



DocuSign Connect Guide

Copyright ©2003-2016 DocuSign, Inc. All rights reserved.

For information about DocuSign trademarks, copyrights and patents refer to the [DocuSign Intellectual Property page](https://www.docusign.com/IP) (<https://www.docusign.com/IP>) on the DocuSign website. All other trademarks and registered trademarks are the property of their respective holders.

No part of this document may be reproduced or transmitted in any form or by any means, electronic or mechanical, for any purpose, without the express written permission of DocuSign, Inc. Under the law, reproducing includes translating into another language or format. Every effort has been made to ensure that the information in this manual is accurate. DocuSign, Inc. is not responsible for printing or clerical errors. Information in this document is subject to change without notice.

DocuSign Connect Guide March 2016

If you have any comments or feedback on our documentation, please send them to us at: Documentation@DocuSign.com.

Summary of changes for this version:

- Updated the **Require Acknowledgement** setting description by removing the statement about requiring the DocuSign certificate and added that DocuSign logs a failure if the attempt to reach the external endpoint returns anything other than an HTTP 200.
- Removed note that the **Include Envelope Voided Reason** was not applicable if the Use Soap Interface was selected.
- Updated TLS information to say that DocuSign uses TLS v1.2 by default for all Connect messages, but will support earlier versions of TLS if required by the listener.

Table of Contents

DocuSign Connect	4
Introduction	4
Setup	5
Setting Up Connect in Your Account	5
XML Information Structure	9
Key Transaction Elements.....	10
Form Field Data	11
Technical Details	11
Running the Service	11
Best Practices.....	11
Recommendations for Receiving Documents	12
Connect SOAP Publishing.....	12
Publish Failures	13
<i>Manually Publishing Transactions</i>	13
<i>Sample API Calls and Schema:</i>	13
Sample Connect Message	15

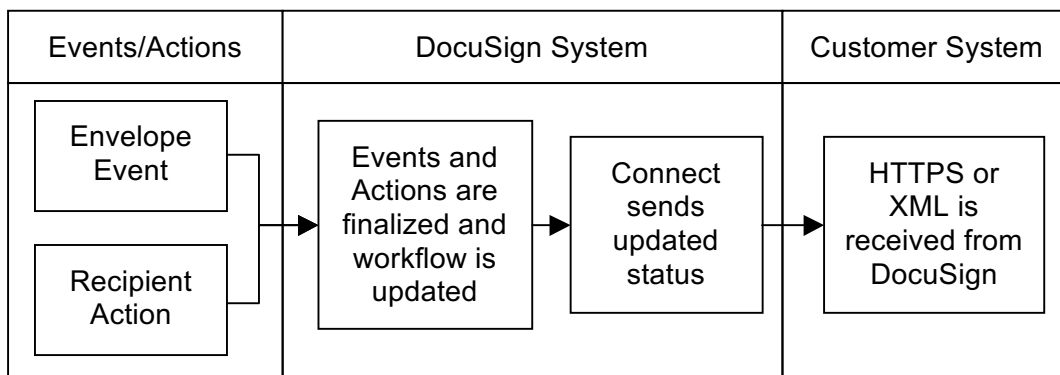
DocuSign Connect

This guide provides a developer or business analyst with information about how DocuSign® Connect, sometimes referred to as the publisher in this document, works and discusses the data elements that are available.

Introduction

DocuSign Connect is a push service that sends real-time envelope and recipient data updates to customer listener applications. These updates are generated by changes to the envelope as it progresses from sending to completion. Connect provides updated information about the status of these transactions, including the actual content of document form fields. Connect is useful to organizations that want a real-time view into the transactions across their user base in a centralized location. This information can be customized to drive reporting or workflow specific to that organization's needs. Customers can create multiple Connect configurations, each with different events or users, and set up different listeners to monitor those configurations.

DocuSign Connect watches user envelopes, noting when transactions reach the subscribed event triggers. At that point, Connect sends an XML status change to the customer's listener. The general flow of events is outlined below.



An organization's external application can use a secure (HTTPS) 'listener' or an API interface to accept information from Connect.

