size">
  <xs:simpleType>
    <xs:union memberTypes="size_no size_string" />
  </xs:simpleType>
</xs:element>
```

Para asignar el tamaño a un puerto de salida, cree un puerto para el tamaño. Configure el puerto de salida como una cadena. Puede configurar otra transformación en la asignación para convertir los datos a otro tipo.

# Generación de mensajes SOAP del servicio web

Este capítulo incluye los siguientes temas:

- [Resumen de la generación de mensajes SOAP del servicio web, 367](#)
- [Interfaz de usuario de transformación, 368](#)
- [Relaciones entre puerto y nivel de jerarquía , 369](#)
- [Claves, 370](#)
- [Asignar puertos, 371](#)
- [Cómo pivotar puertos de ocurrencia múltiple , 373](#)
- [Asignación de datos desnormalizados, 374](#)
- [Tipos derivados y sustitución de elementos, 376](#)
- [Cómo generar construcciones XML en mensajes SOAP, 378](#)

## Resumen de la generación de mensajes SOAP del servicio web

El servicio de integración de datos genera datos XML a partir de grupos de datos de entrada cuando genera un mensaje SOAP. Cuando se crea una transformación de consumidor de servicio web, una transformación de salida de servicio web o una transformación de fallo, se configuran los puertos de entrada que deben asignarse a la jerarquía de mensajes SOAP.

Para configurar una transformación para que genere un mensaje SOAP, cree grupos de puertos de entrada y asigne cada grupo a un grupo de la jerarquía de mensajes SOAP. Un WSDL o esquema define la estructura del mensaje SOAP.

Puede configurar grupos de datos en el mensaje SOAP a partir de datos de entrada desnormalizados. También puede pivotar datos de entrada de ocurrencia múltiple en nodos de ocurrencia múltiple en los mensajes SOAP.

Puede asignar datos a tipos derivados, elementos anyType o grupos de sustitución en un mensaje SOAP. Debe elegir qué tipos pueden aparecer en el mensaje SOAP cuando se define una transformación. Los tipos que elija determinarán los puertos de entrada que necesita crear.

Cuando se visualiza la jerarquía de mensajes SOAP en Developer Tool, la jerarquía contiene claves. Las claves no aparecen en el mensaje SOAP. El servicio de integración de datos utiliza claves para definir

relaciones entre grupos principales y secundarios en el mensaje SOAP. Para configurar valores clave, asigne datos de entrada a claves en el mensaje SOAP.

## TEMAS RELACIONADOS

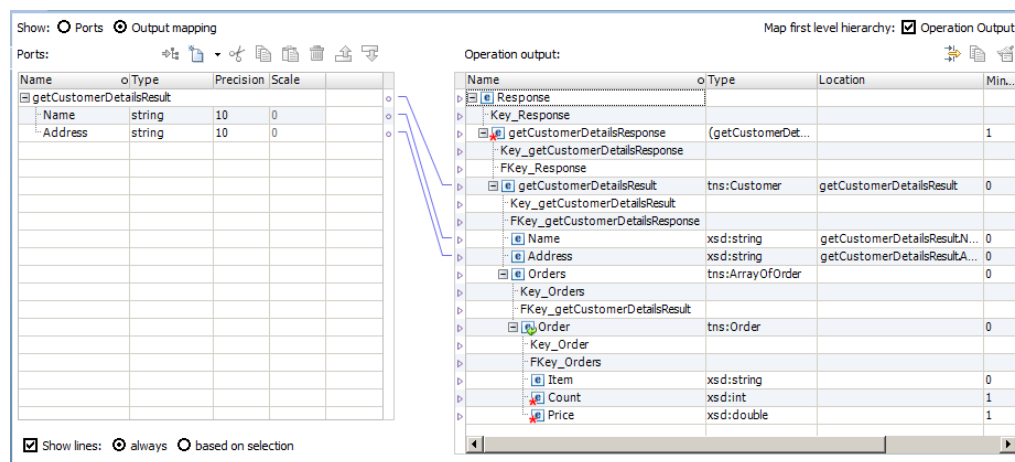
- [“Asignación de entrada de transformación de consumidor de servicio web” en la página 342](#)

# Interfaz de usuario de transformación

La transformación de salida, la transformación de fallo y la transformación de consumidor de servicio web de un servicio web contienen una interfaz de usuario que permite configurar el mensaje SOAP.

Para configurar una transformación para que genere un mensaje SOAP, cree puertos de entrada en una estructura similar a la de la jerarquía de mensajes SOAP. El WSDL o esquema determina la estructura de la jerarquía. Asigne cada puerto de entrada a un nodo en el mensaje SOAP.

La siguiente figura muestra una asignación entre los puertos de entrada y los nodos de mensaje SOAP en una transformación de salida de servicio web:



## Área Puertos de entrada

Cree grupos de puertos de entrada en el área **Puertos de entrada**. Incluya puertos de entrada para cada nivel en la jerarquía de mensajes SOAP que desee asignar.

Debe crear un grupo de entradas Respuesta o Solicitud, así como los grupos secundarios que reciben los datos.

Cuando cree grupos de puerto de entrada, defina una clave primaria en cada grupo principal. Defina una clave externa en cada grupo secundario. La clave externa relaciona el grupo con un grupo principal.

No es necesario que defina claves para el nivel Respuesta o el nivel raíz WSDL salvo que esté pasando datos en el nivel raíz WSDL. Por ejemplo, el nivel raíz puede contener encabezados HTTP.

Puede crear grupos de puertos similares a los siguientes grupos para clientes y pedidos:

```
Response
  Response_Key

Customer_Details_Root
```

```

Key_Cust_Det
FK_Response_Key

Customer
  Customer_ID
  FK_Cust_Det
  Name
  Address

  Orders
    Order_Num
    FK_Cust_ID

    Order_Items
      Order_Num
      Item
      Count
      Price

```

## Área Operación

El área **Operación** muestra los elementos en la jerarquía de mensajes SOAP definidos por el WSDL o el esquema. El mensaje SOAP no tiene que contener todos los elementos del WSDL o del esquema. El mensaje contiene los datos que se asignan de los puertos de entrada.

Los nodos de ocurrencia múltiple y los nodos complejos definen los niveles de jerarquía en la estructura de mensajes SOAP. Developer Tool añade claves a los niveles para crear relaciones jerárquicas entre ellos. Todos los niveles de la jerarquía, excepto los niveles hoja, poseen una clave primaria. Cada nivel secundario tiene una clave externa que apunta a un nivel principal. Las claves que aparecen en la jerarquía de mensajes SOAP no aparecen en la instancia de mensaje SOAP. El servicio de integración de datos necesita que las claves tengan valores para relacionar los niveles de los datos cuando genera el mensaje SOAP.

La columna **Ubicación** contiene el nombre de grupo y el puerto de entrada que contiene los datos para un elemento en el mensaje SOAP. La columna **Ubicación** está vacía hasta que se asigna un puerto de entrada al nodo.

En la figura precedente, el mensaje SOAP contiene una instancia individual de detalles y pedidos de cliente. El grupo Pedidos contiene un elemento de ocurrencia múltiple llamado Pedido. La jerarquía de mensajes SOAP tiene los siguientes niveles que se relacionan mediante una clave:

```

Response
  GetCustomerDetailsResponse
    GetCustomerDetailsResult
      Orders
        Order

```

El nivel Respuesta representa la raíz del mensaje de respuesta. El servicio de integración de datos requiere que este nivel anexe encabezados al mensaje SOAP.

El nivel GetCustomerDetailsResponse es la raíz del mensaje.

## Relaciones entre puerto y nivel de jerarquía

Cuando se asignan puertos de entrada a la jerarquía de mensajes SOAP, se mantiene una relación entre un grupo de entrada y un nivel de la jerarquía de mensajes SOAP. Por ejemplo, puede tener dos grupos de entrada, Departamento y Empleado.

El grupo de entrada Departamento recibe las siguientes filas:

Dept_num	Nombre	Ubicación
101	HR	Nueva York
102	Producto	California

El grupo de entrada Empleado recibe las siguientes filas:

Dept_num	Empleado
101	Alice
101	Bob
102	Carol
102	Dave

Asigne el número de departamento en el grupo Empleado como clave externa que establezca la relación entre el grupo Departamento y el grupo Empleado. El número de departamento aparece en el nivel de jerarquía de departamento, pero no el nivel de empleado.

El mensaje SOAP contiene la siguiente estructura XML:

```
<department>
  <dept_num>101</dept_num>
  <name>HR</name>
  <location>New York</location>

