ADOBE® LIVECYCLE® ES3 OVERVIEW

Legal notices

Contents

Chapter 1: About This Document

Chapter 2: About Adobe LiveCycle Enterprise Suite 3

Chapter 3: Foundation

Chapter 4: LiveCycle Development Tools

Chapter 5: LiveCycle Modules

Chapter 6: Business Process Management

Chapter 7: Forms automation

Chapter 8: Document information and security

Chapter 9: Communications management

Chapter 10: Content Management

Chapter 11: Services for LiveCycle Modules

Chapter 1: About This Document

Adobe® LiveCycle® Enterprise Suite 3 (ES3) software enables organizations to automate and accelerate the flow of business-critical information among employees, customers, suppliers, and constituents.

This document provides an overview of Adobe LiveCycle.

Who should read this document

This document is intended for business analysts, form authors, process developers, and programmers who need to know how they can use LiveCycle to accomplish the following:

- Design business processes
- Create forms
- Develop process diagrams or client applications to start the business processes

  This document is also useful to administrators who plan to install, configure, or administer LiveCycle.

Additional information

The resources in this table can help you learn more about LiveCycle.

For information about	See
Preparing to install LiveCycle	Preparing to Install LiveCycle (Single Server) Preparing to Install LiveCycle (Server Cluster)
Installing and deploying LiveCycle	Installing and Deploying LiveCycle Using JBoss Turnkeyor Installing and Deploying LiveCycle for JBoss Installing and Deploying LiveCycle for WebSphere Installing and Deploying LiveCycle for WebLogic
Installing and configuring LiveCycle Workbench 10	Installing LiveCycle Workbench 10
How to create your first LiveCycle application	Creating Your First LiveCycle Application
Descriptions of LiveCycle services	Services Reference for LiveCycle
Other services and products that integrate with LiveCycle	www.adobe.com
Patch updates, technical notes, and additional information about this product version	Adobe LiveCycle Technical Support

Chapter 2: About Adobe LiveCycle Enterprise Suite 3

Adobe LiveCycle Enterprise Suite 3 (ES3) is an enterprise server platform that enables organizations to automate and streamline business processes that span systems, business rules, and departments. The platform provides all the tools required to design, implement, and automate the documents and processes that are used by your organization. Your organization can use these processes internally and with external customers and stakeholders.

Faster time-to-value

LiveCycle provides a faster time-to-value through a seamless, tightly integrated development environment that utilizes LiveCycle for faster development.

Model-driven application development in LiveCycle enables business analysts or developers to graphically define data objects and their behaviors. They can use these definitions to create rich user interfaces with little or no coding. Services for managing these objects are automatically generated; user interfaces are bound to these services by using a simple drag-and-drop interface. The common data model allows modular building blocks to be built for use in common workflows, resulting in consistency and efficiency. The common data model also allows developers to create applications faster and collaborate more effectively. Developers can view data across processes or components and identify the relationships between them.

LiveCycle provides streamlined development with the reduced number of steps it takes to complete development tasks. Team-based features allow administrators to control which developer or development team can access which assets. Control and collaboration is possible through check-in and check-out capabilities.

Rich user experiences

LiveCycle combines Adobe PDF and Adobe Flex® technology to provide a unified developer experience for creating applications that can provide more engaging user experiences.

Guides (deprecated) minimize the skill set required for building processes and Rich Internet Applications (RIAs). For example, business analysts can now create Guides (deprecated) based on the data model they define without help from a developer. Business analysts can preview and quickly create Guides (deprecated) and submit them to a workflow with virtually no coding or Java™ skills. Guides (deprecated) also provide support for built-in validation and formatting styles for data such as credit card numbers.

Important: *Effective March 10th, 2012, Adobe is deprecating the Guides capabilities of Adobe® LiveCycle® ES. The Guides functionality is available for upgrade purposes only and will be removed from the product after two major releases.*

Performance and productivity

Application building blocks allow developers to reuse components in repeatable tasks. Business users can simplify daily tasks, customize their workspace, collaborate with others through screen sharing, VoIP, or instant messaging.

Workbench is an Eclipse-based development environment you can use to create and manage business processes and the forms that are used in those processes.

Key LiveCycle components

LiveCycle provides the resources that business analysts, form and application developers, and administrators require to create and deliver applications that support user interaction in cross-platform technologies, including Adobe Reader®, HTML, and Adobe Flash®.

Foundation and modules are installed and deployed by using a single installation and deployment framework. The resultant suite integrates into a unified solution, which plugs into an enterprise back-end environment and supports various cross-platform clients for end-user interaction.

Foundation

Foundation provides the underlying server capabilities that enable the deployment, execution, and management of modules. Foundation consists of several pieces.

Foundation components

Components that enable the LiveCycle Server to integrate with a common IT infrastructure. For example, these components can enable the following tasks:

- Querying a directory through LDAP
- Sending and receiving email
- Querying a relational database
- Writing files to the file system

Service container

The service container provides the common run-time environment to support all modules and associated services.

Administration tools

Foundation includes several administration tools:

LiveCycle Administration Console: A web-based interface that system administrators use to manage a LiveCycle deployment. The key features are as follows:

- Administration pages for modules and services
- Configuration of server settings, such as port numbers
- Configuration of users, user groups, roles, and associated permissions
- Deployment and configuration of LiveCycle applications

See LiveCycle Administration Console.

Configuration Manager: Enables the configuration and deployment of the product, including adding service packs and patches.

LiveCycle Launch Pad: An Adobe AIR® client application that provides access from your desktop to the services on your LiveCycle Server.

LiveCycle modules

Modules provide the functional services that are required to build customer engagement applications that can be deployed to the LiveCycle Server. Here are examples of some of the business functions module offer:

- Capturing data in Guides (deprecated)
- Encrypting PDF documents
- Applying policies or usage rights to PDF documents
- Converting an image to PDF
- Remote invocation of services

Any additional modules you deploy provide the specific business functions that you build LiveCycle applications on. Some modules, such as LiveCycle Rights Management and LiveCycle Reader Extensions, include web-based interfaces that let users interact with components. For example, users can use the interfaces to apply policies or usage rights to documents that are part of a business processes. LiveCycle Workspace lets end users initiate and participate in form-based business processes.

Modules typically consume documents or produce documents. Module components can perform any of the following tasks:

- Render PDF forms
- Encrypt PDF documents with passwords
- Apply digital signatures to PDF documents
- Merge data into a simple PDF form
- Assemble information from multiple PDF documents into a single document

Using development tools, you can combine process management with electronic forms, document security, and document generation into a LiveCycle application. The application integrates components to ensure that business processes work inside and outside the firewall, and for users in online and offline environments.

Enterprise Rich Internet Applications (RIAs)

Data Services Integrate RIAs with LiveCycle, Java 2 Platform, Enterprise Edition (J2EE) applications, and business logic.

Business process management

LiveCycle Process Management Streamline human-centric business processes across your firewall.

Forms automation

LiveCycle Forms Deploy interactive XML-based forms in Adobe Reader, Adobe Flash Player, or web browsers.

Reader Extensions Fill, sign, comment on, or save PDF files using only Adobe Reader.

Document and information security

Rights Management Manage usage rights to protect sensitive documents in PDF, Microsoft Office, or CAD formats.

LiveCycle Digital Signatures Automate the signing, certification, and validation of digital signatures in PDF documents.

Communications management

LiveCycle Output Dynamically generate personalized documents on demand in print and electronic formats.

LiveCycle Production Print Dynamically generate personalized documents for high-volume printing and enveloping

LiveCycle PDF Generator Automate the creation and assembly of PDF documents from virtually any file format.

Correspondence Management enables you to quickly assemble correspondence from both pre-approved and custom-authored content in a streamlined process from creation to archival.

Content management

Connectors for ECM Extend your LiveCycle applications to connect with industry-leading enterprise content management (ECM) systems.

LiveCycle development tools

LiveCycle provides development tools so that various users, from business analysts to (J2EE) developers, can collaborate on the design, development, testing, and deployment of a LiveCycle application.

LiveCycle Workbench ES3: An Eclipse-based development environment that lets users build applications that consist of forms and business processes. Form developers can create forms that have either a fixed layout or a flowable layout. A *fixed layout* remains exactly as it was designed; it does not change to accommodate the amount of incoming data). A *flowable layout* expands or shrinks to accommodate the amount of data being merged or entered. These forms can be rendered by using a number of client technologies, such as PDF, HTML, and Flash. Business analysts and process developers can use Workbench to design and deploy business processes that leverage the modules.

LiveCycle Designer: A graphical form design tool that simplifies the creation of forms. Form designers can create applications that generate dynamic, data-driven documents and produce customized business documents for print, web, or archival. They can build and maintain data capture solutions that read from, validate against, and add to corporate data sources. With Designer, form developers can integrate PDF documents into existing workflows by binding forms to XML schemas, XML sample files, databases, and web services. They can merge forms and documents that are created in Designer with business data and render them as many file types. Some of these file types are PDF and printing for PCL, Adobe PostScript® and Zebra (ZPL) printers.

Adobe Flash® Builder™: An Eclipse-based development environment for developing RIAs with the Flex framework. Using Flash Builder 4.5, developers can quickly build and deploy applications that are expressive, intuitive, and interactive. Using Flash Builder, they can also customize their applications to ensure an appearance that represents the needs of end users.

LiveCycle SDK: Provides developers with samples and tools that they can use to build new LiveCycle applications or integrate with existing applications.

Architecture

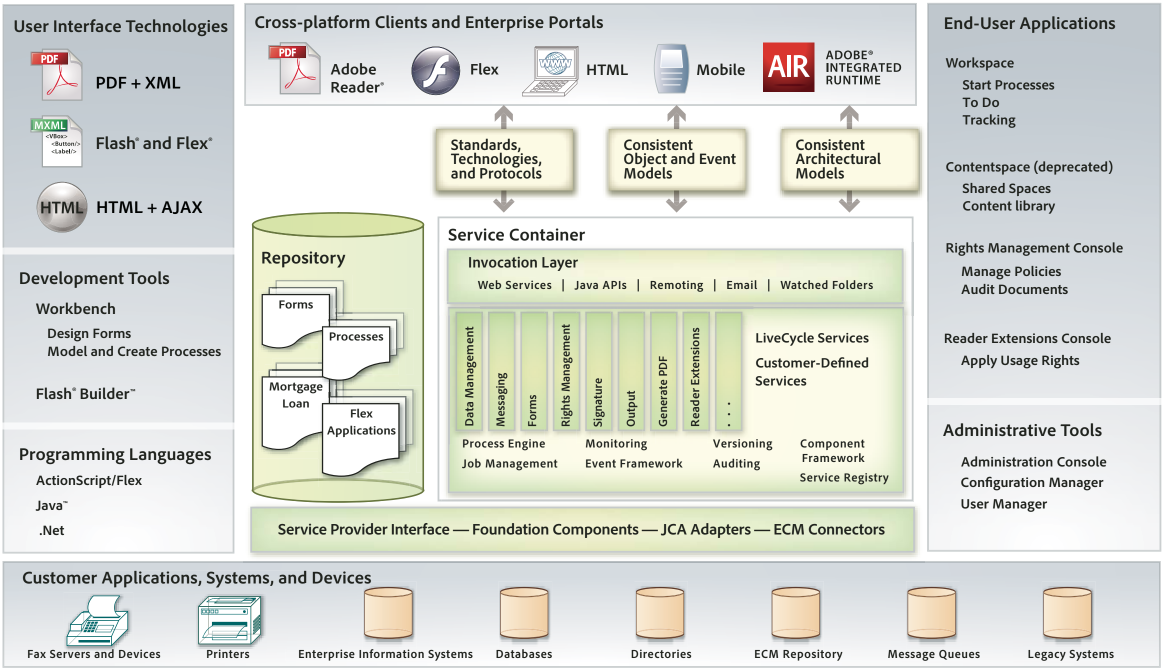
LiveCycle implements a service-oriented architecture that relies on loosely coupled software services to support business processes and perform operations on documents. Services run within the service container.

The *service container* is an extensible component model based on software-oriented architecture (SOA) principles; components can be added, upgraded, reused, or removed with minimal administration. Components are independent units of functionality that plug into the service container for deploying, orchestrating, and executing services. The service container coordinates and monitors the components.