- The 'listener' is an application that accepts XML transactions sent from DocuSign Connect as events happen. This interface is not a SOAP API, such as the other interfaces in the DocuSign System, instead the messages are sent through an HTTP POST.
- The API interface uses DocuSign's SOAP or REST API, like the other interfaces in the DocuSign System, to receive XML information posts.

Note: In cases where envelopes are sent through the DocuSign API with event notification enabled and Connect is also used, when the events for both the API and Connect are set to be the same, then two separate notifications are sent for the envelope. One from the API and another from Connect.

Setup

In order to use DocuSign Connect, Connect must be enabled in your DocuSign account. It is not enabled by default. Once enabled, the Connect Settings page can be accessed from the Classic DocuSign Experience web application.

At a high level, the following steps must be taken to use Connect:

- 1) Request your account be configured to publish transaction updates. Your DocuSign Account Manager can help you with this step.
- 2) Select the events you will use as triggers for updating the information. Events can include several items such as document sent, viewed, signed, completed, etc.

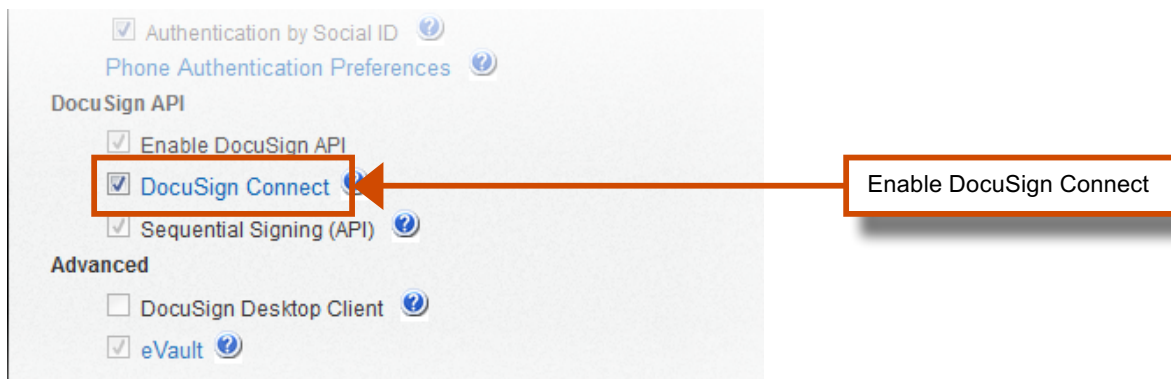
Note: You can create multiple Connect configurations with different event triggers and users.

- 3) Develop an understanding of the XML data sent from Connect to your application.
- 4) Determine how your application will accept information from Connect; by secure listener or by an API interface.
 - If using a secure listener: Create an application at your HTTPS location that can accept data, parse the inbound XML data and utilize it. This is a web application written specifically for your business.
 - If using an API interface: Create a Retrieving Service endpoint (URL), method name, method's argument name, and Namespace. Refer to the [Connect SOAP Publishing section](#) for more information about using the SOAP interface. Inclusion of the DocuSign X.509 certificate in the SOAP header is optional.

Setting Up Connect in Your Account

Note: This procedure describes setting up a custom Connect configuration in the Classic DocuSign Experience web application. If you are using the New Administrator Experience, please refer to the [New Administrator guide](#) for information on setting up Connect configurations.

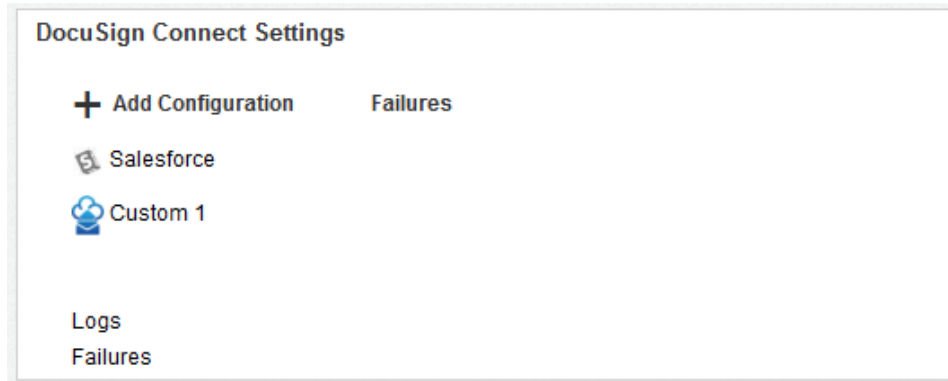
1. In order to access the DocuSign Connect page, log on to your DocuSign account with Account Administrator rights. In the Classic DocuSign Experience web application, click your profile image in the upper right and select **Preferences**. In the navigation pane on the left side of the page, under the Account Administration heading, click **Features**.
2. Scroll down to the DocuSign API heading and select **DocuSign Connect**.