  <employee>
    <name>Alice</name>
  </employee>

  <employee>
    <name>Bob</name>
  </employee>
</department>

<department>
  <dept_num>102</dept_num>
  <name>Product</name>
  <location>California</location>

  <employee>
    <name>Carol</name>
  </employee>

  <employee>
    <name>Dave</name>
  </employee>
</department>
```

## Claves

Una jerarquía de mensajes SOAP incluye claves. El servicio de integración de datos necesita claves para construir la jerarquía XML en el mensaje SOAP.

Debe asignar datos de puertos de entrada a las claves en la jerarquía de mensajes SOAP. Asigne datos a las claves en cada nivel al que suministre datos. Cuando tenga un nodo de ocurrencia múltiple, tendrá que relacionar el nodo con un nodo principal.

Las claves aparecen en el mensaje SOAP sin tipos. Cualquier puerto que se asigna a una clave debe ser del tipo de datos string, integer o bigint. La clave primaria del grupo principal y la clave externa de cada grupo

secundario deben tener el mismo tipo de datos, la misma precisión y la misma escala. Puede asignar las claves generadas a las claves de mensajes SOAP.

Puede asignar un puerto a un nodo y a una clave que estén en el mismo nivel de jerarquía. Por ejemplo, puede asignar Employee\_ID a un nodo en el mensaje SOAP y, además, puede asignarlo a una clave en el nivel Empleado.

Cuando los nodos de dos grupos de la jerarquía posean una relación principal-secundario, realice las siguientes tareas:

- Asigne un puerto a la clave primaria en el grupo del nodo principal.
- Asigne un puerto a la clave externa en el grupo del nodo secundario.

También puede asignar claves primarias a puertos de entrada para eliminar filas con una clave primaria con el valor null o con claves primarias duplicadas.

Puede crear una clave compuesta en un mensaje SOAP mediante la asignación de varios puertos a la misma clave. Utilice claves compuestas cuando tenga que desnormalizar datos y mantener claves únicas para ciertas combinaciones de valores de ocurrencia múltiple. Puede crear claves compuestas que contengan valores de cadena, de bigint o de entero.

**Nota:** Puede incluir una transformación de expresión en la asignación de operación para generar valores de clave.

### Ejemplo de clave compuesta

Configure una clave de división-departamento única a partir de los siguientes grupos de puertos:

```
Company
  Company_Num
  Company_Name

Division
  Company_Num
  Division_Num
  Division_Name

Department
  Division_Num
  Dept_Num
  Dept_Name
  Location
```

Dept\_Num es exclusivo dentro de una división, pero no lo es para todas las divisiones de una empresa.

Puede configurar un grupo Departamento que contenga la información sobre la división y el departamento. Configure el número de división y el número de departamento como parte de la clave compuesta:

```
Department
  Division_Num + Dept_Num (key)
  Dept_Name
  Location
```

El orden en el que se asignan los puertos determina el valor de la clave.

## Asignar puertos

Después de crear puertos de entrada, asigne cada uno de ellos a la jerarquía de mensajes SOAP. La ubicación del puerto aparece junto al nodo en el área **Operación**.

Puede asignar puertos a los siguientes tipos de nodos:

#### Nodo atómico

Un elemento o atributo simple que no tiene elementos o atributos secundarios y que no es divisible.

#### Nodo atómico de ocurrencia múltiple

Un elemento o atributo simple que ocurre varias veces en la misma ubicación de la jerarquía.

#### Nodo complejo

Un elemento que contiene otros elementos.

Si el nodo principal no tiene una ubicación, el nodo principal recibe como ubicación el nombre del grupo de entrada. Cuando el nodo principal tiene una ubicación, cada nodo del nivel jerárquico debe tener una ubicación de salida desde la misma ubicación.

Se puede asignar un nombre de grupo de entrada a un nodo principal en un nivel de jerarquía. Developer Tool actualiza el campo de ubicación para el nodo principal en la jerarquía. Developer Tool no actualiza los nodos secundarios que pertenecen al grupo en la jerarquía. Cuando se asignan puertos de entrada a los nodos secundarios, la ubicación de cada puerto de entrada debe ser la misma que la del nodo principal.

Después de asignar un grupo de entrada a un nivel de jerarquía, se puede modificar más adelante. Puede hacer clic en **Borrar** o puede eliminar las líneas entre las áreas Puertos y Operación. Para eliminar las líneas, arrastre el puntero de las líneas para seleccionarlás. Haga clic en **Eliminar**.

## Asignación de un puerto

Cuando se asigna un puerto a un nodo en el mensaje SOAP, Developer Tool proporciona resultados diferentes según el tipo de nodo al que se asigna el puerto.

En la siguiente tabla se describen los resultados que se obtienen cuando se asigna un puerto individual a nodos de destino diferentes en el área **Operación**:

Nodo de destino	Resultados
Nodo atómico	Cuando se asigna un puerto individual a un nodo y el nodo principal no tiene ninguna ubicación, el nodo recibe la ubicación del puerto. La ubicación del nodo principal recibe la ubicación del grupo de entrada para el puerto individual. Cuando se asigna un puerto individual a un nodo y el nodo principal ya posee una ubicación, puede cambiar la ubicación del nodo principal y borrar la ubicación de los demás nodos secundarios situados en el mismo nivel. La ubicación del nivel de jerarquía cambia al nombre de grupo del puerto.
Nodo atómico de ocurrencia múltiple o la clave primaria del nodo atómico de ocurrencia múltiple	Cuando se asigna un puerto individual al nodo atómico de ocurrencia múltiple, Developer Tool establece la ubicación para el nodo atómico en el grupo del puerto seleccionado.
Nodo complejo	Cuando se asigna un puerto individual a un nodo complejo, Developer Tool establece la ubicación del nodo complejo como ubicación del grupo que contiene el puerto. Developer Tool le solicita el nodo atómico de ocurrencia múltiple al que asignar el puerto. Si todos los nodos atómicos de ocurrencia individual poseen una ubicación, no podrá asignar el nodo complejo.

## Asignación de un grupo

Cuando se asigna un grupo de entrada a un nodo en el mensaje SOAP, Developer Tool proporciona distintos resultados según el tipo de nodo al que se asigna el puerto.

En la siguiente tabla se describen los resultados que se obtienen cuando se asigna un grupo a un nodo en el área **Operación**:

Nodo de destino	Resultados
Nodo atómico	No se puede asignar un grupo a un nodo atómico.
Nodo atómico de ocurrencia múltiple	Se le insta a que elija un puerto en el grupo de entrada para actualizar la ubicación para el nodo y la clave primaria.
Nodo complejo de ocurrencia múltiple	Developer Tool define la ubicación para el nodo complejo como la ubicación del grupo.

## Asignación de varios puertos

Cuando se asignan varios puertos a un nodo en el mensaje SOAP, Developer Tool proporciona resultados distintos según el tipo de nodo al que se hayan asignado los puertos. Puede asignar varios puertos al mismo tiempo si los asigna desde el mismo grupo.

En la siguiente tabla se describen los resultados para el nodo cuando se asignan varios puertos a nodos:

Nodo Destino	Resultados
Nodo atómico individual	Cuando se asignan varios puertos a un nodo individual, se actualiza la ubicación para más de un nodo atómico individual en el área <b>Operación</b> . Si la jerarquía no tiene suficientes nodos en el nivel para actualizarlos, Developer Tool solo asigna puertos para los nodos disponibles.
Nodo atómico de ocurrencia múltiple	Cuando se asignan varios puertos a un nodo atómico de ocurrencia múltiple, se pivotan los puertos en múltiples ocurrencias del nodo. Developer Tool crea instancias del nodo según el número de puertos que se asignan. Aparece un mensaje que describe el número de puertos que se han proyectado.
Nodo complejo de ocurrencia múltiple	Cuando se asignan varios puertos a un nodo complejo, debe seleccionar los nodos atómicos de ocurrencia individual que deben actualizarse. Se pivotan los puertos en múltiples ocurrencias del nodo. Developer Tool crea instancias del nodo según el número de puertos que se asignan.

## Cómo pivotar puertos de ocurrencia múltiple

Puede asignar varios puertos de entrada a un nodo de ocurrencia múltiple en el mensaje SOAP. Developer Tool pivota los datos de entrada en múltiples nodos en el mensaje SOAP.

Para cambiar el número de elementos que deben pivotarse, seleccione **Reemplazar pivote existente** en el cuadro de diálogo **Opciones de asignación**.

Si elimina una de las instancias de puerto pivotado del área **Puertos**, Developer Tool elimina todas las instancias del área **Operación**.

## Ejemplo de pivote

Un grupo de entrada puede tener las siguientes filas:

Núm.	Nombre	Ubicación	emp_name1	emp_name2	emp_name3
101	HR	Nueva York	Alice	Tom	Bob
102	Producto	California	Carol	Tim	Dave

Cada fila contiene un número de departamento y tres nombres de empleado.

Empleado es un nodo de ocurrencia múltiple en la jerarquía de mensajes SOAP. Puede asignar todas las instancias de Empleado de la fila de entrada a la jerarquía de mensajes SOAP. Seleccione todas las instancias de Empleado. Haga clic en **Asignar**. El cuadro de diálogo **Opciones de asignación** le pedirá que seleccione un nodo de la lista.

Developer Tool cambia el nodo Empleado para incluir los múltiples nodos de nombre en la jerarquía de mensajes SOAP:

```
Department
  num
  name
  location
  Employee (unbounded)
    emp_name1
    emp_name2
    emp_name3
```

El mensaje SOAP devuelve la siguiente jerarquía:

```
<department>
  <num>101</num>
  <name>HR</name>
  <location>New York</location>
  <employee>
    <emp_name>Alice</name>
  </employee>
  <employee>
    <emp_name>Tom</name>
  </employee>
  <employee>
    <emp_name>Bob</name>
  </employee>
</department>

<department>
  <num>102</num>
  <name>Product</name>
  <location>California</location>
  <employee>
    <emp_name>Carol</name>
  </employee>
  <employee>
    <emp_name>Tim</name>
  </employee>
  <employee>
    <emp_name>Dave</name>
  </employee>
</department>
```

## Asignación de datos desnormalizados

Puede asignar datos desnormalizados y pasarlos a nodos normalizados en un mensaje SOAP.

Cuando se asignan datos desnormalizados, se pasan datos de un grupo de entrada a varios nodos en la jerarquía de mensajes SOAP. Puede crear relaciones de grupo en el mensaje SOAP similares a los siguientes tipos de relaciones:

#### Relación de nodos lineal

El nodo A es el nodo principal del nodo B. El nodo B es un nodo principal del nodo C. El nodo C es el nodo principal del nodo D.

#### Relación de nodos jerárquica

El nodo A es un nodo principal del nodo B. El nodo A también es el nodo principal del nodo C. Los nodos B y C no están relacionados.

La siguiente tabla muestra filas de entrada que contienen datos desnormalizados de división y de departamento:

División	Dept_Num	Dept_Name	Teléfono	Employee_Num	Employee_Name
01	100	Contabilidad	3580	2110	Amir
01	100	Contabilidad	3580	2113	Robert
01	101	Ingeniería	3582	2114	Stan
01	101	Ingeniería	3582	2115	Jim
02	102	Instalaciones	3583	2116	José

Los datos de entrada contienen números y nombres únicos de empleados. Los datos de división y de departamento se repiten para cada empleado de un mismo departamento y división.

#### Relación de grupos lineal

Cuando se configuran puertos, puede configurar un grupo aparte para División, Departamento y Empleado. División es un grupo principal de Departamento, y Departamento es el grupo principal de Empleado. Puede configurar grupos en la siguiente estructura lineal:

```

Division
  Division_Key
  Division_Num
  Division Name

  Department
    Department_Key
    Division_FKey
    Dept_Num
    Dept_Name
    Phone

    Employee
      Department_Fkey
      Employee_Num
      Employee_Name

```

El mensaje SOAP contiene instancias únicas de División y de Departamento, aunque Division\_Num y Dept\_Num se repiten en los datos de entrada. Defina Division\_Num como clave primaria en el grupo División. Defina Dept\_Num como clave primaria en el grupo Departamento.

#### Relación de grupos jerárquica

Puede crear una jerarquía de grupo que contenga el grupo principal División y los grupos secundarios Departamento y Empleado. Departamento y Empleado no mantienen una relación clave primaria-clave externa. Departamento y Empleado son grupos secundarios de División. Puede configurar los grupos en la siguiente estructura:

```

Division
  Division_Key
  Division_Num

```

```

Division_Name

Department
  Division_FKey
  Dept_Num
  Dept_Name

Employee
  Division_FKey
  Employee_Num
  Employee_Name

```

## Tipos derivados y sustitución de elementos

Puede asignar puertos de entrada a tipos derivados complejos, elementos anyType y grupos de sustitución en un mensaje SOAP. El mensaje puede incluir elementos para el tipo base y los tipos derivados.

En una relación de tipos, el tipo base es el tipo desde el cual se deriva otro tipo. Un tipo derivado hereda elementos desde el tipo base. Un tipo complejo extendido es un tipo derivado que hereda elementos desde un tipo base e incluye elementos adicionales. Un tipo complejo restringido es un tipo derivado que restringe algunos elementos del tipo base.

### Cómo generar tipos derivados

Cuando un WSDL o esquema incluye tipos derivados, debe elegir los tipos que desea incluir en el mensaje SOAP.

Por ejemplo, el WSDL define un tipo base AddressType. El WSDL también contiene un USAddressType y un UKAddressType que son los AddressTypes derivados.

Cada tipo contiene los siguientes elementos:

- AddressType: calle, ciudad
- USAddressType (extiende AddressType): estado, zipCode
- UKAddressType (extiende AddressType): postalCode, país

Cuando se elige un USAddressType en el área Operación, Developer Tool crea un grupo para el elemento USAddressType en el mensaje SOAP. El grupo incluye la calle y la ciudad desde la dirección base y el estado y el zipCode para el USAddress. Los tipos derivados que extienden tipos base siempre incluyen los elementos desde el tipo base.

Si elige todos los tipos derivados disponibles para el mensaje SOAP, Developer Tool crea grupos similares a los siguientes en la jerarquía SOAP:

```

Address
  Address: Address
    Street
    City

Address:USAddressType
  Street
  City
  State
  ZipCode

Address: UKAddressType
  Street
  City

```

PostalCode  
Country

Necesitará definir grupos de puertos de entrada para Address, USAddress y UKAddress.

## Cómo generar elementos anyType y atributos

Ciertos elementos de esquema y atributos permiten que haya cualquier tipo de datos en un mensaje SOAP.

El elemento anyType representa una selección de todos los tipos conocidos globalmente. Antes de asignar un puerto a un elemento anyType en un mensaje SOAP, elija un tipo complejo disponible o xs:string. Si el WSDL o el esquema no contienen un tipo complejo, Developer Tool reemplaza el tipo de elemento anyType con xs:string.

Para elegir un tipo de elemento complejo en el área Operación, haga clic en **Elegir** en la columna **Tipo** del elemento anyType. Aparece una lista de tipos complejos disponibles y xs:string.

Los siguientes elementos y atributos permiten cualquier tipo de datos:

### elemento anyType

Permite que un elemento sea de cualquier tipo de datos en el archivo XML asociado.

### elemento anySimpleType

Permite que un elemento sea cualquier simpleType en el archivo XML asociado.

### CUALQUIER elemento de contenido

Permite que un elemento sea cualquier elemento global definido en el esquema.

### atributo anyAttribute

Permite que un elemento sea cualquier atributo ya definido en el esquema.

## Cómo generar grupos de sustitución

Utilice grupos de sustitución para sustituir un elemento por otro en un mensaje SOAP. Los grupos de sustitución funcionan de forma similar a los tipos derivados, excepto que las definiciones de elemento incluyen un nombre de grupo de sustitución.

Por ejemplo, puede tener un tipo base Dirección y los tipos derivados USAddress y UKAddress:

```
xs:element name="Address" type="xs:string"/>
<xs:element name="USAddress" substitutionGroup="Address"/>
<xs:element name="UKAddress" substitutionGroup="Address"/>
```

Cuando se configura la jerarquía de mensajes SOAP, puede elegir qué elemento sustituir por Dirección en el mensaje SOAP.

# Cómo generar construcciones XML en mensajes SOAP

Un WSDL o un esquema puede contener elementos de selección, de lista o de unión. Las transformaciones de servicio web pueden generar mensajes SOAP que contengan estos elementos.

## Elemento de selección

Un elemento de selección restringe un elemento secundario a uno de los elementos en la declaración <selección>.

Para asignar puertos a un mensaje SOAP que contenga elementos de selección, cree un grupo de entrada que incluya todos los elementos en la construcción de selección. Por ejemplo, una descripción de un artículo puede ser una dimensión o un peso:

```
item: description, choice {dimension, weight}
```

Cuando la descripción es una dimensión, la descripción es un tipo complejo que contiene la longitud, la anchura y la altura.

Cuando la descripción es un peso, la descripción es un tipo de cadena simple.

Los datos de entrada tienen las siguientes columnas y filas:

descripción	longitud	anchura	altura	peso
caja	20 cm	18 cm	15 cm	NULL
café	NULL	NULL	NULL	500 g

El mensaje SOAP contiene un grupo de artículos que contiene dimensiones o descripciones de pesos:

```
Item
  Description
    Dimension
      Length
      Width
      Height
    Weight
```

Los valores NULL en los datos de entrada se convierten en elementos ausentes en la salida XML.

El mensaje SOAP contiene los siguientes datos:

```
<item>
  <desc>box</desc>
  <dimension>
    <length>20cm</length>
    <width>18cm</width>
    <height>15cm</height>
  </dimension>
</item>

<item>
  <desc>coffee</desc>
  <weight>500g</weight>
</item>
```

## Elemento de lista

Una lista es un elemento XML que puede contener varios valores de tipo simple en el mismo elemento o atributo. El servicio de integración de datos puede procesar una lista en los datos de entrada si la lista se representa como una cadena de datos consolidados.

Si cada artículo de la lista es un elemento individual como ClassDates1, ClassDates2 y ClassDates3, el servicio de integración de datos no puede procesar los elementos como una lista. Puede utilizar una transformación de expresión para combinarlos en una cadena si necesita devolver una lista en un mensaje SOAP.

Las siguientes filas de entrada contienen un elemento de lista llamado ClassDates que contiene los días de la semana:

CourseID	Nombre	ClassDates
Matemáticas 1	Álgebra básica	Lun Mié Vie
Historia 1	Historia del mundo	Mar Jue

El servicio de integración de datos puede devolver un mensaje SOAP con la siguiente estructura XML:

```
<class>
  <courseId>Math 1</courseId>
  <name>Beginning Algebra</name>
  <classDates>Mon Wed Fri</classDates>
</class>
<class>
  <courseId>History 1</courseId>
  <name>World History</name>
  <classDates>Tue Thu</classDates>
</class>
```

## Elemento de unión

El elemento de unión es un tipo simple que una unión de más de un tipo. Cuando un mensaje SOAP contiene un elemento de unión, debe asignar un puerto de entrada individual que contenga los datos en una cadena.

Por ejemplo, el mensaje SOAP contiene un elemento llamado tamaño. Tamaño es una unión de entero y cadena:

```
<xs:element name="size">
  <xs:simpleType>
    <xs:union memberTypes="size_no size_string" />
  </xs:simpleType>
</xs:element>
```

Las cadenas de entrada contienen artículos con una descripción y un tamaño. Un artículo puede tener un tamaño numérico, como 42. Asimismo, un artículo puede tener un tamaño que es un valor de cadena, como grande, mediano o pequeño.

La siguiente tabla muestra filas de entrada con un tamaño numérico y tamaño de cadena:

Desc	Tamaño
zapatos	42
camisa	grande

Cree un puerto para el tamaño del artículo. Asigne el puerto como una cadena. El mensaje SOAP contiene los siguientes elementos:

```
<item>
  <desc>shoes</desc>
  <size>42</size>
</item>

<item>
  <desc>shirt</desc>
  <size>large</size>
</item>
```

# Transformación de media ponderada

Este capítulo incluye los siguientes temas:

- [Resumen de la transformación de media ponderada, 380](#)
- [Cómo configurar una transformación de media ponderada, 380](#)
- [Ejemplo de puntuaciones de coincidencia ponderadas, 381](#)
- [Propiedades avanzadas de Transformación de media ponderada, 381](#)

## Resumen de la transformación de media ponderada

La transformación de media ponderada es una transformación pasiva que lee puntuaciones de coincidencia de varias coincidencias de operaciones y produce una sola puntuación de coincidencia.

Puede aplicar una ponderación numérica a cada puntuación que entra en la transformación de media ponderada. Una ponderación es un valor entre cero y 1. Puede editar la ponderación aplicada a cada puntuación de entrada para ampliar o reducir su contribución a la puntuación de salida. Aplique ponderaciones que reflejen la importancia relativa de cada columna de datos en el análisis de duplicados.

Utilice la transformación de media ponderada cuando añada transformaciones de comparación a una asignación o mapplet.

**Nota:** También puede asignar ponderaciones en una transformación de coincidencia. Utilice la transformación de coincidencia para configurar estrategias de búsqueda de coincidencias y asignar ponderaciones en una transformación individual. Puede incrustar un mapplet de coincidencia en una transformación de coincidencia.

## Cómo configurar una transformación de media ponderada

Utilice la transformación de media ponderada para ajustar el total de puntuación de coincidencia que una asignación genera para una serie de operaciones de análisis de coincidencias. Edite las ponderaciones relativas de cada puerto de entrada para reflejar las prioridades de las comparaciones de datos que ha definido para el conjunto de datos de origen. Cada puerto de entrada en la transformación de media

ponderada representa una puntuación de coincidencia de salida de una estrategia de transformación de comparación.

Los pasos siguientes describen el proceso para configurar una transformación de media ponderada no reutilizable en un mapplet o asignación que utiliza transformaciones de comparación.

1. Abra un mapplet o asignación de análisis de coincidencias y añada una transformación de media ponderada descendente respecto a las transformaciones de comparación.
2. Conecte la salida de puntuación de una transformación de comparación a un puerto de entrada de media ponderada.  
Repita este paso para otras transformaciones de comparación en el mapplet o asignación.
3. Seleccione la ficha **Puertos** en la transformación de media ponderada.
4. Haga doble clic en el campo **Ponderación** para cada entrada y especifique un valor de ponderación entre 0,001 y 1. El valor de ponderación debe reflejar la importancia relativa de la puntuación de entrada cuando se compara con otras entradas en la transformación.
5. Guarde el mapplet o asignación.

## Ejemplo de puntuaciones de coincidencia ponderadas

Cree una asignación de análisis de coincidencias para determinar el número de nombres de cliente duplicados en una base de datos de clientes. Puede añadir dos transformaciones de comparación para generar puntuaciones de coincidencia para las columnas `Código postal` y `Apellido` en el conjunto de datos.

Muchos registros tienen códigos postales coincidentes, aunque un número de registros mucho menor tiene apellidos coincidentes. Cuando se promedian estas puntuaciones de coincidencia se debe enfatizar la importancia de las coincidencias más destacadas.

Para enfatizar la importancia de puntuaciones de coincidencia de apellido, puede aplicar una mayor ponderación la puntuación de coincidencia de `Apellido`.

Por ejemplo, establezca el valor de **Ponderación** de la entrada de puntuación de `Apellido` en 0,8, y establezca el valor de **Ponderación** de la entrada de puntuación de `Código postal` en 0,4.

## Propiedades avanzadas de Transformación de media ponderada

Configure las propiedades para ayudarle a determinar cómo el servicio de integración de datos procesa los datos para la Transformación de media ponderada.

Puede configurar niveles de seguimiento de registros.

Configure la siguiente propiedad en la ficha **Avanzadas**:

### Nivel de seguimiento

Cantidad de detalles que aparecen en el registro de esta transformación. Puede elegir simplificado, normal, inicialización detallada o datos detallados. El valor predeterminado es Normal.

# Delimitadores de transformación

- [Resumen de delimitadores de transformación, 382](#)

## Resumen de delimitadores de transformación

Los delimitadores de transformación especifican las divisiones entre cadenas de datos.

La siguiente tabla enumera los delimitadores que utilizan las transformaciones para analizar y escribir cadenas de datos:

Nombre de delimitador	Símbolo de delimitador
símbolo "at"	@
coma	,
guión	-
comilla doble	"
barra diagonal	/
punto final	.
hash	#
barra vertical	
punto y coma	;
comilla simple	'
espacio	[Barra espaciadora]
tabulación	[tecla Tab]
subrayado	—

# INDICE

## A

- Acción de SOAP
  - cómo reemplazar una transformación de consumidor de servicio web [348](#)
- almacenamiento de cifrado
  - Transformación de enmascaramiento de datos [139](#)
- análisis de mensajes SOAP
  - descripción [359](#)
  - elemento de unión [366](#)
  - elementos QName [365](#)
  - salida desnormalizada [362](#)
  - salida normalizada [361](#)
  - salida pivotada [363](#)
  - tipos derivados [364](#)
- anyType
  - asignar puertos [377](#)
- archivo de rechazo
  - estrategias de actualización [334](#)
- Archivo WSDL
  - elemento de enlace [338](#)
  - elemento de operación [338](#)
  - elemento de puerto [338](#)
  - elemento de servicio [338](#)
- asignación de entrada
  - Transformación de consumidor de servicio web [342](#)
- asignación de salida
  - Transformación de consumidor de servicio web [344](#)
- asignaciones
  - cómo marcar filas para actualizar [334](#)
  - cómo utilizar transformaciones de enrutador [303](#)
- asignaciones con fallo
  - transformaciones de Java [191](#)
- asignar puertos
  - problemas de calidad [62](#)
- atributos anyAttribute
  - Transformación de consumidor de servicio web [342](#), [344](#)
- autenticación por cookies
  - Transformación de consumidor de servicio web [341](#)

## B

- búsquedas conectadas
  - descripción [246](#)
  - resumen [247](#)
- búsquedas no conectadas
  - descripción [246](#)
  - resumen [247](#)

## C

- cadenas
  - cómo crear rangos [296](#)

- caracteres de cadena de origen
  - Transformación de enmascaramiento de datos [129](#)
- caracteres de reemplazo de cadena de resultado
  - Transformación de enmascaramiento de datos [130](#)
- clase EDatatype
  - expresiones de Java [206](#)
- clase JExpression
  - expresiones de Java [208](#), [209](#)
- clase JExprParaMetadata
  - expresiones de Java [206](#)
- classpath
  - propiedad de asignación [175](#)
- clave de cifrado de almacenamiento
  - Transformación de enmascaramiento de datos [139](#)
- claves
  - jerarquía de mensajes SOAP [370](#)
- claves compuestas
  - Transformación de consumidor de servicio web [342](#)
- claves generadas
  - grupos de salida de servicio web [361](#)
- clústeres
  - Transformación de excepción de registros duplicados [151](#)
- código estático
  - transformaciones de Java [179](#)
- código Java
  - cómo encontrar errores [188](#)
  - en transformaciones de Java [176](#)
- coincidencia de campos
  - Transformación de coincidencia [260](#)
- coincidencia de identidades
  - Transformación de coincidencia [260](#)
- columna de dependientes
  - enmascaramiento de datos [125](#)
- Columna Ubicación
  - transformación de servicio web [369](#)
- cómo compilar
  - transformaciones de Java [187](#)
- cómo compilar código Java
  - ficha Código completo [182](#)
- cómo crear
  - transformaciones de Java [185](#), [186](#)
- cómo crear rangos
  - grupos de datos [298](#)
  - valores de cadena [296](#)
- cómo diseñar
  - transformaciones de Java [171](#)
- cómo encontrar
  - errores en código Java [188](#)
- Cómo reenviar filas rechazadas
  - configuración [334](#)
  - opción [334](#)
- compresión SOAP
  - Transformación de consumidor de servicio web [351](#)
- condición de búsqueda
  - Transformación de enmascaramiento de datos [124](#)

- condición de unión
  - resumen [215](#)
- condición del filtro de grupo
  - Transformación de enrutador [301](#)
- condiciones
  - Transformación de enrutador [301](#)
  - Transformación de incorporación [215](#)
- conexión
  - servicios web [348](#)
- conexiones de servicios web
  - resumen [348](#)
- consideraciones
  - transformaciones de Java [171](#)
- consolidación automática
  - Transformación de excepción de registros duplicados [152](#)
- consolidación manual
  - Transformación de excepción de registros duplicados [152](#)
- consulta SQL
  - transformación de SQL [316](#)
- Continuar ante error de SQL
  - transformación de SQL [317](#), [321](#)

## D

- Datos del clúster
  - grupo de salida [155](#)
- datos pivotados
  - mensajes SOAP [373](#)
- defineJExpression
  - método API de expresión de Java [207](#)
- desenfoque
  - valores de fecha [132](#)
  - valores numéricos [131](#)
- diccionario
  - enmascaramiento de expresiones repetible [118](#)
  - máscara de datos de sustitución [122](#)
- dígito de inicio