The service container can host any number of services and provides the infrastructure necessary to discover, execute and deploy, secure, run, and configure the services.

The service container provides a consistent way to interact with the services within the container, repository, and enterprise integration, regardless of the invocation method. Services can be called by using Java APIs, web services, watched folders, or email. Service endpoints are also exposed as destinations that RIAs can easily call.

This illustration provides a closer view of how the key product components fit together within the server architecture.



The development area includes Workbench tools and the repository. Development of forms and Guides (deprecated), process flows, and related collateral is done in Workbench. The Eclipse-based development environment can also include Flash Builder (available separately). You can use Flash Builder to develop RIAs, create custom components for use in Guides (deprecated), customize Workspace, and more. (See LiveCycle Workspace Help.)

The resultant design assets are stored in the repository, which provides versioning and maintains resource dependencies. This central storage and maintenance model promotes the reusability of artifacts, enables developers to collaborate on application development, and provides security within the development environment.

The common invocation layer ensures consistent interaction with the server through various invocation methods. It supports programmatic and non-programmatic invocation methods, including web services, Java APIs, watched folders, and email-based invocation methods. All of these invocation methods are available to use with any service.

LiveCycle provides integration adapters to ensure compatibility with your enterprise back-end systems. These back-end systems include user directories, authentication systems, ECM systems, web services, JMS, Java Remote Method Invocation (RMI), and email.

The LiveCycle service-oriented architecture maximizes the scalability and reusability of services. It ensures that you can easily add, remove, and upgrade services. User authentication, service deployment, and invocation are all managed within a common server infrastructure to ensure system security and performance.

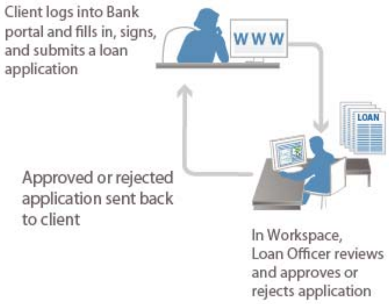
Using cross-platform clients, such as Adobe Reader, Flash Player, and Adobe AIR™, you can ensure that your applications will be easily accessible, immediately usable, and consistent across platforms and devices.

The Adobe AIR run time lets developers use proven web technologies to build RIAs that deploy to the desktop and run across operating systems. Adobe AIR offers an exciting new way to engage customers with innovative, branded, desktop applications without requiring changes to existing technology, people, or processes.

LiveCycle example scenarios

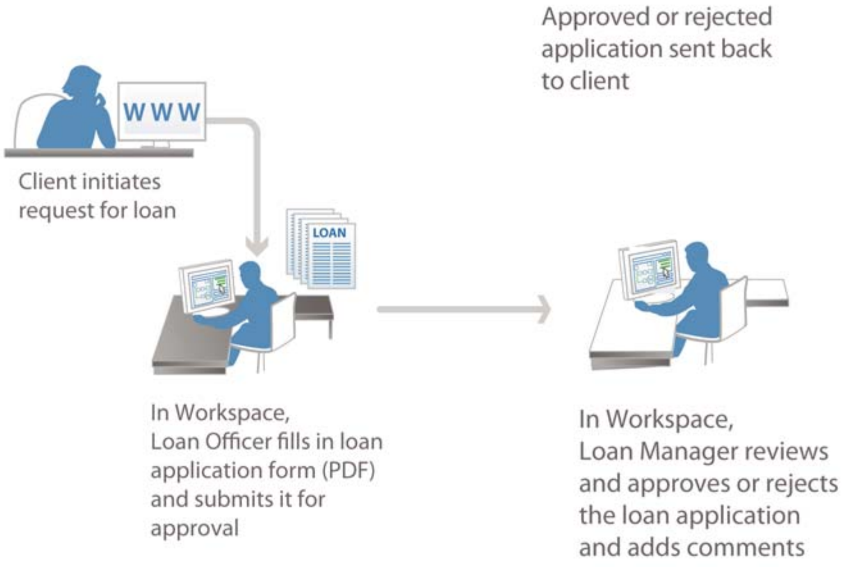
Submit and approval using a Guide (deprecated)

In this example, a bank client applies for a loan through a bank portal. The client fills, signs, and submits an application through a Guide (deprecated), and then Workspace is used to review the loan application. When approved or rejected, the application is sent back to the client.


Client logs into Bank portal and fills in, signs, and submits a loan application
WWW
LOAN
Approved or rejected application sent back to client
In Workspace, Loan Officer reviews and approves or rejects application

Submit and approve a PDF form using Workspace

In this example, a loan application form is submitted, reviewed, and approved using Workspace.


Approved or rejected application sent back to client
WWW
Client initiates request for loan
LOAN
In Workspace, Loan Officer fills in loan application form (PDF) and submits it for approval
In Workspace, Loan Manager reviews and approves or rejects the loan application and adds comments

Generate customer correspondence

Adobe Correspondence Management

Adobe™ provides frameworks that bundle the LiveCycle modules required to develop solutions, together with ready-to-use solution-specific components. These include solution-specific documentation and templates to use as sample implementations. These solution templates assist with the building of proof of concept projects and accelerate development of production solutions.

Correspondence Management: Automates various types of correspondence, from welcome packages and confirmations to proposals and claim letters. Business users can intuitively assemble individualized customer correspondence with pre-approved content blocks, interactive media elements, and pre-filled electronic forms. The letter is delivered securely to the customer, enabling them to fill in and submit the appropriate information, eliminating wasteful paper submissions.

Deployment

Deployment possibilities for LiveCycle are flexible. You can deploy it as a single stand-alone server running single or multiple modules or as a volume production system running multiple modules on clustered servers.

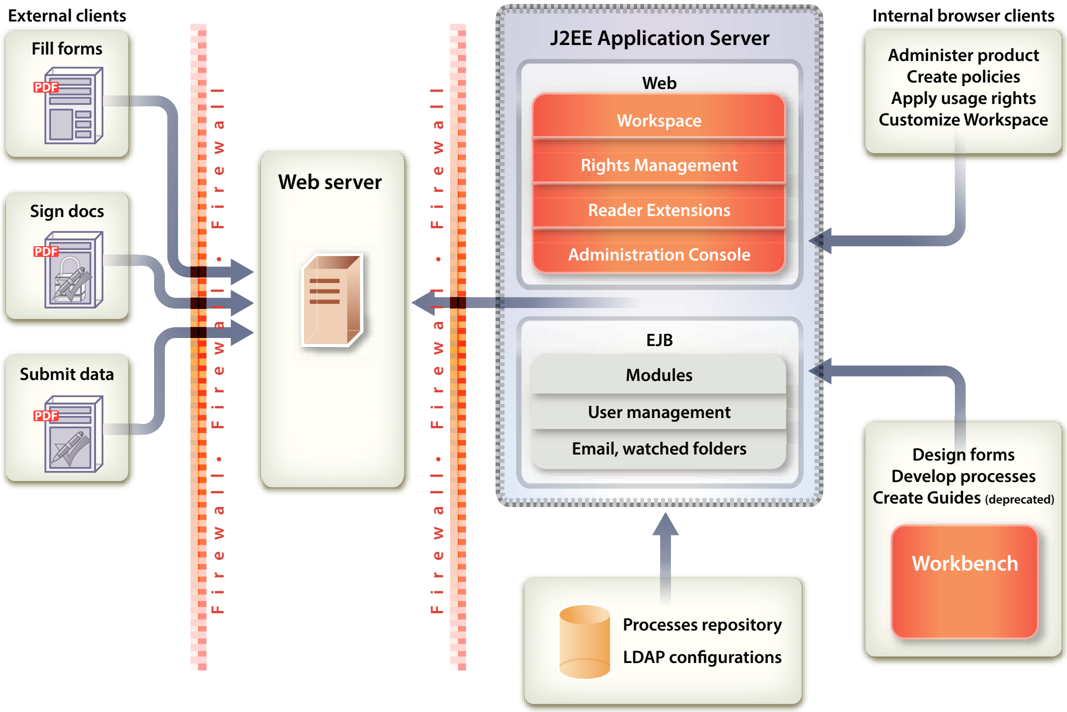
The following sections describe single-node deployment for a small production system, clustered deployment for volume production in an enterprise environment, and an example of deployment phases.

Single node deployment

A single node deployment is useful for running simple jobs. For example, PDF Generator can be deployed to a single node for converting Microsoft Office documents to PDF documents.

You can install and deploy LiveCycle on a single application server instance by using the turnkey method. Installing on a single application server installs and deploys an instance of the Red Hat® JBoss Application Server and the MySQL database server. It also deploys LiveCycle to the application server as a complete and ready-to-use application.

You can configure and deploy LiveCycle automatically to IBM® WebSphere Application Server or Oracle® WebLogic Server if they are already set up. In this case, you must also install a database server. You can install the database, as well as Workbench or the LiveCycle SDK, on the same computer or a separate computer.



Clustered deployment

The structure of the single node deployment is replicated in a clustered enterprise deployment but on a larger scale. In larger scale deployments, database and LDAP servers are typically located remotely, and a more rigorous distinction between development, testing, and production environments is implemented. (See Preparing to Install LiveCycle (Server Cluster) .

Deployment phases

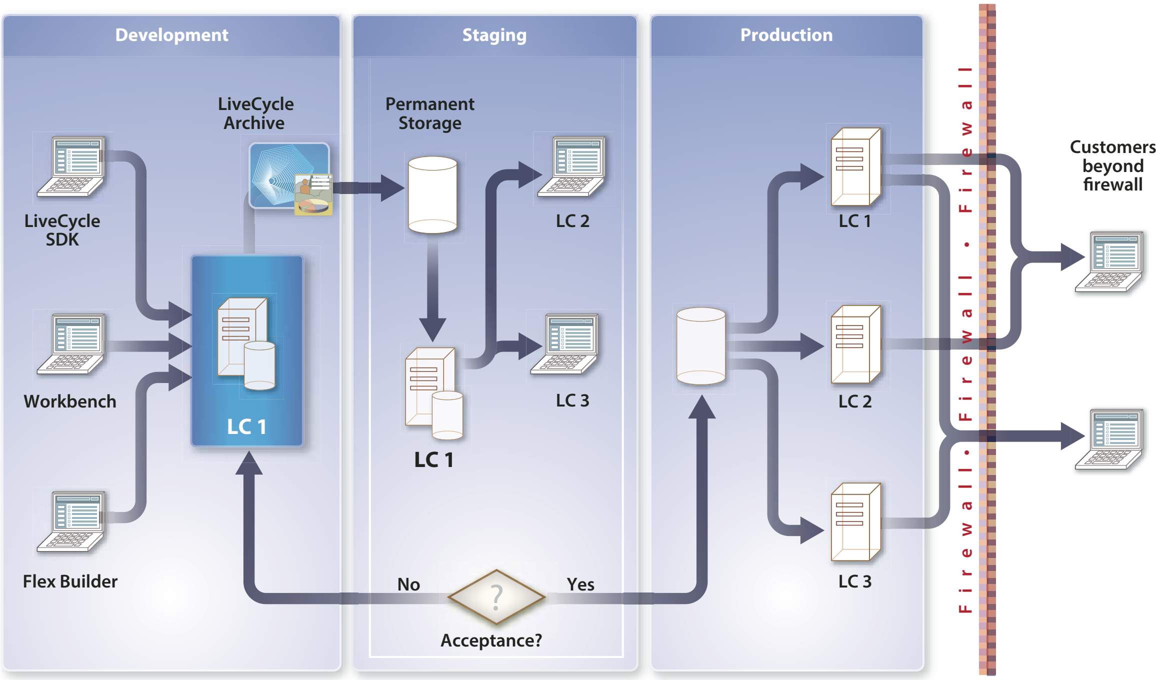
In an enterprise-wide deployment, the system is set up so that several people have different roles in the system. For example, an IT administrator installs and deploys the production system (and possibly the testing system), and sets up the security infrastructure. Form developers design forms and Guides (deprecated); process developers design processes; administrators deploy services, manage server performance, and so on. The system integrates with the organization's back-end database infrastructure.

Typically, various users use LiveCycle in three phases:

Development: Form authors, process developers, or programmers use the development and programming tools to create processes and custom applications for deployment.

Staging: Administrators and testers deploy and test the applications in a system that mimics the final production deployment scenario.

Production: Administrators deploy, monitor, and maintain services and applications. End users interact with services within and outside your organization (and within and outside the firewall).