Click **Save**.

- In the navigation bar on the left side of the page, under the Account Administration heading, click **Connect**. The DocuSign Connect Settings page appears. The page has a list of existing Connect configurations and allows you to access the Connect logs and failures pages.

You can have a maximum of 300 Connect configurations.



Note: There are specific configurations for setting up Connect to work with other external sources, such as DocuSign for Salesforce, eOriginal and Box. Refer to the appropriate guide for information about setting up Connect with those external sources.

- Click **+ Add Configuration** and select **Custom** to add a new Connect Configuration. The configuration page appears.

Note: If you have an existing configuration, you can just click on the configuration name to open that configuration page.

- On the configuration page you set the URL that the envelope information is sent to, select the events that generate information and select the users integrated with the information.

DocuSign Connect Settings > DocuSign Connect for Custom 1

URL to publish to:

Name:

☒ Allow Envelope Publish

☒ Enable Log (maximum 100)

☐ Include Documents

☐ Include Certificate of Completion

☐ Require Acknowledgment

☐ Sign Message with X509 Certificate

☐ Use Soap Interface (Soap Method: DocuSignConnectUpdate)

Namespace:

☐ Include X509 Certificate in Soap Header

☐ Include Time Zone Information

☐ Include Envelope Voided Reason

☐ Include Sender Account as Custom Field

☐ Include Document Fields

☐ Require Mutual TLS

Send information on these envelope events select all clear all	Send information on these recipient events select all clear all	Select users to integrate <input type="checkbox"/> All users integrated select all clear all (includes new users)
<input checked="" type="checkbox"/> Envelope Sent	<input checked="" type="checkbox"/> Recipient Sent	<input type="checkbox"/> Amy Silver
<input checked="" type="checkbox"/> Envelope Delivered	<input checked="" type="checkbox"/> Recipient Delivery Failed	<input type="checkbox"/> Chris Adams
<input checked="" type="checkbox"/> Envelope Signed	<input checked="" type="checkbox"/> Recipient Delivered	<input type="checkbox"/> Dave Dollar
<input checked="" type="checkbox"/> Envelope Completed	<input checked="" type="checkbox"/> Recipient Signed/Completed	<input type="checkbox"/> Dave Schueler
<input checked="" type="checkbox"/> Envelope Declined	<input checked="" type="checkbox"/> Recipient Declined	<input type="checkbox"/> Dianne Berry
<input checked="" type="checkbox"/> Envelope Voided	<input checked="" type="checkbox"/> Recipient Authentication Failure	<input checked="" type="checkbox"/> Doug Rybacki
		<input type="checkbox"/> Douglas Garrett Gibson

Set the DocuSign Connect configuration settings as follows:

- **URL to publish:** This is the web address and name of your listener or Retrieving Service endpoint. You must include the protocol (HTTP: or HTTPS:) in the web address for Demo account. You must include HTTPS:// in the web address for Production accounts because TLS is required in Production. Connect sends to the default ports of 443 for HTTPS: and 80 for HTTP.

Note: If you cannot use port 443 for Production contact DocuSign to review possible options.