  - Números de la seguridad social [136](#)
- Directorio de archivo caché
  - Transformación de excepción de registros duplicados [156](#)
- directorio de la memoria caché
  - Transformación de enmascaramiento de datos [139](#)

## E

- Editor de expresiones
  - cómo validar expresiones [9](#)
  - Descripción [8](#)
- efectos secundarios
  - Transformación de consumidor de servicio web [352](#)
  - transformación de SQL [321](#)
- elemento de unión
  - cómo analizar mensajes SOAP [366](#)
- elementos
  - unión [379](#)
- elementos any
  - Transformación de consumidor de servicio web [342](#), [344](#)
- elementos anyType
  - cómo analizar [363](#)
  - Transformación de consumidor de servicio web [342](#), [344](#)
- elementos de lista
  - cómo analizar mensajes SOAP [366](#)
  - descripción [379](#)
- elementos de selección
  - cómo analizar mensajes SOAP [366](#)

- elementos de selección (*continuado*)
  - cómo se visualizan en una transformación de consumidor de servicio web [343](#), [346](#)
  - descripción [378](#)
- elementos de tipo derivado
  - Transformación de consumidor de servicio web [342](#), [344](#)
- elementos de unión
  - descripción [379](#)
- elementos QName
  - análisis de mensajes SOAP [365](#)
- encabezado HTTP
  - cómo añadir a una transformación de consumidor de servicio web [340](#)
- enlace
  - Elemento de archivo WSDL [338](#)
- enlace de parámetro
  - transformación de SQL [316](#)
- enmascaramiento aleatorio
  - enmascaramiento de valores de cadena [116](#)
  - enmascaramiento de valores de fecha [117](#)
  - valores numéricos [116](#)
- enmascaramiento de claves
  - descripción [120](#)
  - enmascaramiento de valores de cadena [120](#)
  - enmascaramiento de valores de fecha y hora [121](#)
  - enmascaramiento de valores numéricos [121](#)
  - valores numéricos [120](#)
- enmascaramiento de expresiones
  - descripción [118](#)
  - ejemplo de enmascaramiento repetible [119](#)
  - enmascaramiento repetible [118](#)
  - reglas y directrices [120](#)
- enmascaramiento de formato especial
  - direcciones de correo electrónico [133](#)
  - direcciones IP [134](#)
  - Direcciones URL [136](#)
  - Números de la seguridad social [135](#), [136](#)
  - números de SIN repetibles [136](#)
  - números de tarjeta de crédito [132](#)
  - números de teléfono [135](#)
- enmascaramiento dependiente
  - descripción [125](#)
- entrada desnormalizada
  - puertos de servicio web [375](#)
- errores
  - cómo incrementar el umbral en transformaciones de Java [193](#)
- errores ajenos al código de usuario
  - en transformaciones de Java [188](#)
- errores de código de usuario
  - transformaciones de Java [188](#)
- errores de compilación
  - cómo identificar el origen de errores en transformaciones de Java [188](#)
- estrategia de coincidencia
  - Transformación de excepción de registros duplicados [158](#)
- expresiones
  - cómo añadir a un puerto [8](#)
  - cómo añadir comentarios [9](#)
  - cómo validar [9](#)
  - en transformaciones [6](#)
  - transformaciones de Java [200](#)
- Expresiones
  - cómo introducir [8](#)
- expresiones de Java
  - clase EDataType [206](#)
  - clase JExpression [208](#), [209](#)
  - clase JExprParaMetadata [206](#)
  - cómo configurar [201](#)

- expresiones de Java (*continuado*)
  - cómo configurar funciones [202](#)
  - cómo crear [202](#)
  - cómo crear en el cuadro de diálogo Definir función [202](#)
  - cómo generar [201](#)
  - cómo invocar [194](#)
  - cómo invocar con la interfaz avanzada [205](#)
  - cómo invocar con la interfaz simple [204](#)
  - ejemplo de interfaz avanzada [208](#)
  - ejemplo de interfaz simple [204](#)
  - generar código Java [202](#)
  - interfaz avanzada [205](#)
  - interfaz simple [203](#)
  - método API invokeJExpression [204](#)
  - reglas y directrices [204](#), [205](#)
  - reglas y directrices para invocar [194](#)
  - transformaciones de Java [200](#)

## F

- fallos SOAP genéricos
  - Transformación de consumidor de servicio web [350](#)
- ficha AI final
  - transformaciones de Java [181](#)
- ficha Aplicaciones auxiliares
  - transformaciones de Java [179](#), [180](#)
- ficha Código completo
  - errores de compilación de Java [187](#)
  - transformaciones de Java [182](#)
- ficha En entrada
  - transformaciones de Java [181](#)
- ficha Funciones
  - transformaciones de Java [182](#)
- ficha Importaciones
  - transformaciones de Java [179](#), [180](#)
- filas
  - cómo marcar para actualizar [334](#)
- filas de entrada
  - cómo obtener el tipo de fila para [192](#)
- formato de máscara
  - enmascaramiento de valores de cadena [128](#)
  - formatos de máscara especiales [132](#)

## G

- generar código Java
  - expresiones de Java [202](#)
- generar fila de salida
  - transformaciones de Java [192](#)
- generar tabla de registros incorrectos
  - Transformación de excepción de registros incorrectos [61](#)
- Grupo de salida estándar
  - Transformación de excepción de registros duplicados [155](#)
- grupo definido por el usuario
  - Transformación de enrutador [300](#)
- grupo todo
  - cómo se visualizan en una transformación de consumidor de servicio web [343](#)
- grupos
  - cómo añadir a la transformación de enrutador [302](#)
  - definido por el usuario [300](#)
  - Transformación de enrutador [300](#)
- grupos de salida
  - Transformación de excepción de registros incorrectos [54](#)
- grupos de sustitución
  - cómo analizar mensajes SOAP [365](#)

- grupos de sustitución (*continuado*)
  - servicios web [377](#)
  - Transformación de consumidor de servicio web [342](#), [344](#)
- grupos predeterminados
  - Transformación de enrutador [300](#)
- GZip
  - cómo comprimir mensajes SOAP [351](#)

## I

- información del diccionario
  - Transformación de enmascaramiento de datos [124](#)
- Informatica Data Director para Data Quality
  - excepciones de registros incorrectos [57](#)
- interfaz avanzada
  - clase EDataType [206](#)
  - clase JExpression [208](#), [209](#)
  - clase JExprParaMetadata [206](#)
  - cómo invocar expresiones de Java [205](#)
  - ejemplo [208](#)
  - expresiones de Java [205](#)
- interfaz simple
  - ejemplo [204](#)
  - expresiones de Java [203](#)
  - Métodos API de transformación de Java [203](#)
- intervalo
  - enmascaramiento de valores numéricos [131](#)
- intervalo de confirmación de almacenamiento
  - Transformación de enmascaramiento de datos [139](#)
- invoke
  - método API de expresión de Java [211](#)
- invokeJExpression
  - método API [204](#)

## J

- JDK
  - transformación de Java [168](#)
- jerarquía SOAP
  - relación con puertos de entrada [369](#)
- JRE
  - transformación de Java [168](#)

## M

- máscara de sustitución
  - descripción [122](#)
  - propiedades de enmascaramiento [123](#)
- memoria caché
  - Transformación de búsqueda [250](#)
- memoria caché de búsqueda
  - resumen [250](#)
- mensaje SOAP
  - claves [370](#)
- mensajes de solicitud de servicio web simultáneos
  - cómo habilitar una transformación de consumidor de servicio web [348](#)
- mensajes SOAP
  - asignar elementos de lista [379](#)
  - asignar elementos de selección [378](#)
  - asignar nodos de ocurrencia múltiple [361](#)
  - asignar puertos [372](#)
  - asignar puertos a elementos de unión [379](#)
  - asignar varios puertos de entrada [373](#)
  - cómo analizar elementos anyType [363](#)

- mensajes SOAP (*continuado*)
  - cómo analizar elementos de lista [366](#)
  - cómo analizar elementos de selección [366](#)
  - cómo analizar grupos de sustitución [365](#)
  - cómo pivotar datos [373](#)
  - resumen [338](#)
- Método API de expresión de Java
  - getResultDataType [210](#)
- método defineJExpression
  - transformaciones de Java [190](#)
- método failSession
  - transformaciones de Java [191](#)
- método generateRow
  - transformaciones de Java [192](#)
- método getBytes
  - transformaciones de Java [209](#)
- método getDouble
  - transformaciones de Java [210](#)
- método getInRowType
  - transformaciones de Java [192](#)
- método getInt
  - transformaciones de Java [210](#)
- método getLong
  - transformaciones de Java [210](#)
- Método getMetada
  - transformaciones de Java [193](#)
- método getResultDataType
  - Transformaciones de Java [210](#)
- método getResultMetadata
  - transformaciones de Java [210](#)
- método getStringBuffer
  - transformaciones de Java [211](#)
- método incrementErrorCount
  - transformaciones de Java [193](#)
- método invokeJExpression
  - transformaciones de Java [194](#)
- método isNull
  - transformaciones de Java [195](#)
- método isResultNull
  - Transformaciones de Java [211](#)
- método logError
  - transformaciones de Java [196](#)
- método logInfo
  - transformaciones de Java [196](#)
- método resetNotification
  - transformaciones de Java [197](#)
- método setNull
  - transformaciones de Java [197](#)
- Método storeMetada
  - transformaciones de Java [198](#)
- métodos
  - API de transformación de Java [189](#)
- métodos API
  - transformación de Java [189](#)
  - transformaciones de Java [189](#)
- métodos API de expresión de Java
  - defineJExpression [207](#)
  - getDouble [210](#)
  - getInt [210](#)
  - getLong [210](#)
  - getResultMetadata [210](#)
  - getStringBuffer [211](#)
  - invoke [211](#)
  - isResultNull [211](#)
- Métodos API de expresión de Java
  - getBytes [209](#)
- métodos definidos por el usuario
  - transformaciones de Java [179](#)

## N

- nombre de usuario de seguridad WS
  - puerto dinámico [341](#)
- Número de filas afectadas