Administrators move all assets (such as PDF forms, images, process diagrams, and other files required) for an application to work from one phase to the next. Developers typically package all application assets into a LiveCycle archive (DSA) file and transfer it from development to staging to production.

Chapter 3: Foundation

Foundation provides the underlying server capabilities on which LiveCycle solutions can be built. Included in every LiveCycle installation, Foundation provides all the tools required for application development, invocation, management, and administration. It also integrates with your organization's existing infrastructure of databases and LDAP systems, and works to assemble the services invoked by deployed processes.

Foundation includes the following services and components that integrate so that you can deploy and manage business documents and processes:

- Process coordination and service management
- Common architectural foundation
- User management
- Event management
- Administration Console
- Launch Pad

Process coordination and service management

Foundation provides a common run-time environment, or service container, for all services that run in LiveCycle. This common environment facilitates process coordination, enabling developers to link multiple modules in a process.

Backup and restore

LiveCycle is designed to be deployed in a mission-critical environment, where appropriate disaster-recovery procedures are in place. To help ensure recovery from failure, you can now back up the LiveCycle system while it is running. This feature allows maintenance of longer *up times* while at the same time minimizing data loss in the event of a disaster.

IPv6 support

LiveCycle now provides support for Internet Protocol version v6 (IPv6), the successor to the current Internet protocol version in general use on the Internet.

JMX support

To enable system administrators to fine-tune their environment and view availability through tools, such as HP OpenView or IBM Tivoli, using LiveCycle, you can view system level metrics. For example, you can view the number of invocations per second through the Java Management Extensions (JMX) standard.

Job management and monitoring

The Job Manager service can asynchronously invoke a component and retrieve the persisted results as well as monitor the execution of each invocation.

Using the Job Manager service API, developers can do these tasks:

- Create a new asynchronous job using the specified invocation request.
- Complete an existing job using the specified invocation response.
- Terminate, suspend, or resume an existing job identified by the specified job ID.
- Get the job ID that represents the status of a long-lived process. The job status indicates whether a job was queued, running, completed, terminated, suspended, or resumed. The status can also indicate whether a request was issued to complete, terminate, suspend, or resume a job.

  For more information about invoking long-lived processes, see Programming with LiveCycle

  For information about managing processes using Administration Console, see the LiveCycle Administration Console Help.

Process archive and deletion

LiveCycle now provides a set of APIs and sample scripts so that administrators can delete completed processes, including the ability to define queries. An example is deleting all purchase order processes with a value under $10,000 completed more than 6 months ago. With this feature, you can better manage the process data and audit information within the database.

Repository and service registry

The repository provides the capability to manage the assets that developers create as part of their LiveCycle applications.

Developers can access the repository by using the Form Design perspective in Workbench or programmatically using the repository API. A developer must be granted access before accessing the repository. Each time a developer uses Workbench, a connection to the repository is made. The repository is exposed as a hierarchical directory structure. One or more developers can share the repository from Workbench.

Staging and production systems each have their own repository. For example, an organization’s quality assurance team tests a service in their staging environment. When the tests are successful, the team deploys the service into their production environment. When a service is deployed into production, it has no dependency on any design-time assets in the staging environment’s repository. The organization can optionally use the service registry’s access control mechanisms to restrict access to the service deployed in the production environment. This enables the organization to pilot a deployment with a restricted group of users.

Registering assets and services

The repository provides storage capabilities. When a developer creates an application, the developer can deploy the assets in the repository instead of deploying them on a file system. The assets may consist of XML forms, PDF forms (including Acrobat forms), fragments, images, processes, profiles, policies, DDX files, XML schemas, WSDL files, SWF files, and test data.

The repository tracks the version of each asset in a LiveCycle application. At run time, services can retrieve assets from the repository as part of completing an automated business process.

Creating LiveCycle applications

The repository maintains dependency relationships among all the assets it manages. LiveCycle uses these dependency relationships to assemble all the necessary assets into a LiveCycle application.

The application manager supports packaging the assets that are part of a LiveCycle application into a LiveCycle archive file. The archive file facilitates the transfer of an application from development to staging to production.

When a LiveCycle application is deployed, all the assets within it are also deployed. The process of deploying some of those assets results in services being registered in the service registry, which can be invoked by the Invocation framework.

For example, when a process is deployed, a service entry is created in the service registry that allows the process to be invoked as a service. If the service is published, a WSDL file is created and added to the service registry, along with the necessary metadata that the LiveCycle SDK framework uses to invoke the service.

Service registry

The service registry is used at run time to resolve the name of a service to an actual endpoint in the service container. Many different versions of a service can be deployed at any one time in the service registry. The Invocation framework, along with version information provided by the calling application, is used to bind the correct version of the service.

Services require a service container to run, similar to how Enterprise JavaBeans (EJBs) require a J2EE container. LiveCycle includes only one implementation of a service container. This implementation is responsible for managing the life cycle of a service, including deploying it and ensuring that requests are sent to the correct service. The service container is also responsible for managing documents that are consumed and produced by a service.

For more information about the service container, see Programming with LiveCycle.

Scripted deployment

Scripted deployment lets you automate the deployment of applications from a development environment to a staging or production environment.

The contents of a LiveCycle archive are extended to include service configurations, endpoint information, and security profiles. A set of sample scripts are available to use as a starting point when modifying environmental parameters. With these scripts you can modify such parameters as port numbers and host names that differ from one environment to another.

Common architectural foundation

Foundation provides a common architectural foundation that enables a solution component to participate in processes. It provides a common invocation mechanism that ensures consistent access to components, services, and processes. This access is accomplished using the following methods:

- Java API
- Web services
- Watched folders
- Flex Remoting Service
- Representational State Transfer (REST)
- Email

  Foundation also provides a consistent set of public APIs and SPIs. Strongly-typed Java libraries are consistent regardless of the transport protocol used (RMI or SOAP).

Services can be programmatically invoked from client applications that are developed by using a Java integrated development environment (IDE). You can also invoke services from a Flex or Ajax RIA by using Data Services.

To develop a client application in a Java development environment, use Java APIs. LiveCycle also enables client applications to invoke its services by using web services:

Invocation API: A Java API that can be used to programmatically invoke any service. Use the Invocation API to invoke services, such as coordinate services that do not have strongly-typed APIs.

Strongly-typed Java API: A Java API that is used to invoke a specific service. A strongly-typed API is known as a *service client* and is used to invoke only a specific service. That is, you cannot use a service client that belongs to one service to invoke another service. These APIs can use RMI or SOAP as the communication protocol between the client and the LiveCycle server.

Web services: Services in the service container that can be configured to expose a web service, with full support for Web Services Definition Language (WSDL) generation. You can create a proxy library from any service's WSDL and, using the proxy library, you can invoke a service.

Watched folders: A service that can be invoked from a network folder that an administrator configured as a watched folder through the Administration Console. When a file is placed in the folder, a service operation that manipulates the file is invoked.

REST endpoints: LiveCycle creates REST endpoints. If you can design your short-lived orchestrations in such a way that all input is programmatically picked up by actions within the orchestration, and if the output is a single document, you can invoke the orchestration and get the output by using just a web browser.

Email: A service that can be invoked when a configured email account receives an email message, typically with a PDF document as an attachment. A LiveCycle administrator configures the email account details through the Administration Console. After LiveCycle performs the operation, it sends an email message to the recipient with a modified PDF document attached.

For more information about invoking services, see Programming with LiveCycle.

For information about enabling the invocation of processes, see the Installing LiveCycle Workbench 10

For more information about configuring watched folders and email accounts for invoking services, see the LiveCycle Administration Console Help.

User management and authentication

Foundation includes the User Manager component, which allows administrators to maintain a database for all users and groups. This database is synchronized with one or more third-party user directories. User Manager provides authentication, authorization, and user management for services.

User Manager enables Single Sign-On (SSO). SSO allows users to log in by using the Microsoft Windows® authentication mechanism without needing to authenticate. The users can use out-of-box LiveCycle user interfaces such as Workspace or Administration Console. SSO extends to PDF forms rendered within Adobe Reader that leverage web services for prepopulation, look-ups, and data submission. In these cases, the authentication token is leveraged to authenticate the web service call. The same pattern is applied to applications that are built using Flex and remoting to call services that are part of LiveCycle.

Foundation authenticates any user who works with LiveCycle. User Manager implements role-based access control so that administrators can associate users and groups with roles that are already part of the User Manager database. Role-based access control requires access rights to be assigned to roles instead of to individual users. Using the User Management pages in Administration Console, administrators assign appropriate roles to users or groups.

Administrators can create custom roles and associate them with existing permissions. They can also add new users to the database through the Administration Console.

Event management

LiveCycle provides the capability to create and receive business events that occur in processes. It stores events and delivers them to the interested parties as defined in the system through callbacks to event handlers.

LiveCycle supports several event types:

Asynchronous events: A business analyst or developer can define asynchronous events such as Cancel Order or New Order. These events can be linked to processes either to enable the initiation of a process or to handle a complex process flow within an existing process. A process can be initiated based on a New Order event and, while it is running, can receive a Cancel Order event that enables it to change its flow of execution based on its state. If the order was not fulfilled, the process could refund the customer. If the order was shipped, the process could send an invoice to the customer.

Exception events: Are usually defined by component developers and allow handling of errors during process execution. For example, the server is unavailable and triggers an alert to an administrator, or a transaction failure allows the developer to define a compensating transaction.

Timer events: Allow processes to wait a period of time or can be associated with activities so that a different process flow is executed if they are not completed on time.

Event filtering

Processes can look for events that contain specific data. For example, a process can send a document for signature to a customer and then wait for an event when the customer returns a signed document. Event filters can filter the event based on data such as the process ID or customer reference.

Dynamic groups

You can create dynamic groups that automatically include all users who meet a specified set of rules. In a dynamic group, you do not individually select the users who belong to the group. Instead, you specify a set of rules, and all users who adhere to those rules are automatically added to the dynamic group. You can create dynamic groups in either of these two ways:

- Enable the automatic creation of dynamic groups based on email domains, such as @adobe.com
- Create a dynamic group based on specified criteria, including the user's email domain, common name, canonical name, and domain name.

Delta directory synchronization

Delta directory synchronization improves the efficiency of directory synchronization. When delta directory synchronization is enabled, User Management synchronizes only users and groups that are added or updated since the last synchronization.

Improved certificate mapping

One of the steps required to enable certificate-based authentication for users is to create a certificate mapping. A *certificate mapping* defines a map between the attributes of a certificate and the attributes of users in a domain. If the content of these attributes differ, you can use a Java Regular Expression (regex) to match the attributes.

Administration Console

The Administration Console gives administrators access to tools so that they can perform these tasks such as:

- Configure and manage users, groups, and server authentication settings through the User Management pages
- Create and manage invocation endpoints and deploy LCA files without the need for developer tools
- Set up watched folders and email providers for non-programmatic process invocation
- Administer module properties and server settings such as port numbers and log files

For more information, see the documentation available for the LiveCycle Administration Console.

Launch Pad

Launch Pad is an Adobe AIR® client application that provides access from your desktop to the services on your LiveCycle server. Using Launch Pad, you can accomplish these tasks:

- Convert files to PDF documents
- Export PDF documents to other file formats
- Rights-protect documents
- Create PDF packages
- Create custom Launch Pad components efficiently invoke Orchestrations.
- Efficiently roll out service updates/additions/changes from the server, without administering any clients

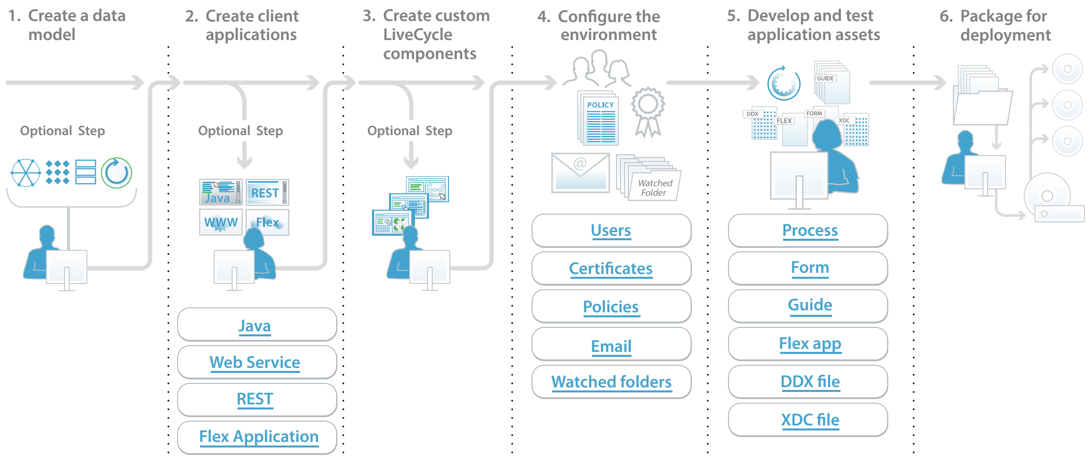
Chapter 4: LiveCycle Development Tools

LiveCycle provides a set of tools that developers can use to define, design, and develop applications. Workbench includes prebuilt components and other application building blocks so that developers, designers, and business analysts can work together to create user interfaces and process flows. Designer is the graphical form design tool that simplifies the creation of forms. The LiveCycle SDK provides a set of samples and tools that developers can use to build new LiveCycle applications or integrate with existing applications.