- **Name:** This is name of the Connect configuration. The name helps identify the configuration in the list.
- **Allow Envelope Publish:** This option is selected by default. When this option is selected, data is sent to the web address. Clear this option to stop sending data while maintaining the custom information.
- **Enable Log:** Select this option to enable logging for this connection. It is recommended that you enable this option to facilitate troubleshooting any problems. If you do not want to enable logging for this connection, clear this box. You can have a maximum of 100 active logs for your account. The entries in active logs can be viewed on the **Logs** page, which is accessed by clicking **Logs** on the DocuSign Connect Settings page.
- **Include Documents:** Select this option to have Connect send the PDF documents along with the update XML. If you do not want to receive the PDF documents with the updates, clear this box. For more information about this see [Recommendations for Receiving Documents](#).
- **Include Certificate of Completion:** Select this option to have Connect include the Certificate of Completion with completed envelopes. If you do not want to receive the Certificate of Completion with the updates, clear this box.
- **Require Acknowledgement:** Select this option to log posting failures. DocuSign waits 100 seconds for an acknowledgement before recording a failure. DocuSign logs a failure if the attempt to reach the external endpoint returns anything other than an HTTP 200. The acknowledgement failure messages are logged in the **Failures** page, which is accessed by clicking **Failures** on the DocuSign Connect Settings page. When this option is selected, DocuSign will automatically attempt to repost any failures. Alternately, you can manually repost from the **Failures** page. See the [Publish Failures](#) section for more information.
- **Sign Message with X509 Certificate:** Select this option to have a message sent to the listener signed with a DocuSign X.509 certificate. The text is viewable, this just verifies the content viewed is what was sent.

Note: The **Include X509 Certificate** in the SOAP header and the **Sign Message with X509 Certificate** option are mutually exclusive.

- **Use Soap Interface:** Select this option to use the SOAP interface Retrieving Service. If this option is selected, you must add your **Namespace** for the service.

Optionally, if you want DocuSign to include the X.509 certificate in the SOAP header, select **Include X509 Certificate**. With this option the message is sent to the listener with the DocuSign X.509 certificate added to the SOAP Header. This allows you to check the information in the certificate to verify that it is signed by DocuSign.

Note: The **Include X509 Certificate** in the SOAP header and the **Sign Message with X509 Certificate** option are mutually exclusive.

- **Include Time Zone Information:** Select this option to include the envelope time zone information.
- **Include Envelope Voided Reason:** Select this option to include the reason an envelope was voided (as entered by the person that voided the envelope) in the message.
- **Include Sender Account as Custom Field:** Select this option to include the sender account information as a custom field.
- **Include Document Fields:** Select this option to include the Document Fields associated with envelope documents. Document Fields are optional custom name-value pairs added to documents using the API. Custom Document Field information is returned with status, but otherwise is not used by DocuSign.
- **Require Mutual TLS:** Select this option to enable mutual authentication at Transport Layer Security (TLS) with DocuSign. DocuSign uses TLS v1.2 by default for all Connect messages, but will support earlier versions of TLS if required by the listener.

If the **Sign Message with X509 Certificate** option or the **Include X509 Certificate** in the SOAP header option is selected, DocuSign will sign the message with our X509 certificate but it also requires that your listener provide an X509 certificate to DocuSign for mutual authentication.

Note: At this time, self-signed certificates are not supported in Connect and the Certificate Authority associated with your certificate must be in DocuSign's CA store.

- **Trigger Events:** Select the trigger events for updating your system. The two left columns of the table (envelope events and recipient events) show the DocuSign events that can trigger updates. The primary difference between envelope events and recipient events is that envelope events are only triggered when the envelope status changes, while recipient events are triggered each time information for a recipient changes.

Note: In cases where multiple events for a single envelope or recipient are queued, only one set of XML data with the most recent change is published to a customer's listener.

Example: For an envelope with one signer, when the signer finishes signing the envelope an Envelope Signed event is triggered, then the DocuSign system processes the envelope, and then an Envelope Completed event is triggered. If both of these events are in the Connect queue at the same time, only the Envelope Completed event is sent to a customer's listener.

The following tables provide information about when particulate envelope and recipient events are sent by Connect.

Envelope Event Descriptions

Envelope Event	Description
Sent	This event is sent when the email notification, with a link to the envelope, is sent to at least one recipient. The envelope remains in this state until all recipients have viewed the envelope.