  - transformación de SQL [315](#)
- Números de la seguridad social
  - enmascaramiento de código de área [135](#)
  - enmascaramiento de datos repetibles [135](#)
- Números de SIN
  - enmascaramiento de datos repetibles [136](#)
  - enmascaramiento de números de la seguridad social [136](#)
- NumRowsAffected
  - salida de filas [319](#)

## O

- operación
  - Elemento de archivo WSDL [338](#)
- optimización de inserción
  - habilitar en la transformación de SQL [322](#)
  - Transformación de consumidor de servicio web [352](#)
  - transformación de SQL [322](#)
- optimización de primera selección
  - Transformación de consumidor de servicio web [352](#)
  - transformación de SQL [321](#)

## P

- paquetes Java
  - cómo importar [179](#)
- problemas de calidad
  - asignar puertos a [62](#)
- procesamiento al nanosegundo
  - transformaciones de Java [173](#)
- procesamiento de alta precisión
  - transformaciones de Java [173](#)
- propiedad de ordenación
  - Transformación de excepción de registros duplicados [156](#)
- propiedades avanzadas
  - Transformación de consumidor de servicio web [348](#)
  - Transformación de excepción de registros duplicados [156](#)
  - transformaciones de Java [173](#)
- Propiedades avanzadas
  - Transformación de excepción de registros incorrectos [62](#)
  - transformación de SQL [312](#)
- puerto de filtro
  - Transformación de consumidor de servicio web [352](#)
- Puerto de salida estándar
  - descripción [59](#)
- puerto SQLException
  - transformación de SQL [315](#)
- puertos
  - asignar a mensajes SOAP [372](#)
  - entrada de servicio web desnormalizado [375](#)
  - Transformación de enrutador [303](#)
  - Transformación de excepción de registros duplicados [154](#)
  - Transformación de excepción de registros incorrectos [58](#)
  - transformaciones de Java [172](#)
- puertos de entrada
  - transformaciones de Java [172](#)
- puertos de entrada de SQL
  - transformación de SQL [313](#)
- puertos de problema de calidad
  - En blanco o nulo [54](#)

- puertos de problemas de calidad
  - descripción [58](#)
- puertos de salida
  - transformación de Java [172](#)
  - transformaciones de Java [172](#)
- puertos de transferencia
  - transformación de SQL [314](#)

## R

- Recuento máximo de filas de salida
  - transformación de SQL [317](#), [319](#)
- recuentos de errores
  - cómo incrementar para transformaciones Java [193](#)
- reemplazar acción de SOAP
  - Transformación de consumidor de servicio web [348](#)
- registros
  - transformaciones de Java [196](#)
- registros rechazados
  - Transformación de excepción de registros incorrectos [59](#)
- reglas de enmascaramiento
  - caracteres de cadena de origen [129](#)
  - caracteres de reemplazo de cadena de resultado [130](#)
  - desenfoque [131](#)
  - formato de máscara [128](#)
  - formatos de máscara especiales [132](#)
  - intervalo [131](#)

## S

- salida de error HTTP
  - cómo habilitar una transformación de consumidor de servicio web [348](#)
- salida de fallos genéricos
  - cómo habilitar una transformación de consumidor de servicio web [348](#)
- salida desnormalizada
  - análisis de mensajes SOAP [362](#)
- salida pivotada
  - análisis de mensajes SOAP [363](#)
- salida única
  - Transformación de enmascaramiento de datos [124](#)
- seguridad de la capa de transporte
  - Transformación de consumidor de servicio web [339](#)
- servicio
  - Elemento de archivo WSDL [338](#)
- Servicio de integración de datos)
  - modo de reinicio [197](#)
- servicio web
  - asignar puertos a anyTypes [377](#)
  - grupos de sustitución [377](#)
  - tipos derivados [376](#)
- Sin estado
  - transformaciones de Java [173](#)
- snippets de código
  - cómo crear para transformaciones de Java [177](#)
- snippets de código Java
  - cómo crear para transformaciones de Java [177](#)
- solución de problemas
  - transformaciones de Java [187](#)
- sustitución de cadenas
  - transformación de SQL [317](#)

## T

- tabla de almacenamiento
  - enmascaramiento de expresiones [119](#)
  - máscara de datos de sustitución [123](#)
- tabla de almacenamiento compartido
  - Transformación de enmascaramiento de datos [139](#)
- Tabla de problemas
  - cómo generar [61](#)
- tabla de registros únicos
  - cómo crear [163](#)
- Tamaño de archivo caché
  - Transformación de excepción de registros duplicados [156](#)
- tamaño de la memoria caché
  - Transformación de enmascaramiento de datos [139](#)
- Tareas humanas
  - con excepciones de registros incorrectos [57](#)
- técnicas de enmascaramiento
  - enmascaramiento de datos [116](#)
- tipo de condición
  - propiedades avanzadas de la transformación de incorporación [216](#)
  - transformación de incorporación simple [215](#)
- tipo de condición avanzada
  - Transformación de incorporación [216](#)
- tipo de condición simple
  - Transformación de incorporación [215](#)
- tipos de datos
  - transformaciones de Java [170](#)
- tipos de datos de Java primitivos
  - transformaciones de Java [170](#)
- tipos derivados
  - cómo analizar mensajes SOAP [364](#)
  - servicios web [376](#)
- todos los grupos
  - cómo se visualizan en una transformación de consumidor de servicio web [346](#)
- transformación de agregación
  - cómo crear una transformación reutilizable [79](#)
  - cómo ordenar datos [78](#)
  - entrada ordenada [77](#)
  - expresiones de agregado [74](#)
  - expresiones de no agregado [77](#)
  - funciones de agregado [75](#)
  - funciones de agregado anidadas [75](#)
  - memorias cachés de agregado [74](#)
  - puertos [73](#)
  - puertos agrupar por [76](#)
  - resumen [72](#)
- Transformación de agregación
  - Combinación de estrategia de actualización [335](#)
  - cómo crear una transformación no reutilizable [79](#)
  - cómo desarrollar [73](#)
  - consejos [80](#)
  - propiedades avanzadas [73](#)
  - solución de problemas [80](#)
- Transformación de analizador
  - resumen [282](#)
- Transformación de asociación
  - resumen [69](#)
- transformación de búsqueda
  - cómo crear búsquedas no conectadas [256](#)
  - cómo crear una transformación no reutilizable [255](#)
  - cómo crear una transformación reutilizable [254](#)
  - cómo desarrollar [248](#)
  - condición de búsqueda [248](#)
  - conectada [246](#), [247](#)
  - no conectada [246](#)
  - no conectadas [247](#)

transformación de búsqueda (*continuado*)  
 propiedades avanzadas para búsquedas de objetos de datos lógicos [254](#)  
 propiedades avanzadas para búsquedas en la tabla de referencia [254](#)  
 propiedades avanzadas para búsquedas relacionales [254](#)  
 propiedades avanzadas para transformaciones de búsqueda de archivo sin formato [253](#)  
 propiedades del tiempo de ejecución [251](#)  
 reglas y directrices para la condición de búsqueda [249](#)  
 resumen [245](#)

Transformación de búsqueda  
 almacenar en memoria caché [250](#)

Transformación de clasificador  
 algoritmo clasificador [85](#)  
 modelo de clasificador [85](#)  
 resumen [84](#)

Transformación de coincidencia  
 coincidencia de campos [260](#)  
 coincidencia de identidades [260](#)  
 ejemplo de estrategia de coincidencia [158](#)  
 resumen [259](#)

Transformación de comparación  
 resumen [93](#)

Transformación de consolidación  
 resumen [98](#)

Transformación de consumidor de servicio web  
 asignación de entrada [342](#)  
 asignación de salida [344](#)  
 autenticación por cookies [341](#)  
 cómo añadir encabezados HTTP [340](#)  
 cómo asignar nodos de salida [344](#)  
 cómo asignar puertos de entrada [342](#)  
 cómo crear [354](#)  
 cómo habilitar la salida de error HTTP [348](#)  
 cómo habilitar la salida de fallos genéricos [348](#)  
 cómo visualizar claves [343](#), [346](#)  
 compresión SOAP [351](#)  
 Control de errores [350](#)  
 fallos SOAP genéricos [350](#)  
 habilitar la optimización de inserción [353](#)  
 mensajes de solicitud de servicio web simultáneos [348](#)  
 mensajes SOAP [338](#)  
 nombre de seguridad WS dinámica [341](#)  
 operaciones [339](#)  
 optimización de inserción [352](#)  
 optimización de primera selección [352](#)  
 optimización del filtro [352](#)  
 propiedades avanzadas [348](#)  
 resumen [337](#)  
 seguridad [339](#)  
 seguridad de la capa de transporte [339](#)  
 URL de punto final [341](#)  
 URL dinámica de servicio web [341](#)

Transformación de conversión de mayúsculas y minúsculas  
 resumen [81](#)

Transformación de decisión  
 estrategia de ejemplo de problemas de calidad [57](#)  
 resumen [144](#)

transformación de enmascaramiento de datos  
 enmascaramiento de direcciones IP [134](#)

Transformación de enmascaramiento de datos  
 archivo de valores predeterminados [136](#)  
 caracteres de cadena de origen [129](#)  
 descripción [115](#)  
 desenfoque [131](#)  
 diccionario para la máscara de sustitución [122](#)  
 directorio de la memoria caché [139](#)

Transformación de enmascaramiento de datos (*continuado*)  
 directrices de enmascaramiento de expresiones [120](#)  
 enmascaramiento aleatorio [116](#)  
 enmascaramiento de datos dependiente [125](#)  
 enmascaramiento de direcciones de correo electrónico [133](#)  
 enmascaramiento de direcciones URL [136](#)  
 enmascaramiento de expresiones [118](#)  
 enmascaramiento de expresiones del nombre del diccionario [118](#)  
 enmascaramiento de expresiones repetible [118](#)  
 enmascaramiento de números de la seguridad social [135](#), [136](#), [139](#)  
 enmascaramiento de números de teléfono [135](#)  