LiveCycle Application Development Workflow

This illustration shows an example of the application development workflow of a LiveCycle application. Some steps are optional, and the order of the steps is suggested.

LiveCycle Application Development Workflow



This diagram illustrates a possible workflow for developing LiveCycle applications. The order of the steps is only a suggestion. Also, applications do not typically require each step or all the listed items.

Model and create processes

Workbench is an integrated development environment (IDE) that process developers, form developers, and form authors use to model, create, or automate processes and forms. Workbench includes the tools necessary for designing Guides (deprecated), PDF forms, process flows, and output templates in a collaborative environment. The Eclipse-based development environment can also include Flash Builder, which developers can use to develop RIAs that facilitate end-user participation in business processes. With Flex components, developers can extend the functionality of Guides (deprecated) and customize Workspace.

***Note:** Flash Builder is a separate development tool which is licensed with Foundation.*

Developers use Workbench to manage the resources and services that processes and forms use. The forms can be rendered by using several client technologies such as PDF, HTML, and Flash. Business analysts and process developers use Workbench to design and deploy business processes that leverage the LiveCycle modules.

For file management in a distributed development environment, Workbench uses the concept of an application check-in and checkout. In LiveCycle, an *application* is a container for storing assets that are required for implementing a LiveCycle solution. Examples of assets are form designs, fragments, images, processes, event types, DDX files, Guides (deprecated), HTML pages, and SWF files. The permissions system is enhanced for application assets and usability changes to enhance developer productivity.

Forms and process flows are deployed to and managed by the LiveCycle server. Developers export forms, processes, and related assets such as images, DDX schemas, and XML schemas as a LiveCycle archive (DSA) file. The DSA is then deployed as an application to the LiveCycle server.

Design forms

Designer is the graphical form-design tool that simplifies the creation of forms. Form developers can use Designer to build and maintain data capture solutions that read from, validate against, and add to corporate data sources. Developers can integrate PDF documents into existing workflows by binding form designs to XML schemas, XML sample files, data models, databases, and web services. Designer can operate stand-alone and communicates with Workbench, enabling workflows that span both applications. Form developers can use the form creation wizard within Workbench or work directly from Designer.

Fragments

Fragments are reusable components of a form or document. Using fragments simplifies and speeds up the creation and maintenance of large numbers of forms. When creating a form, you insert a reference to the required fragment and the fragment appears in the form. The fragment reference contains a subform that points to the physical XDP file.

Action Builder

Action Builder is a tool in Designer that makes it easier to build actions in forms without requiring Java coding. Action Builder helps form authors build common interactions in forms without writing scripts.

Here are examples of common actions that form authors can build using Action Builder:

- Add buttons that a form filler can click to add or remove sections in the form or rows in a table
- Set the value of a field, such as prepopulating a date/time field object with the current date or a numeric field object with a specific value
- Set the background color of fields
- Hide or show objects or set the focus to a specific field
- Create custom actions by using scripting objects and the function within the scripts
- Build actions within fragments

Build components and client applications

The LiveCycle SDK provides developers with samples and tools for building components and client applications that programmatically interact with services in LiveCycle.

Programmer tools include reference and task-based documentation to enable programmers to use Java APIs, WSDLs, and LiveCycle Remoting to programmatically interact with services. Also included are complete Javadoc documentation and samples for developing custom services and applications, as well as customizing the Workspace and Guide (deprecated) interfaces.

For more information about developing client applications using APIs and LiveCycle Remoting, see Programming with LiveCycle.

Chapter 5: LiveCycle Modules

Enterprise Rich Internet Applications (RIAs)

Data Services

Data Services provides a comprehensive set of data-enabling features, which are deployed in a Java web application. Data Services also provides a separate set of features specifically for working with data in Foundation. These features are installed and deployed in Foundation.

With Data Services, you can build data-rich Flex and Ajax applications that securely and cooperatively interact with a variety of data sources. The data sources include web services, XML data, and custom Java destinations, including Data Management Service, Hibernate, and SQL.

Key features

Data Services ES3 provides the following set of services that you can use to securely access, transmit, display, and update data:

Message Service: Use a client-side API and a corresponding server-side Message Service to create Flex messaging applications.

RPC services: Access data through HTTP GET or POST (HTTP services), SOAP (web services), or Java objects (remote object services).

Data Management Service: Build applications that provide data synchronization, data replication, and occasionally connected application services. Data Management Service supports paged data, which lets you manage large result sets by returning subsets of data, called *pages*, to the Flex client application.

Conflict resolution: Data Management Service tracks the original state of the object so that it can accurately detect conflicts and provide an API for programmers to use to resolve them.

PDF creation: Build a Flex application that can generate a PDF document that includes static images or dynamic or static data.

Security: Restrict access to a privileged group of users by applying a security constraint in a destination definition.

Flex-Ajax Bridge: Expose a Flex application or component to scripting in a web browser.

Ajax Data Services: Lets Ajax developers access the messaging and data management capabilities of Data Services directly from JavaScript.

Offline data caching: Lets you cache client data requests and data changes to the local file system for later retrieval when an application resumes.

Real-time Quality of Service: Lets Flex clients select custom, data-access policies for real-time data.

Open adapter architecture: Data Management Service lets you retrieve data by using a variety of mechanisms. Data Services includes adapters for JMS, Hibernate, Java, SQL, and ActionScript.

Java 1.5 enumeration support: Enumerated types are serialized to string values in ActionScript by default.

The following features are integrated with Foundation:

LiveCycle Remoting: Invokes services through the Flex RemoteObject tag.

Proxy: Redirects service traffic from different domains. Provides client authentication, whitelists of permitted URLs, server-side logging, localization support, and centralized management of LiveCycle Remoting.

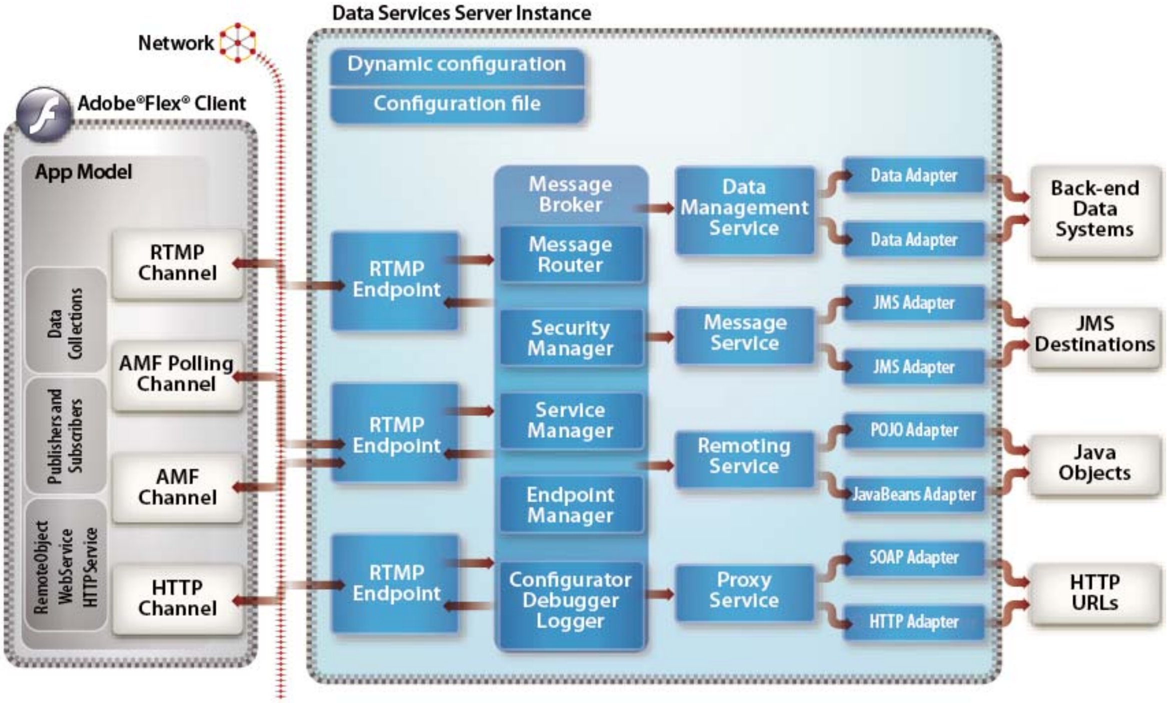
How Data Services works

Data Services is a J2EE web application that you can deploy on a variety of J2EE application servers and servlet containers.

Data Management Service, Message Service, and the Remote Procedure Call (RPC) services are all built on a core messaging infrastructure. Data Services uses the messaging infrastructure and the Action Message Format (AMF) protocol to ensure optimum performance. You can also use other protocols, such as HTTP and Real Time Messaging Protocol (RTMP).

Data Services uses XML configuration files to manage channels, destinations, message topics, message queues, and other settings. Data Services also provides run-time configuration. Server configuration can also be defined programmatically, which improves the ease of integrating existing code into a Data Services application.

The Flex framework provides MXML and ActionScript APIs that let you use Data Services in your applications. With Data Management Service, multiple clients are automatically synchronized when they retrieve data from the same destination. With Message Service, you can create applications that act as producers, consumers, or both. Data Services client applications use channels and destinations that are declared on the Data Services server instance. They can also use dynamic configuration to create destinations at run time.



Using Data Services

With Data Services, developers define a set of *destinations* using XML configuration files. These definitions are used by the built-in service adapters provided as part of the application. The following adapters can be used:

- Low-level adapters to connect to Java objects (data access objects), JMS topics and queues, and Adobe ColdFusion® components (CFCs)
- Higher-level adapters for common persistence solutions such as Hibernate, EJBs, and Spring.

The following sections highlight major services in Data Services.

LiveCycle Remoting

With LiveCycle Remoting integration, Flex clients can call services in Foundation. LiveCycle Remoting supports synchronous and asynchronous service invocation. LiveCycle performs much faster than web services by using the more efficient AMF3 binary protocol.

Message Service

This messaging feature is based on established publish and subscribe messaging standards and terminology. It provides a client-side API and a corresponding server-side Message Service for creating Flex messaging applications. The Message Service also enables participation in JMS messaging. The messaging and real-time infrastructure enables collaboration and data-push applications to be built in a scalable and reliable manner. This is done while preserving the lightweight web deployment model.

Data Management Service

The Data Services data management feature spans the client and server tiers to provide the top-level functionality for distributed data in Flex applications. With this feature, you can build applications that provide data synchronization, data replication, and occasionally connected application services. Additionally, you can manage large collections of data and nested data relationships, such as one-to-one and one-to-many relationships. You can also use Data Management Service adapters to integrate with data resources.

A client-side Data Services component, which you can create in MXML or ActionScript, calls methods on a destination configured in the server-side Data Management Service. The client-side component performs the following activities:

Filling client-side objects with data from remote data resources

Synchronizing the versions of data in multiple client instances with the server-side destination.

RPC services

RPC services let you interact with server-side RPC services to provide data to your applications.

You can access data through HTTP GET or POST (HTTP services), SOAP (web services), or Java objects (remote object services). Another common name for an HTTP service is a REST-style web service. REST stands for Representational State Transfer and is an architectural style for distributed hypermedia systems.