Envelope Event	Description
Delivered	This event is sent when <u>all</u> recipients have opened the envelope through the DocuSign signing website. This does not signify an email delivery of an envelope.
Signed/Completed	The envelope has been signed and completed by all the recipients.
Declined	The envelope has been declined by one of the recipients.
Voided	The envelope has been voided by the sender.

Recipient Event Descriptions

Recipient Event	Description
Sent	This event is sent when an email notification is sent to the recipient signifying that it is their turn to sign an envelope.
Delivery Failed	This event is sent when DocuSign gets notification that an email delivery has failed. The delivery failure could be for a number of reasons, such as a bad email address or that the recipient's email system auto-responds to the email from DocuSign.
Delivered	This event is sent when the recipient has viewed the document(s) in an envelope through the DocuSign signing web site. This does not signify an email delivery of an envelope.
Signed/Completed	This event is sent when the recipient has completed (signed) the envelope. If the recipient is not a signer, this is sent when the recipient has completes their actions for an envelope.
Declined	This event is sent when the recipient declines to sign the document(s) in the envelope.
Authentication Failed	This event is sent when the recipient fails an authentication check. In cases where a recipient has multiple attempts to pass a check, it means that the recipient failed all the attempts.

- **User Accounts:** Select the users associated with the trigger events. These are the users whose trigger events are sent to the listener or Retrieving Service endpoint. If a user is not selected, no information is sent about the user's envelopes. A list of all users associated with your account is shown. You can select **All users integrated**, which selects all the current users and adds new users when they are added to the account.

6. Click **Save** to save your configuration. You can modify this information or add another custom tab by repeating the previous steps.

XML Information Structure

The DocuSign Connect publisher sends the status update in the body of an HTTP or SOAP XML post. The receiving web server needs to take the entire structure and parse the XML in order to make use of the various elements available in the XML.

Key Transaction Elements

The key transaction elements available are listed below. The full XML structure contains more, but these are the most commonly used data elements:

Status Information

- Sent Date/Time
- Envelope Status (in process, completed)
- Envelope ID

Envelope Information

- Document(s)
- Recipients
- Tabs
- Subject
- Email
- Custom Fields

Recipient activity and information

- Recipient ID(s)
- Recipient Email(s)
- Recipient Username(s)
- Recipient Note(s)
- Recipient Type (Signer, CC, CD)
- Recipient Sent Date/time
- Recipient Delivered Date/time
- Recipient Signed Date/time
- Recipient Routing Order
- Recipient Status Code (created, sent, delivered, signed, declined)
- Recipient Event (viewed, declined, signed, etc.)

Document Information

- Document Name(s)
- Document ID(s)
- Document Password(s)

Document Content

- Custom Tab Name
- Custom Tab Value
- Custom Tab Label

- Custom Tab Required/Not Required
- Custom Tab Type (text, checkbox, radio, list)
- TabTypeCode (signature, initial, name, company, title, date)
- Document PDF Bytes (base 64)

To make use of this information your application must parse the inbound XML looking for the data associated with each node you are evaluating. Then the application must extract the data and place it into the external application.

Form Field Data

DocuSign Connect is able to publish not only the status of the transaction, but also the values contained in any form fields or envelope fields in the transaction. This is useful to help interpret what transaction data can be updated into the external system. DocuSign supports many different field types including checkbox, radio button, form field, and drop down list. These all have a common name structure and the value from the signer can be extracted from the XML structure.

Technical Details

The XML post from DocuSign contains the EnvelopeStatus object along with DocumentPDF objects, if the configuration has the checkbox to include the push of the documents.

The DocuSign 3.0 API WSDL file that contains definitions for both structures is located on the DocuSign website. It can be found at: <https://www.docusign.net/api/3.0/api.asmx?wsdl>.

Running the Service

Once you activate DocuSign Connect, DocuSign sends an XML object to the secure URL entered on the configuration page for every event selected and from every user selected. If your application is not configured to accept these post messages, the DocuSign system will not return an error.

It is important that you do not turn on DocuSign Connect if you have not configured a listener or Retrieving Service endpoint at the URL entered in the configuration page. Once you have the listener set up, you may test the publisher by sending transactions and observing the behavior of your application.