 enmascaramiento de tarjetas de crédito [132](#)  
 enmascaramiento de valores de fecha [131](#)  
 formato de máscara [128](#)  
 formatos de máscara especiales [132](#)  
 intervalo [131](#)  
 intervalo de confirmación de almacenamiento [139](#)  
 máscara de sustitución [122](#)  
 números de SIN repetibles [136](#)  
 propiedades de la máscara de sustitución [123](#), [124](#)  
 propiedades de la sesión [139](#)  
 salida única [139](#)  
 SSN repetibles [135](#)  
 tabla de almacenamiento [119](#)  
 tabla de almacenamiento compartido [139](#)  
 tablas de almacenamiento [123](#)  
 tamaño de la memoria caché [139](#)  
 técnicas de enmascaramiento [116](#)  
 Transformación de enmascaramiento de datos [139](#)

Transformación de enrutador  
 cómo conectar en asignaciones [303](#)  
 condición del filtro de grupo [301](#)  
 ejemplo [301](#)  
 grupos [300](#)  
 propiedades avanzadas [303](#)  
 puertos [303](#)  
 resumen [300](#)

Transformación de estandarizador  
 resumen [327](#)

Transformación de estrategia de actualización  
 Combinación de agregación [335](#)  
 cómo crear [334](#)  
 cómo reenviar filas rechazadas [334](#)  
 expresiones [334](#)  
 pasos para la configuración [333](#)  
 propiedades avanzadas [334](#)  
 resumen [333](#)

Transformación de etiquetador  
 resumen [231](#)

Transformación de excepción  
 Vista Asignación problemática [61](#)

Transformación de excepción de registros duplicados  
 clústeres [151](#)  
 cómo generar una tabla de registros duplicados [153](#)  
 cómo ordenar registros [156](#)  
 configurar [163](#)  
 consolidación automática [152](#)  
 ejemplo [157](#)  
 ejemplo de asignación [157](#)  
 ejemplo de salida [160](#)  
 ejemplo de salida del clúster [161](#)  
 ejemplo de valores de configuración [159](#)  
 Grupo de salida estándar [155](#)  
 grupos de salida [155](#)  
 propiedades avanzadas [156](#)  
 puertos [154](#)  
 resumen [150](#)  
 Vista de configuración [152](#)

## Transformación de excepción de registros incorrectos

- asignación [56](#)
- configurar [63](#)
- ejemplo [64](#)
- ejemplo de salida de problemas [67](#)
- ejemplo de salida de registros incorrectos [66](#)
- flujo de proceso [55](#)
- grupos de puertos [58](#)
- grupos de salida [54](#)
- mapplet de ejemplo [64](#)
- problemas de calidad [57](#)
- propiedades avanzadas [62](#)
- puertos de entrada [58](#)
- puertos de problema de calidad en blanco [54](#)
- puertos de salida [59](#)
- resumen [53](#)
- salida de ejemplo [66](#)
- salida de ejemplo de registros correctos [68](#)
- Vista de configuración [59](#)

## transformación de expresión

- propiedades avanzadas [165](#)
- puertos [165](#)
- resumen [164](#)

## Transformación de expresión

- cómo desarrollar [164](#)

## transformación de filtro

- condición de filtro [167](#)
- propiedades avanzadas [166](#)
- resumen [166](#)

## Transformación de filtro

- consejos sobre el rendimiento [167](#)
- filas con valores nulos [167](#)

## Transformación de fusión

- resumen [280](#)

## Transformación de generador de claves

- resumen [226](#)

## transformación de incorporación

- cómo configurar el orden de clasificación [219](#)
- entrada ordenada [219](#)
- propiedades avanzadas [214](#)
- puertos [214](#)
- rendimiento [224](#)
- resumen [213](#)
- unión normal [216](#)

## Transformación de incorporación

- cómo bloquear canales [223](#)
- cómo guardar filas maestras en la memoria caché [224](#)
- cómo unir datos de un mismo origen [221](#)
- condiciones [215](#)
- no ordenada [223](#)
- ordenada [224](#)
- outer join completo [218](#)
- outer join de detalles [218](#)
- reglas y directrices [225](#)
- resumen [330](#)
- tipo de condición [215](#), [216](#)
- tipo de condición avanzada [216](#)
- tipo de condición simple [215](#)
- tipos de unión [216](#)

## transformación de Java

- métodos API [189](#)
- resumen [168](#)

## Transformación de ordenación

- resumen [305](#)

## Transformación de rango

- cómo definir grupos para [298](#)
- opciones [296](#)
- propiedades avanzadas [298](#)

## Transformación de rango (*continuado*)

- puerto de rango [297](#)
- Puerto RANKINDEX [297](#)
- puertos [296](#)
- resumen [295](#)

## transformación de SQL

- cardinalidad entre fila de entrada y fila de salida [317](#)
- cómo definir la conexión de la base de datos [326](#)
- cómo definir la consulta [316](#)
- cómo definir puertos de salida [314](#)
- cómo restringir filas de salida [319](#)
- continuar ante error de SQL [321](#)
- descripción de puertos de entrada [313](#)
- ejemplo [323](#)
- enlace de parámetro [316](#)
- instrucción de consulta [325](#)
- número de filas afectadas [315](#)
- número de salida de filas [319](#)
- optimización de inserción [322](#)
- optimización de primera selección [321](#)
- propiedades de la optimización de inserción [322](#)
- puerto SQLException [315](#)
- puertos [313](#)
- puertos de transferencia [314](#)
- sustitución de cadenas en consultas [317](#)
- Vista Propiedades avanzadas [312](#)

## transformación de unión

- outer join principal [217](#)

## Transformación SQL

- resumen [311](#)

## transformaciones

- activas [1](#)
- cómo crear [9](#)
- cómo desarrollar [5](#)
- cómo editar reutilizables [6](#)
- conectadas [2](#)
- expresiones [6](#)
- Java [168](#)
- no conectadas [2](#)
- pasivas [2](#)
- resumen [1](#)
- reutilizables [5](#)
- validación de expresiones [9](#)

## transformaciones activas

- descripción [1](#)
- Java [168](#), [170](#)
- Rango [295](#)

## transformaciones conectadas

- Java [168](#)
- Rango [295](#)

## transformaciones de Java

- activas [170](#)
- almacenar metadatos [198](#)
- Ámbito de transformación [173](#)
- asignaciones con fallo en [191](#)
- classpath a nivel de asignación [175](#)
- código Java [176](#)
- cómo compilar [187](#)
- cómo comprobar valores nulos en [195](#)
- cómo crear [185](#), [186](#)
- cómo crear puertos [172](#)
- cómo crear snippets de código Java [177](#)
- cómo definir valores nulos en [197](#)
- cómo diseñar [171](#)
- cómo identificar el origen de errores de compilación [188](#)
- cómo obtener el tipo de fila de entrada [192](#)
- cómo restablecer variables en [197](#)
- conversión de tipos de datos [170](#)

## transformaciones de Java (*continuado*)

- errores ajenos al código de usuario [188](#)
- errores de código de usuario [188](#)
- errores de compilación [187](#)
- ficha Al final [181](#)
- ficha Aplicaciones auxiliares [179](#), [180](#)
- ficha Código completo [182](#)
- ficha En entrada [181](#)
- ficha Funciones [182](#)
- ficha Importaciones [179](#), [180](#)
- método defineJExpression [190](#)
- método failSession [191](#)
- método generateRow [192](#)
- método getInRowType [192](#)
- Método getMetadata [193](#)
- método incrementErrorCount [193](#)
- método invokeJExpression [194](#)
- método isNull [195](#)
- método logError [196](#)
- método logInfo [196](#)
- método resetNotification [197](#)
- método setNull [197](#)
- Método storeMetadata [198](#)
- métodos API [189](#)
- no reutilizables [169](#)
- pasivas [170](#)
- procesamiento al nanosegundo [173](#)
- procesamiento de alta precisión [173](#)
- propiedades avanzadas [173](#)
- puertos de entrada [172](#)
- puertos de salida [172](#)
- recuperando metadatos [193](#)
- registros [196](#)
- reutilizables [169](#)
- Sin estado [173](#)
- solución de problemas [187](#)
- tipos de datos de Java primitivos [170](#)
- valores de puerto predeterminados [172](#)
- transformaciones de servicio web
  - Columna Ubicación [369](#)
- transformaciones no conectadas
  - transformación de búsqueda [247](#)
- transformaciones pasivas
  - descripción [2](#)
  - Java [168](#), [170](#)
- transformaciones reutilizables
  - cómo editar [6](#)
  - descripción [5](#)
- tratar fallo como error
  - cómo habilitar una transformación de consumidor de servicio web [348](#)

## U

- umbral inferior
  - configurar [59](#), [152](#)
- umbral superior
  - configurar [59](#), [152](#)
- URL de punto final
  - Transformación de consumidor de servicio web [341](#)
- URL dinámica
  - Transformación de consumidor de servicio web [341](#)

## V

- valores de cadena
  - enmascaramiento de datos de claves [120](#)
  - enmascaramiento de datos personalizados [116](#)
- valores de fecha
  - enmascaramiento de datos aleatorio [117](#)
- valores de fecha y hora
  - enmascaramiento de datos [121](#)
- valores de puerto
  - transformaciones de Java [172](#)
- valores nulos
  - cómo comprobar en transformaciones de Java [195](#)
  - cómo definir para transformaciones Java [197](#)
- valores numéricos
  - enmascaramiento aleatorio [117](#)
  - enmascaramiento de claves [121](#)
- valores predeterminados
  - enmascaramiento de datos [136](#)
- variables
  - transformaciones de Java [179](#)
- variables de instancia
  - transformaciones de Java [179](#)
- variables estáticas
  - transformaciones de Java [179](#)
- Vista Asignación problemática
  - Transformación de excepción de registros incorrectos [61](#)
- Vista de configuración
  - Ejemplo de transformación de excepción de registros incorrectos [65](#)
  - Transformación de excepción de registros duplicados [152](#)
  - Transformación de excepción de registros incorrectos [59](#)
- vista Puertos
  - salida de transformación SQL [314](#)
- Vista Tiempo de ejecución
  - transformación de SQL [326](#)
- vistas del editor
  - transformaciones de Java no reutilizables [169](#)
  - transformaciones de Java reutilizables [169](#)