In a typical Flex application, an RPC component sends data as input to one or more RPC services. When an RPC service runs, it returns its results data to the RPC component that made the request.

LiveCycle Remoting is an extension of the Remoting RPC service.

Flex-Ajax Bridge

The Flex-Ajax Bridge (FABridge) is a small code library that you can insert into a Flex application to expose it to scripting in the browser. You can also use this library in a Flex component, or an empty SWF file.

In JavaScript, you can use FABridge to make your ActionScript classes available without any additional coding. This method is used instead of defining new, simplified APIs to expose a graph of ActionScript objects to JavaScript. After you insert the library, you can do anything with JavaScript that you can do with ActionScript.

Ajax Data Services

Ajax Data Services is a JavaScript library that lets Ajax developers access the messaging and data management capabilities of Data Services ES3 directly from JavaScript. Ajax Data services lets you use Flex clients and Ajax clients that share data in the same messaging application or distributed data application.

Many Ajax applications are taking on the responsibilities of round-tripping data. Using Ajax Data Services, you can integrate real-time data streaming and messaging, along with a rich data management API that dramatically simplifies the development of RIAs.

Chapter 6: Business Process Management

Process Management

Process Management lets users design, manage, monitor, and optimize human-centric processes. With Process Management, users can also manage automated business process applications that connect systems and people. Process Management enables the automation of human tasks and long-lived processes that involve asynchronous transactions.

Key features

Process Management offers the following features:

- Task assignment and management
- Workspace

Task assignment and management

Process Management provides task assignment and task management services that business analysts and developers use within Workbench.

Business analysts and developers can use the User service within Workbench to define task assignments in a long-lived process. Each task assignment defines the initial user, the form type that is routed to a user, task instructions, and rules for reminders, escalations and deadlines. Process Management supports several form types, including Acrobat forms and XDP forms rendered as PDF, HTML, and Guides (deprecated). Developers can also augment the User service with data mapping.

The User 2.0 service routes tasks to end users through their assigned queues. It coordinates tasks submitted through Workspace. The User 2.0 service can also route Flex applications that are built using Flash Builder and Data Services.

Process Management includes the following improved functionality:

- Ability to initiate a new process from a previously archived form
- New platform certifications

Business calendars

You can create business calendars and have processes escalate based on definitions. Tasks within business processes are usually created with predefined time for completion, reminders, and escalation rules. For example, a task must be completed within five business days and, if escalated to a manager, that manager must complete the task within two business days. Administrators can now create business calendars, configure non-working days, and map these calendars to specific people based on their location, business unit, and so on. When tasks are scheduled, these non-working dates are considered. You can configure multiple business calendars to meet your organizational needs. For example, you can configure calendars based on geographic area or for an organizational structure such as Customer Support.

Email notifications

Email notifications are a common mechanism for notifying users that a task is assigned, escalated, or must be completed. In LiveCycle, you can modify email notifications on a per-task basis, enabling more contextual content and flexibility. You can also create email templates on a per-task basis.

Out of office

A common problem occurs when tasks are assigned to users who are out of the office due to business travel or vacation. In LiveCycle, users can leverage Workspace to define when they are out of the office. Users can also assign tasks to either delegate to another person or remain in their task list.

Workspace

Workspace is an intuitive Flex-based application that lets end users initiate and participate in form-based business processes by using a web browser.

Using Workspace, users can perform these tasks:

- Fill forms that initiate a business process.
- Complete tasks by responding to email notifications that they receive for new tasks. Developers can include user actions in the email message as links. Users either click the action in the email message or reply with the action as the first line of the reply message.
- Open forms that were routed to a user or to a group queue to review, approve, or provide more information.
- When users open their task, they can add comments to it and view the comments that other reviewers added. Users can also see which action other users selected when they submitted their task.
- Add attachments and comments to a task and restrict access.
- Search for forms that are part of a completed business process or active processes that the user initiated or participated in.
- Have custom searches and filters based on process variables.
- View process categories and a list of tasks.
- Select processes and place them in a Favorites folder for easy access.
- Share tasks and consult with other users.
- Track processes and look at the audit trail.
- Initiate new processes from previously archived forms.
- Receive a request for Workspace electronic signature by confirming the validity of the information when completing a task. All the information required to determine whether the user confirmed the validity is stored as process data.

  Single Sign-on (SSO) for Windows lets users go to Workspace without having to authenticate, and 508 compliance lets users with disabilities use Workspace through screen readers such as JAWS.

  Workspace can also be viewed using the Safari browser or integrated as a portlet within a SharePoint portal.

  In the development environment, developers can use Flash Builder to customize the Workspace web interface that end users see. This lets your organization ensure that the Workspace user interface is branded appropriately and customized to meet your business requirements.

LiveCycle Mobile

Provides on-the-go access to your mobile-enabled forms, content, and tasks. Mobile keeps business workflows moving by allowing you to perform tasks and capture data from your mobile device. It can accept or reject travel expenses, document drafts - processes that often cannot wait for you to start your laptop. Data capture experiences allow users to not only enter text, but also snap pictures and add other content. Also, since mobile workers aren't always in a location where there's a connectivity, tasks can be approved, and data captured while the user is offline, and then queued up for submission back to the server when connectivity is available. It also enables Apple® iOS® devices to browse, open, and email files that are stored in your organization's WebDAV-enabled content management repository.

Users can download and install the application for Android™, Blackberry, and iOS devices to view and complete tasks, as well as capture data and view content while on-the-go.

Forms for mobile environment are created using Guide Builder, in Workbench.

Completion policies

You can complete an Assign Multiple Tasks operation before all the generated tasks are completed. This feature is useful when a decision can be made about a review without receiving a response from every reviewer. For example, the acceptance of a proposal requires a majority of approvals from committee members. You can complete the Assign Multiple Tasks operation immediately after more than 50% of the tasks are completed when the Approve action is selected.

Collection data and XPath functions

The information that is submitted for each task of an Assign Multiple Tasks operation is saved in a collection variable called *Task Result Collection*. XPath functions can be used to evaluate the results. For example, you can determine how many people selected a specific action or the percentage of people who submitted the action. These functions are useful when assessing the results of document reviews that occur in series.

Multiple user tasks

The User service provides the new Assign Multiple Tasks operation for assigning tasks to several users simultaneously. It allows process developers to construct parallel task assignments based on a list of users or a group.

The Assign Multiple Tasks operation is useful when you need several people to provide similar information. For example, at the end of each fiscal quarter, a process assigns a task to the vice president of each geographical sales group of your organization. To complete their task, each vice president attaches their quarterly sales report and then submits the task. The process retrieves each attachment from the collection of task results and sends them to the senior vice president of sales.

This operation is also useful when you require several people to review and approve the same information (document review-and-approval processes).

Services included with Process Management

Process Management includes these services:

- Complete Task
- Default Render
- Email Notification
- Form Augmenter
- Queue Sharing

- Render Guide (deprecated)
- Render HTML Form
- Render PDF Form
- Shared Tasks for Shared Queues
- Share Task Service
- Stall
- Submit Guide (deprecated)
- Submit HTML Form
- Submit PDSF Form
- User 2.0
- Wait Point
- Workspace Queue Sharing

For more information about the services included in this solution component, see Services Reference for LiveCycle .

Content Services (Deprecated)

***Note:** Adobe is migrating Adobe® LiveCycle® Content Services ES customers to the Content Repository built on the modern, modular CRX architecture, acquired during the Adobe acquisition of Day Software. The Content Repository is provided with LiveCycle Foundation and is available as of the LiveCycle ES3 release.*

Adobe LiveCycle Content Services (deprecated) is a module that offers content management services that let users design, manage, monitor, and optimize human-centric processes.

Content Services (deprecated) provides the following services to address organizational needs:

- Library services for checking in and out documents, versioning, auditing, and document cross-linking.
- Search services for advanced distributed searches; combined metadata content, location, object type, and multi-category search of content categorization.
- Automatic metadata extraction and categorization.
- Transformation services for transforming content from one format to another (for example, from Word to PDF).
- Records management for retention and archival policies; file plans to automatically classify and schedule records based on preexisting plans and standardized structures, and type-based plans that automatically classify and schedule records based on existing plans.
- Records management for automated life-cycle management; schedule, content, and metadata change activation based on simple rules.
- Automatic document numbering: DOD5015.2 Administrator Templates to support Department of Defense records and filing requirement for metadata definitions, file plans, and functionality.

Content Services (deprecated) provides several interfaces so that end users can interact with the repository:

- Adobe LiveCycle Contentspace (deprecated), which provides a web interface for shared content spaces, discussion groups, and searches. You can use Contentspace to conduct reviews for many types of content by making PDF documents available to others to review. Reviewers use the commenting and markup tools to add their comments to the PDF document. They can then publish their comments in a shared space where they can view and reply to comments from other reviewers.
- The two types of reviews you can conduct: on-demand reviews and structured reviews.
- Office plug-in for easy access to content from within Microsoft Office. The office plug-in lets you open, save, and edit documents that are stored in Contentspace from Word, Excel, and PowerPoint. You can browse content spaces and documents, view document details, start a workflow, view tasks, and search for and check out documents. You can download the plug-in from Contentspace at Company Home > Client Plugins > Content Services Plugins location.
- Shared folders for easily dragging and dropping content.

  To help customers develop their applications, Content Services (deprecated) includes samples that customers can use to quickly understand the component and develop an engagement application. Samples demonstrate how to access, render, and submit forms through Contentspace, the client web application. Other samples demonstrate how to protect content and invoke generic processes.

Key features

Content Services (deprecated) offers these features to address organizational needs:

- Process automation
- Information protection
- Document output

Process automation

Content Services (deprecated) can leverage the LiveCycle framework for business events by invoking processes based on content events, such as adding new documents to the repository. Or, a process can wait for content events to occur, such as a completed document review. For example, an insurance company can receive thousands of claims per day. When claims are converted to PDF documents, they are routed to a central repository and stored in folders according to the ZIP codes in the claims. The workflow is then triggered to route the claims to regional processing centers. In this example, the rules store the contents, and the content metadata was used to drive the process of sorting the claims by ZIP code.

With Content Services (deprecated), users can attach files, such as reports to a form, and submit the form. After the form is submitted, both the data and the attachments are saved, and can be opened and viewed in the same form layout. For example, a loan officer just received a credit report and employment report for a client. The loan officer opens the case form, attaches the reports to the form, and writes notes to indicate that the material is complete. The loan officer then submits the form. The form and the reports are then ready for the final approver to view in the same form layout.

You can also access different types of forms or content in the same portal. For example, an insurance agent can access both forms for home insurance and forms for car insurance. Because all forms have the same attributes, such as customer name, home address, and ZIP code, the insurance agent can search for any attribute. The insurance agent can also search car insurance forms by manufacturer and model. The agent can search home insurance forms by type of residence and number of rooms.

Using Content Services (deprecated), you can also search and browse by association between processes and documents, and by process and document attributes. For example, a court clerk wants to find all outstanding processes that are related to a case file.The clerk also wants to find all documents that are related to the case and all other cases that are pending from the same plaintiff.

Process initiation

Developers can invoke processes from Contentspace and leverage Workspace for reviewing and approving content.

Form rendering

Developers can prepopulate a form by using content or metadata. For example, car insurance buyers log into their accounts, open their renewal forms, and see that the form has current records, including claim history. When buyers click the claim history, they can see claim report files for each incidence they previously submitted.

Also, developers can render forms with attachments from Content Services (deprecated). They can update the form with a new version of assets referenced during run time, and keep form template versions consistent during the life cycle of the process. For example, a loan manager opens a case from a task list and checks attachments in the form, including an employment history and credit report. The loan manager also knows that the marketing team sent a letter that morning to announce the new company logo that is effective immediately. The manager finds the form and sees the new logo. The manager also receives email from the business unit management team announcing that a new form layout is available for new application cases.

Data storage

Content Services (deprecated) stores metadata in a database and content in a file system. Using a database immediately brings in the benefits of databases, such as transaction support, scaling, and administration capabilities. Content is stored in the file system to allow for large content, random access, streaming, and options for different storage devices.

Archiving

Administrators can archive content from completed processes in Content Services (deprecated). For example, an administrator can create a final package of documents that leverage the Assembler service or store multiple documents in the repository. In another scenario, a mortgage application is approved, and the final record is archived in PDF/A format. The record is then placed in a records management system with a proper archive policy.