Test code for an HTTPS listener based on PHP is available from DocuSign. You can get a copy by downloading the SDK from the DocuSign website.

Note: You must include the protocol (HTTP: or HTTPS:) in the web address for Demo account. You must include HTTPS:// in the web address for Production accounts because TLS is required in Production. Connect sends to the default ports of 443 for HTTPS: and 80 for HTTP. If you cannot use port 443 for Production contact DocuSign to review possible options.

Best Practices

In order to take advantage of DocuSign Connect, a clear understanding of the use of the information needs to be understood. Be sure to ask questions such as:

1. What data do you want to capture?
2. Who will be accessing this information?
3. What decisions or reporting will be generated?

4. Should the document be pushed?

These questions must be thought out and agreed in order to deploy a solution that will meet your business needs. Additionally, developing the secure listener application to have some flexibility can enable modifications to the data that is collected without requiring coding for minor adjustments. This field mapping approach enables future modifications and changes that can be made by analysts.

Recommendations for Receiving Documents

There are several factors to consider when determining how to receive envelope PDF documents from DocuSign in your document management system. While it is perfectly acceptable to just select the [Include Documents option](#) in your Connect configuration to have Connect include the PDF documents when an event is triggered; you might consider using the DocuSign API to retrieve the documents when a triggering event (such as envelope status Complete) is received if the following are considerations for your architecture:

- Throttling of document retrieval: If for some reason your service is not available to Connect or you are expecting a very high volume, your processing routines might be overwhelmed by the Connect messages. If you are not appropriately scaled to handle the volume or catch-up spikes from connection outages, you might be at risk of an out of memory error.
- File size and storage: Connect sends PDF documents in base 64 encoding format, which is on average 1/3 larger than a binary file format. This might or might not be a storage consideration for you.

In these cases, DocuSign often recommends that corporate and enterprise customers get the completed envelope message through Connect without including the PDF documents, then makes an API call to get those PDF documents at a time that make sense to your workflow and allow you to throttle the processing of those files.

Connect SOAP Publishing

The following configurations are available in Connect:

- Retrieving Service endpoint (URL).
- Retrieving Service method name. DocuSignConnectUpdate.
- Retrieving Service method's argument name. DocuSignEnvelopeInformation.
- Retrieving Service Namespace. The default is DocuSignConnectListener.
- X.509 security in the SOAP Header. DocuSign uses the standard WSE3 BinarySecurityToken in the Soap Header to pass the certificate.

When planning the SOAP Publishing Retrieval Service, you need to do the following:

1. Create a web service that has the DocuSign API as a web reference.
2. Create a method in your web service that matches the Retrieving Service method name.
3. Have a single argument on the method that matches

The argument must be of the type *DocuSignAPI.DocuSignEnvelopeInformation*, where *DocuSignAPI* is configurable based on the namespace used when importing the DocuSign API web service.

4. Return the EnvelopeID passed to it, if received. Otherwise, it should return a SOAP Fault.

Publish Failures

If the **Require Acknowledgement** option is selected and a publication message fails to be acknowledged within 100 seconds, the message goes back into the queue and the system will retry delivery after a successful acknowledgement is received. If the delivery fails a second time, the message is not returned to the queue for sending until Connect receives a successful acknowledgement and it has been at least 24 hours since the previous retry. There is a maximum of ten retries.

You can always view the list of Connect publish failures by going to the **Failures** page by clicking **Failures** on the DocuSign Connect Settings page. You can manually republish these items from the **Failures** page.

Additionally, you can use the API to request the Connect publish failures, see the [Sample API Calls and Schema](#) below. In cases where your listener is “down” you make one call (GetConnectFailures) to find out what messages were missed and republish those events to your listener with PublishConnectFailures. You might not even need to call either method, as the next notification event that Connect successfully publishes will trigger an auto-republish of any messages in the queue.

If **Require Acknowledgement** is not selected, the delivery of the status is not guaranteed and the DocuSign System normally does not retry delivery in cases where the web server is unavailable. A common solution to ensure that no status changes are missed during the day is to create a nightly query using the DocuSign API RequestStatuses method to reconcile any events that were missed.