Information protection

Content Services (deprecated) can leverage Rights Management to automatically apply policies to content that is stored in the repository. Using Rights Management, you can protect PDF documents. The policies are applied transparently to content when the following happens:

- Content is retrieved from the repository and saved on the local file system.
- Content is sent by email to someone who is external to the organization.
- Content is placed on devices such as USB keys.

Document output

Content Services (deprecated) has prebuilt actions so that developers can automatically convert documents to other formats (for example, from Word to PDF).

Additionally, developers can configure actions to start short-lived processes for the following purpose:

- Assembling a set of documents within a shared space into a package

- Generating PDF files from three-dimensional content
- Converting content to PDF/A format for archiving.

How Content Services captures content

In a typical process that uses Content Services (deprecated), a developer creates a LiveCycle application so that users can fill and submit data to a repository for another person to review.

How Content Services secures content

When users open downloaded documents, they are prompted to present their credentials. The Rights Management server enforces the latest policies associated with the document and users. For example, an automobile manufacturer wants to protect all engineering specifications with suppliers even when the specifications are viewed offline. The administrator can log in to the original manufacturer system and apply permissions and rights to the specifications. Permissions and rights determine whether the supplier can create, update, delete, copy, or print the documents. The permissions and rights are applied when suppliers open the documents from inside or outside the system, or when online or offline.

Services included with Content Services

Content Services (deprecated) includes these services:

- Document Management
- Node Service for managing metadata
- Search Service for performing queries

  For more information about the services included in this solution component, see.Services Reference for LiveCycle

Chapter 7: Forms automation

Forms

Forms enables organizations to extend their intelligent data capture processes by deploying electronic forms in PDF, HTML, and SWF format over the Internet. Forms also retrieves form data from central repositories and merges it with the specified form. With Forms, end users can access online forms without downloading additional software, fill the forms online, or save them to fill offline.

Key features

Forms offers the following key features:

- Renders PDF, HTML, or Guides (deprecated)
- Enables form data integration, which is the ability to import data into and extract data from PDF forms
- Includes Data Services support for Guide (deprecated) rendering
- Renders forms based on fragments
- Performs form assembly

Rendering forms and integrating form data

Using Forms you can render and process interactive forms and large data sets. You can create applications that perform interactive form-rendering operations such as these:

- Render interactive PDF documents. For forms created in Designer that have a flowable layout, Forms adds extra fields and boilerplate. These components are added as a result of merging the form design with data or as a result of scripting.
- Render interactive HTML forms. Forms automatically detects the browser type and platform, and then dynamically generates an HTML document that is based on a form design created in Designer.
- Swap the cascading style sheet (CSS) for the generated HTML form so that it can be controlled by the portal server. This feature also facilitates accessibility for HTML forms to emphasize particular styles that are more compliant to accessibility guidelines. Developers now control the form and field styles.
- Detect whether form design scripts should run on the client or on the server when rendering the form.
- Validate data entry by performing calculations, accessing databases, or enforcing business rules on field-level data, and then return the resulting data to the browser.
- Load XML data into an XML Data Package (XDP) file or into a PDF file that contains XDP information. Forms retrieves form data from central repositories and merges it with the specified form when rendering the form.
- Extract XML data from an XDP file. Forms can process form data that a user submitted. Form data can be submitted to an organization’s core systems, therefore increasing the quality of data gathered, improving customer service, and leveraging investment in core systems.
- With event-based caching invalidation, you can better control how the caching is managed for forms. For example, when only certain fragments are updated, the server can wipe only the affected forms that use that fragment.
- (* **New for 10** *) Create bookmarks in XFA forms. You can create bookmarks like those in PDF documents in XFA forms. You can create bookmarks for XFA form containers like subforms, fields, or a areas.

Developing Guides (Deprecated)

Guides (deprecated) can streamline and simplify the way that users fill a form. Form developers can quickly develop Guides (deprecated) in Workbench by using a data model and Guide Builder. After a Guide (deprecated) is created, it is rendered in the Forms Service API or the Render Guide (deprecated) service in Workbench (or a custom service). Developers can also use Flash Builder to create custom Guide (deprecated) components based on the Guide (deprecated) components that are included in Workbench.

Rendering forms based on fragments

Forms can render forms that are based on fragments that you create using Designer. A *fragment* is a reusable part of a form and is saved as a separate XDP file that can be inserted into multiple form designs. For example, a fragment can include an address block or legal text.

Using fragments simplifies and speeds up the creation and maintenance of large numbers of forms. When creating a new form, you insert a reference to the required fragment and the fragment appears in the form. The fragment reference contains a subform that points to the physical XDP file.

Here are advantages of using fragments:

Content reuse: You can reuse content in multiple form designs. When you need to use some of the same content in multiple forms, using a fragment is faster and simpler than copying or re-creating the content. Using fragments also ensures that the frequently used parts of a form design have consistent content and appearance in all the referencing forms.

Global updates: You can make global changes to multiple forms only once, in one file. You can change the fragment content, script objects, data bindings, layout, or styles, and all XDP forms that reference the fragment reflect the changes.

Shared form creation: You can share the creation of forms among several resources. Form developers with expertise in scripting or other Designer advanced features can develop and share fragments that take advantage of scripting and dynamic properties. Form designers can use those fragments to lay out form designs. This approach ensures that all parts of a form have a consistent appearance and functionality across multiple forms designed by multiple people.

Assembling PDF documents and forms

Forms can combine content from multiple PDF documents to create consistently formatted and seamless PDF documents.

You can use Forms for document assembly in the following types of workflows:

Customer communications: Automating the batch creation and assembly of customer invoices, statements, letters, form packages, insurance policy statements, marketing materials, and loan packages

Regulatory filings: Integrating document assembly into a regulatory document filing process

Archive preparation: Automating batch document customization for archiving, such as adding watermarks, or inserting or extracting metadata

Sales force automation: Preparing requests for quotes (RFQs) or generating proposals from multiple sources

Document conversion

The DocConverter service converts documents to PDF/A:

- Transforms PDF forms, XML forms (created in Designer), and PDF forms created in Acrobat to PDF/A-1b or PDF/A-2b
- Converts signed or unsigned PDF documents (Digital Signatures required)

- Validates the compliance of a PDF/A file and converts it if necessary
- The DocConverter service is supported on all standard LiveCycle platforms and does not require calls to Acrobat to convert or report on compliance.

How Forms works

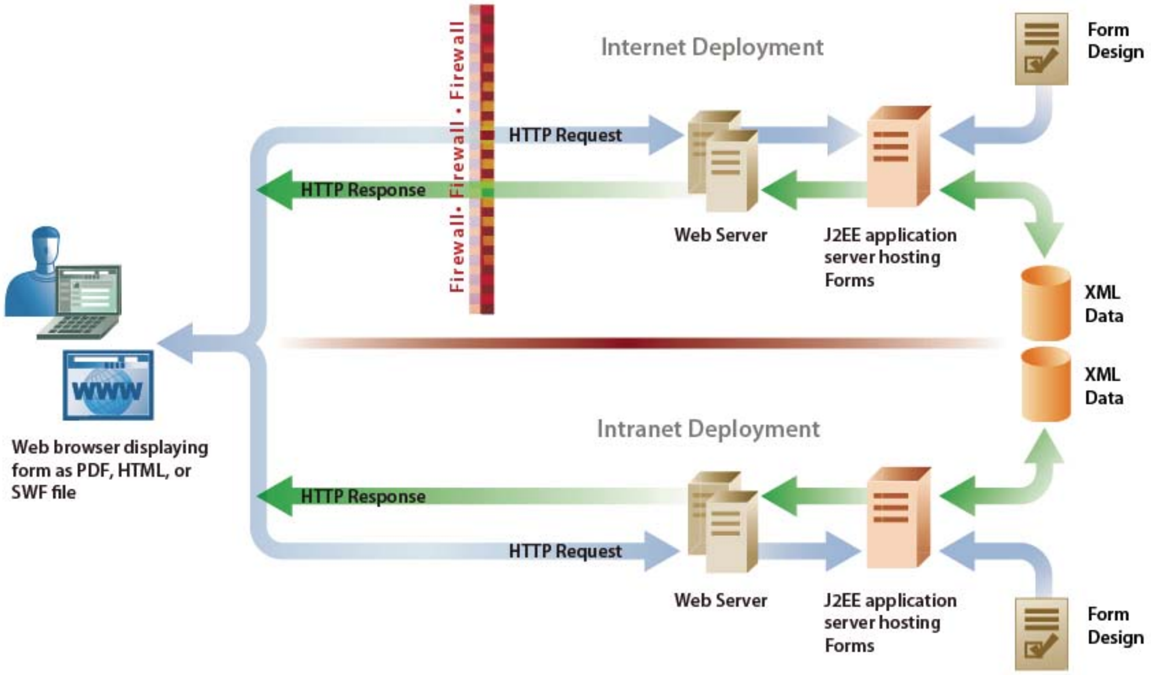
The forms for use with Forms are typically created in Designer . Forms also supports Acrobat forms. The form author can deploy the form designs for use with Forms either as XDP files or PDF files, depending on the requirements of the business process. Forms can render an XDP file as an HTML form or a PDF form. A PDF file is rendered as a PDF form.

The end-user environment consists of a web browser (for HTML forms) together with Adobe Reader (for PDF forms). Forms can detect the browser type and dynamically generate a PDF form or an HTML form based on the following input:

- Form design created in Designer
- Form preference that the developer identifies in the LiveCycle application.

When end users request a document from Forms, the request initiates a series of specific processes and interactions among the web application, Forms, and the web browser. The request can be initiated by clicking a button or an image on a web page, After receiving the form, end users can interact with it online. When end users are finished with the form, they submit it, along with form data, back to Forms.

The following illustration and the list below provide an example of how Forms processes a request from an end user.


Firewall- Firewall • Firewall
Internet Deployment
Form Design
HTTP Request
HTTP Response
Web Server
J2EE application server hosting Forms
XML Data
XML Data
Web browser displaying form as PDF, HTML, or SWF file
Intranet Deployment
HTTP Response
HTTP Request
Web Server
J2EE application server hosting Forms
Form Design

1 The end user accesses a web page and requests a form.

2 The web application invokes Forms and requests the form.

3 Forms retrieves the form design from a repository and data, and then merges the form design with the data to prepopulate parts of the form. The data can come from a variety of sources, such as an enterprise database, another form, or another application.

4 Forms determines the format to render the prepopulated form as based on the browser information that is passed with the call. The format of a form can also be set programmatically by using the Forms Service API.

5 Forms transforms the form design into PDF or HTML and then returns the prepopulated form to the end user.

6 The end user completes the form and then submits the form data back to Forms. Before form data is submitted back to Forms, applicable client-side scripts are run. For example, a user may be prompted to provide a value for a mandatory form field.

7 Forms extracts the submitted data, runs server-side scripts associated with the button that was clicked, and then executes the calculations and validations on the form.

8 Forms returns the results. If validations fail, the result may be a form that is returned to the end user. However, if validations are successful, the result may be XML data.

Services included with Forms

Forms includes these services that are part of LiveCycle:

- Assembler
- Encryption
- Forms
- Form Data Integration
- PDF Utilities

For more information about the services included in this solution component, see Services Reference for LiveCycle .

Reader Extensions

With Reader Extensions, your organization can extend the functionality of Adobe Reader by adding usage rights to the PDF documents it distributes. Reader Extensions lets your organization easily share interactive PDF documents with external parties by extending the functionality of Adobe Reader. Reader Extensions supports any PDF document that is rendered in Adobe Reader 8.0 and later.

Usage rights are permissions that Reader Extensions adds to a PDF document to activate features that are not usually available when the PDF document is opened using Adobe Reader. Adobe Reader users do not require additional software or plug-ins to work with PDF documents enabled by Reader Extensions.

With Reader Extensions, you can select usage rights for individual or batches of PDF documents by using the wizard-like, web-based application. Using the developer or programmer tools, you can assign usage rights to many documents in an automated batch process.

Key features

PDF documents that have usage rights added permit document recipients and users to do the following tasks:

- Complete PDF documents and forms online or offline, allowing recipients to save copies locally for their records.

- Save PDF documents to a local hard drive to retain the original document and additional comments, data, or attachments.
- Attach files and media clips to PDF documents.
- Sign, certify, and authenticate PDF documents by applying digital signatures using industry-standard public key infrastructure (PKI) technologies.
- Submit completed or annotated PDF documents electronically.
- Use PDF documents and forms as an intuitive development front end to internal databases and web services.
- Share PDF documents with others so that reviewers can add comments by using intuitive markup tools. For example, tools such as electronic sticky notes, stamps, highlights, and text strikethrough (the same functions that are available in Acrobat) can be used.
- Use barcoded forms decoding with the Barcoded Forms service.
- Use unlimited credentials.

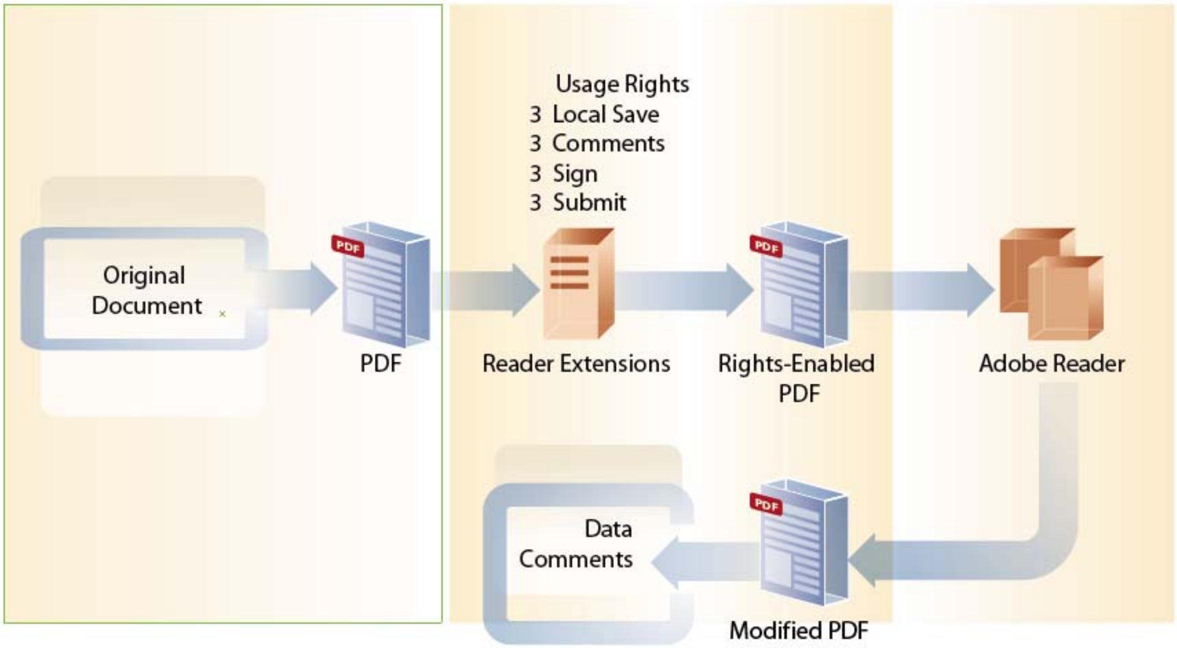
These special user capabilities are automatically activated when a rights-enabled PDF document is opened within Adobe Reader. When a user is finished working on a rights-enabled document, those functions are once again disabled in Adobe Reader. They remain disabled until the user receives another rights-enabled PDF document.

The specific usage rights that you can set are determined by a credential that you purchase from Adobe. Credentials typically provide permission to set a group of related usage rights, such as those pertaining to interactive forms. Each credential provides the right to create a specific number of rights-enabled documents. An evaluation credential provides permission to create an unlimited number of draft documents within a limited time period.

How Reader Extensions works

Most of the complexity of Reader Extensions is hidden from those who interact with the rights-enabled PDF documents or selects usage rights through the web-based application. However, before you implement LiveCycle, it is important to understand how Reader Extensions works.

The following illustration and list below provide an example of how Reader Extensions works:



1. A developer selects the usage rights by using any of these methods and uploads the document to the LiveCycle server:
   - Accesses the web-based application provided with Reader Extensions
   - Adds the Reader Extensions service in a process created in Reader Extensions
   - Uses the APIs provided with the LiveCycle SDK
2. Reader Extensions adds the selected usage rights to the document.
3. The rights-enabled PDF document is available.
4. When end users open the rights-enabled document using Adobe Reader, they can interact with the document according to the usage rights added to it.
5. End users can sign and submit the document electronically. OR, they can forward the document to others who can interact with the document according to the embedded usage rights.
6. If the PDF document is returned to the originating enterprise, the enterprise applications extract the updated information.

Two-dimensional barcodes

With Reader Extensions, you can add one-dimensional and two-dimensional barcodes to interactive PDF forms. You can then publish the barcoded forms to a website or distribute them by email or CD. When a user fills a barcoded form by using Adobe Reader or Acrobat, the barcode is updated automatically to encode the user-supplied form data. The user can submit the form electronically or print it on paper and submit it by mail or fax. You can later extract the user-supplied data as part of a LiveCycle process. This is accomplished by routing the data to the appropriate business processes based on the form type or the data itself.

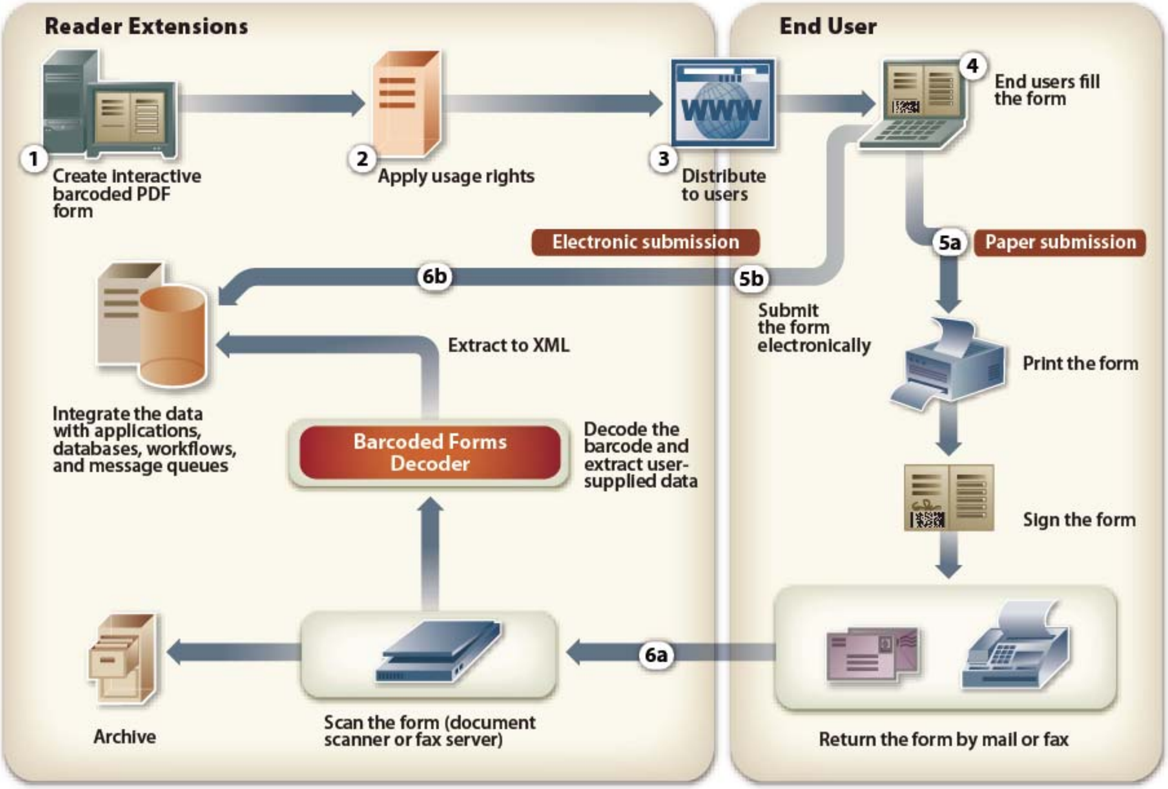
Barcoded forms can eliminate the need for optical character recognition (OCR)-based forms processing and the attendant costs of manual data entry. Data captured from fill-and-print barcoded forms can be reinserted into your electronic process quickly and automatically with 100% accuracy. Furthermore, you can retain a digital image of the submitted signed form for archiving purposes.

Key features

Reader Extensions 2D barcodes offer the following key features:

- Provides a unified approach for both paper and digital forms processing
- Automates the extraction and translation of barcoded data into core IT processes
- Supports barcodes encoded as XML, tab-delimited, or other user-defined formats

The following illustration and list below provides an example of how 2D barcodes work.


Reader Extensions
End User
1 Create interactive barcoded PDF form
2 Apply usage rights
3 Distribute to users
4 End users fill the form
Electronic submission
5a Paper submission
6b
5b Submit the form electronically
Extract to XML
Print the form
Integrate the data with applications, databases, workflows, and message queues
Barcoded Forms Decoder
Decode the barcode and extract user-supplied data
Sign the form
6a
Archive
Scan the form (document scanner or fax server)
Return the form by mail or fax

1 Your organization’s form author creates an interactive barcoded PDF form using Designer or Acrobat Professional.

2 Using the Reader Extensions web application, a user applies usage rights to the barcoded PDF form.

3 The user electronically publishes the barcoded form through the web, email, or as a CD.

4 The end user opens the barcoded PDF form in Adobe Reader or Acrobat and fills the form. As the user fills the form, the user’s data is automatically encoded in the barcode.

5 a) For a paper submission, the user prints and signs the form, and mails or faxes the form to your organization.

b) For an electronic submission, the user clicks a submit button to submit the form data electronically.

6 a) For a paper submission, when the completed form is received, your organization scans the form into an electronic image. The Barcoded Forms service locates the barcode on the scanned image, decodes it, and extracts the data into your specified format.

b) For an electronic submission through the Submit by Email button, the data, other than the barcode data, is directly submitted to the processing center as XML.

***Note:** The Barcoded Forms service can decode a PDF file that was saved in Acrobat when the file is directly submitted to the decoder in the same way a scanned TIFF file is submitted.*

Authoring barcoded forms

Form authors create the forms by using Designer or Acrobat Professional. In the authoring phase, the form author can specify any format to encode the data in the barcode, such as XML or tab-delimited characters.

In Designer, form authors create an interactive PDF form from scratch or by using a form template. Form authors can drag images and other objects, such as list boxes, text fields, command buttons, and barcodes onto the form. They can then resize and position the images and objects to suit your organization's requirements.

Designer provides more advanced features that let form authors use scripting objects, integrate a form with a data source, and create forms with a flowable layout. One advantage of authoring forms using Designer is that form authors are working directly in the form's source.

If the forms authored in Acrobat Professional have many custom scripts attached to the form objects, save time and effort by adding barcodes to the forms.

Creating a process

Developers can optionally create a process by using Workbench to include business processes specific to Reader Extensions. When integrated with other modules by using processes designed in Workbench, a single unified forms process can easily support different paper form submissions, each with their own specific workflow. (See also Installing LiveCycle Workbench 10)

Adding barcoded forms usage rights for Adobe Reader

Your organization must add barcoded forms usage rights to a PDF document before publishing the form to your customers. These usage rights activate the barcode data so that any commercial barcode decoder can read the barcode on the PDF form. Without extending the PDF form's usage rights with the barcoded forms usage right, the barcode will be illegible to all decoders, including those provided by Adobe.

In addition to the barcoded forms usage rights, the following functionality is enabled on the form:

- Saving completed or partially completed forms locally for offline filing and archiving
- Adding comments to and routing forms through email for third-party reviews
- Applying digital signatures to authorize applications or transactions
- Submitting form data electronically

Adobe Reader 8.0 and later does not require additional software or plug-ins to work with PDF documents enabled by Reader Extensions.

These special user capabilities are automatically activated when a rights-enabled PDF document is opened within Adobe Reader. When the user finishes working with a rights-enabled document, those functions are once again disabled in Adobe Reader. They remain disabled until the user receives another rights-enabled PDF document.

Usage rights are granted on a per-form basis or a per-document basis and do not apply to any other form or document. Adobe licenses the barcoded forms usage right based on the number of consumers of the form.