Manually Publishing Transactions

You can also manually publish transactions to your listening web server using the Envelope Report tool in the Classic DocuSign Experience web application. This is done by going to the web application **Preferences** section and, under the Account Administration section, clicking **Envelope Publish**. Select the search criteria for your reports and click **View**, a list of envelope events that meet the search criteria is displayed. Select the **Publish XML** check boxes for envelopes you want to publish, select (or verify that it is selected) the **Apply DocuSign Connect Settings** check box, and then click **Publish XML**.

Sample API Calls and Schema:

```
[WebMethod]
public ConnectFailure[] GetConnectFailures(ConnectFailuresFilter
ConnectFailuresFilter)

[WebMethod]
public PublishConnectFailuresResult[]
PublishConnectFailures(PublishConnectFailuresFilter PublishConnectFailuresFilter)

<xs:simpleType name="ConnectPublishStatus">
  <xs:annotation>
    <xs:documentation>The list of allowable DocuSign Connect publish
statuses.</xs:documentation>
  </xs:annotation>
  <xs:restriction base="xs:string">
    <xs:enumeration value="Queued" />
    <xs:enumeration value="Success" />
    <xs:enumeration value="Fail" />
  </xs:restriction>
</xs:simpleType>
<xs:element name="ConnectFailuresFilter">
  <xs:annotation>
```

```

    <xs:documentation>This element provides the filtering criteria for requesting
DocuSign Connect failures.</xs:documentation>
  </xs:annotation>
  <xs:complexType>
    <xs:all>
      <xs:element name="AccountId" type="dsx:DSXId" />
      <xs:element name="DateFrom" type="xs:dateTime" minOccurs="0" />
      <xs:element name="DateTo" type="xs:dateTime" minOccurs="0" />
    </xs:all>
  </xs:complexType>
</xs:element>
<xs:element name="ConnectFailure">
  <xs:annotation>
    <xs:documentation>This element provides DocuSign Connect
failures.</xs:documentation>
  </xs:annotation>
  <xs:complexType>
    <xs:all>
      <xs:element name="EnvelopeId" type="dsx:DSXId" />
      <xs:element name="AccountId" type="dsx:DSXId" />
      <xs:element name="Failed" type="xs:dateTime" />
      <xs:element name="Retried" type="xs:dateTime" minOccurs="0" />
      <xs:element name="RetryCount" type="xs:int" />
      <xs:element name="EnvelopeStatus" type="dsx:EnvelopeStatusCode" />
      <xs:element name="Sender" type="xs:string" />
      <xs:element name="Email" type="xs:string" />
      <xs:element name="Subject" type="xs:string" />
      <xs:element name="Error" type="xs:string" />
    </xs:all>
  </xs:complexType>
</xs:element>
<xs:element name="GetConnectFailures">
  <xs:complexType>
    <xs:all>
      <xs:element ref="dsx:ConnectFailuresFilter" />
    </xs:all>
  </xs:complexType>
</xs:element>
<xs:element name="PublishConnectFailuresFilter">
  <xs:annotation>
    <xs:documentation>This element provides the filtering criteria for publishing
DocuSign Connect failures.</xs:documentation>
  </xs:annotation>
  <xs:complexType>
    <xs:all>
      <xs:element name="AccountId" type="dsx:DSXId" />
      <xs:element name="Synchronous" type="xs:boolean" minOccurs="0" />
      <xs:element name="EnvelopeIds">
        <xs:complexType>
          <xs:sequence>
            <xs:element name="EnvelopeId" type="dsx:DSXId" maxOccurs="unbounded" />
          </xs:sequence>
        </xs:complexType>
      </xs:element>
    </xs:all>
  </xs:complexType>
</xs:element>
<xs:element name="PublishConnectFailuresResult">
  <xs:annotation>
    <xs:documentation>This element provides the filtering criteria for requesting
DocuSign Connect failures.</xs:documentation>
  </xs:annotation>
  <xs:complexType>
    <xs:all>

```

```

<xs:element name="EnvelopeId" type="dsx:DSXId" />
<xs:element name="Status" type="dsx:ConnectPublishStatus" />
<xs:element name="StatusMessage" type="xs:string" />
</xs:all>
</xs:complexType>
</xs:element>
<xs:element name="PublishConnectFailures">
  <xs:complexType>
    <xs:all>
      <xs:element ref="dsx:PublishConnectFailuresFilter" />
    </xs:all>
  </xs:complexType>
</xs:element>

```

Sample Connect Message

The information below is a sample DocuSign Connect message in XML format. Note that the personal information (names and email addresses), document names, and PDFBytes information has been removed from this sample.

```

<?xml version="1.0" encoding="utf-8"?>
<DocuSignEnvelopeInformation xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns="http://www.docusign.net/API/3.0">
  <EnvelopeStatus>
    <RecipientStatuses>
      <RecipientStatus>
        <Type>Signer</Type>
        <Email>user.email@address.com</Email>
        <UserName>User Name</UserName>
        <RoutingOrder>1</RoutingOrder>
        <Sent>2010-06-26T09:19:18.883</Sent>
        <Delivered>2010-06-26T09:19:40.723</Delivered>
        <DeclineReason xsi:nil="true" />
        <Status>Delivered</Status>
        <RecipientIPAddress>::1</RecipientIPAddress>
        <CustomFields />
        <TabStatuses>
          <TabStatus>
            <TabType>Custom</TabType>
            <Status>Active</Status>
            <XPosition>364</XPosition>
            <YPosition>52</YPosition>
            <TabLabel>Radio</TabLabel>
            <TabName>Two</TabName>
            <TabValue />
            <DocumentID>1</DocumentID>
            <PageNumber>2</PageNumber>
            <OriginalValue />
            <ValidationPattern />
            <RoleName>TestRole</RoleName>
          </TabStatus>
        </TabStatuses>
        <AccountStatus>Active</AccountStatus>
        <RecipientId>fb89d2ee-2876-4290-b530-ff1833d5d0d2</RecipientId>
      </RecipientStatus>
    </RecipientStatuses>
    <TimeGenerated>2010-06-26T09:19:45.771206-07:00</TimeGenerated>
    <EnvelopeID>0aa561b8-b4d9-47e0-a615-2367971f876b</EnvelopeID>
    <Subject>CreateEnvelopeFromTemplates Test</Subject>
    <UserName>User Name</UserName>
    <Email> user.email@address.com </Email>
    <Status>Delivered</Status>
    <Created>2010-06-26T09:16:21.27</Created>
  </EnvelopeStatus>
</DocuSignEnvelopeInformation>

```



```
<Sent>2010-06-26T09:19:19.01</Sent>
<Delivered>2010-06-26T09:19:40.747</Delivered>
<ACStatus>Original</ACStatus>
<ACStatusDate>2010-06-26T09:16:21.27</ACStatusDate>
<ACHolder>ACHolder Name</ACHolder>
<ACHolderEmail> ACHolder.email@address.com </ACHolderEmail>
<ACHolderLocation>ACHolder Location</ACHolderLocation>
<SigningLocation>Online</SigningLocation>
<SenderIPAddress>::1 </SenderIPAddress>
<EnvelopePDFHash />
<CustomFields>
  <CustomField>
    <Name>Envelope Field 1</Name>
    <Show>False</Show>
    <Required>False</Required>
    <Value />
  </CustomField>
  <CustomField>
    <Name>Envelope Field 2</Name>
    <Show>False</Show>
    <Required>False</Required>
    <Value />
  </CustomField>
</CustomFields>
<AutoNavigation>true</AutoNavigation>
<EnvelopeIdStamping>true</EnvelopeIdStamping>
<AuthoritativeCopy>false</AuthoritativeCopy>
<DocumentStatuses>
  <DocumentStatus>
    <ID>1</ID>
    <Name>Document_Name</Name>
    <TemplateName>radio parents</TemplateName>
    <Sequence>1</Sequence>
  </DocumentStatus>
</DocumentStatuses>
</EnvelopeStatus>
<DocumentPDFs>
  <DocumentPDF>
    <Name>DocumentPDF_Name</Name>
    <PDFBytes>PDFBytes_Information</PDFBytes>
  </DocumentPDF>
</DocumentPDFs>
</DocuSignEnvelopeInformation>
```