Updating barcodes during form completion

When a user fills a barcoded form electronically using Adobe Reader or Acrobat, the barcode is automatically updated with the user-supplied information.

Note: *If a user fills a barcoded form using an earlier version of Acrobat or Adobe Reader that does not support barcodes, a gray rectangle replaces the barcode. A gray rectangle indicates that the barcode cannot be updated. It also ensures that form processors do not process barcoded forms inadvertently in situations where the barcode does not accurately reflect the form's user-supplied data.*

Decoding barcodes to extract barcode data

The process at your forms processing center can affect your ability to successfully process and decode barcodes from barcoded forms. Key steps in processing barcoded forms include preparing documents, capturing data from barcodes, and routing captured data to enterprise systems.

The process of capturing data from a barcode varies depending on the type of device you use to process a barcoded form. You can select from the following options:

- Document scanner and the Barcoded Forms service
- Fax server and the Barcoded Forms service

The Barcoded Forms service locates the barcode on the scanned image (in TIFF or PDF), decodes it, and extracts the data in the specified format. The extracted data can then be used by another module such as Forms as part of a business process. For example, Forms can regenerate the original form automatically with the data the user entered or import the data into a blank form. This completes the digital-to-paper-to-digital cycle (round-trip).

Processing captured barcode data

Using the process you created, LiveCycle can automatically forward captured form data to the appropriate enterprise processing application. Because you can specify the data format in the authoring phase, moving form-based data across multiple enterprise applications is effortless. You can also archive data for visual presentation months or years later exactly as it was entered into the original PDF form.

Services included with Reader Extensions

Reader Extensions includes these services:

- Barcoded Forms
- Encryption
- Form Data Integration
- Reader Extensions
- PDF Utilities
- XMP Utilities

For more information about the services included in this solution component, see Services Reference for LiveCycle .

Chapter 8: Document information and security

Rights Management

Rights Management ensures that the business-critical information you distribute is exposed only to the intended people. You control how people can use the information to prevent it from circulating too far. Your control over the use of information continues even after you distribute the information.

Key features

Using Rights Management, you can protect PDF, Word, Excel, and PowerPoint documents by using confidentiality policies. You can restrict corporate training videos to the intended recipients. You can also provide new video workflows based on the same set of SDKs and APIs that are available through Rights Management.

The PTC Pro/Engineer WildFire 4 (Pro/E) product includes direct protection of their native CAD documents. By using Pro/E along with this version of Rights Management, you can distribute product manufacturing instructions and CAD materials. These documents will have the same level of protection and value proposition as securing PDF files with Acrobat and Rights Management. This partnership release enables secure collaboration and versioning during the product design phase. It also simplifies collaboration with external parties during a bidding, Request for Quotation (RFQ), or manufacturing phase.

A *policy* is a collection of information that includes document confidentiality settings and a list of authorized users. The confidentiality settings you specify in a policy determine how a recipient can use documents that you apply the policy to. Because PDF documents can contain any type of information, such as text, audio, and video files, you can use Rights Management to safely distribute information saved in a PDF document.

You can use policies to do these tasks:

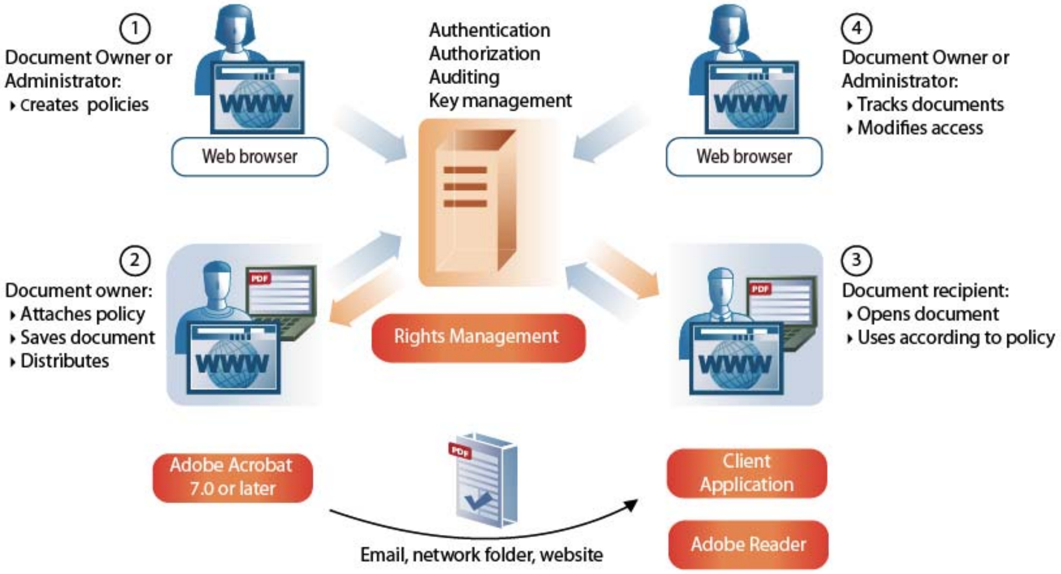
- Specify who can open policy-protected documents. Recipients can belong to your organization or can be external to your organization. You can also specify different confidentiality options on the same policy for different users.
- Specify the document confidentiality settings. You can restrict access to various Acrobat and Adobe Reader features. These restriction may apply to the following rights:
  - The right to print and copy text
  - The right to make changes
  - The right to add signatures and comments to a document.
- Administrators can also specify the following additional confidentiality options:
  - The recipient’s right to view a document offline
  - The right of the user who applies the policy to revoke the document access rights or switch the policy.
- After distributing a policy-protected document, you can monitor and revoke document access, switch the policy, and change access and confidentiality settings. Users can change confidentiality settings in policies they create. Administrators can change any organizational or user-created policy.

- Define dynamic watermarks that are applied to documents protected with the policy. These watermarks can contain multiple elements, each consisting of text or PDF. You can specify their positioning and appearance on documents applied with the policy. With the use of variables, you can have dynamically created watermark content that includes the user name, applied policy, and timestamp.
- Enable or disable Extended Usage Tracking. Rights Management supports tracking of user events associated with various operations performed on a PDF file. The Rights Management object can be accessed using a JavaScript. A button click, a multimedia file being played, or saving of a file are some examples of events that can fired from a policy protected PDF. Using the Rights Management object, you can also retrieve user information. Tracking of events may be enabled from the Rights Management server at the global level or at a policy level.

Using Rights Management, users can open and use protected documents when they are not connected to the Rights Management server. The user's client application must regularly synchronize with the server to keep documents valid for offline use. By default, synchronization occurs automatically every four hours and as required when a user is connected to the Rights Management server. If the offline period for a document expires while the user is offline, the user must reconnect to the server. Reconnecting enables the client application to synchronize with the server. In the Rights Management configuration file, you can specify the default frequency of the automatic background synchronization. This setting acts as the default time-out period for client applications unless the client explicitly sets its own time-out value.

How Rights Management secures a document

The following illustration and list shows various users interacting with policy-protected PDF documents and Rights Management. (If Rights Management protects other file types, such as a DOC file, users can interact with these file types in the same manner.)



Here is an example of how Rights Management works:

1. The document owner or administrator creates policies by using the Rights Management web application accessible through Administration Console. Document owners can create user policies accessible only to them. Administrators can create organizational policies within policy sets that are accessible to permitted users, and can also designate policy set coordinators. The policies are stored in a database that connects to the application server.

   Developers can also automate the creation of policies by using the Process Design perspective in Workbench or the Rights Management API.

2. The document owner applies the policy, and saves and distributes the document by using the web pages or Adobe Acrobat 7.0 or later. The document can be distributed by email, through a network folder, or on a website.

   Developers can also automate the application of policies to documents. They can also automate the distribution of these documents to end users by using the Process Design perspective in Workbench or the Rights Management API.

3. The document recipient opens the document in Acrobat 7.0 or later or Adobe Reader 7.0 or later. The recipient can use the document according to its policy.

4. The document owner, policy set coordinator, or administrator can track documents and modify access to them by using the web pages. Developers can also track documents by using the Process Design perspective in Workbench or the Rights Management API.

Programmatically applying policies

On a mass production environment such as generating monthly invoices for a telecom company, creating and applying policies that are specific to each document can become a resource-intensive process. In such cases, you can use the Rights Management Java API to create and apply policies that are specific to users, rather than to documents based on abstract policies. The license generated for a user is later used for all documents that are accessible to the user.

Using the APIs, you create an abstract policy that is a policy templates with all policy attributes such as document security settings and usage rights, except the list of principals. Administrators can create any number of policies from the abstract policy with different principals who should have access to the documents. Changes made to the abstract policy do not affect the actual policies that are generated from the abstract policies.

In the case of monthly invoice generation of a telecom company, you create an abstract policy, users, and then generate the licenses for each user that is later applied to the documents for each user.

You cannot create the abstract policy from the Rights Management Web pages. You can, however, administer the policies that you create from the abstract policy from the Rights Management web pages. Policies that are created using this method are identical in behavior to those created from Rights Management web pages.

See Programming with LiveCycle for more information.

Rights Management security

To ensure the confidentiality of documents that are protected by policies, Rights Management implements three layers of security:

- Authentication
- Authorization
- Document confidentiality

Authentication

All users are required to log in to interact with Rights Management. Users can log in through Acrobat or through the Rights Management web application.

Rights Management supports four methods of authentication:

- Username/Password. Users are prompted for their user name and password.
- Kerberos (from Acrobat on Windows only). Users of Acrobat or Adobe Reader for Windows can be transparently authenticated.
- Client certificate based. Users can use their certificates installed in client machines to authenticate themselves.
- Extended authentication. If available, users can be authenticated using any extended authentication providers.

After users are initially authenticated and Rights Management receives subsequent messages from clients, it uses Security Assertion Markup Language (SAML) authentication assertions to verify the identity of the message sender.

Authorization

Rights Management uses a role-based model to control access to the web application features. Roles also determine whether users can protect documents with policies through Acrobat. Rights Management implements these roles:

Administrators: Have complete access to the server configuration and can manage all aspects of policies, policy-protected documents, external users, administrator accounts, and event audits.

Users: Can create and manage their own policies, policy-protected documents that they distributed, and events that are associated with those documents.

External users: Can create a Rights Management user account when an administrator explicitly invites them or when they are added to a policy.

Document confidentiality

Rights Management uses several technologies to protect documents and to provide access to them.

In general, Rights Management uses a symmetric cryptographic key system for encryption. Client applications such as Acrobat perform document encryption. Documents are never sent to Rights Management; Rights Management encrypts policies and licenses that are associated with documents.

The method used to protect documents depends on whether the policy requires users to access documents while online or whether the policy enables offline use.

Services included with Rights Management

Rights Management includes these services:

- Rights Management
- Encryption
- PDF Utilities
- XMP Utilities

For more information about the services included in this solution component, see Services Reference for LiveCycle .

Digital Signatures

Digital Signatures lets you use digital signatures to preserve and verify the integrity and authenticity of a document in these situations:

- When it is transferred among users within and beyond the firewall
- When it is downloaded offline
- When it is submitted back to your organization.

With Digital Signatures, you can automate the process of certifying and signing documents. You can also automate the process of validating signatures in documents that are submitted back to your organization.

Key features

Digital Signatures can apply security features to any PDF document whether it is generated by other Adobe server products, on a desktop by Acrobat, or even a third-party solution. Because PDF documents can contain any type of information, such as text, audio, and video files, you can use Digital Signatures to secure any type of information saved in a PDF document.

Digital Signatures can apply the following security features through automated business processes or programmatically through the API:

Certification and Approval signatures: Specify digital signing of documents so that recipients can validate the authenticity and integrity of the content. Digital signatures can be applied individually or in batches by using digital certificates from third-party vendors. With digital signatures applied, documents maintain authenticity even when archived.

Signature validation: Specify signature validation so that your organization can verify the authenticity of documents it receives.The Signature service verifies signatures on a PDF document and returns information about the overall validity of a PDF document. The validity of a PDF document includes the signed content, and the identity and trust settings of the signer. Document validity determines whether Modification Detection and Prevention (MDP) and Modification Detection and Protection Plus (MDP+) rules are adhered to.

Dynamic control: Design forms in such a way that you can control the signature properties dynamically on the client for fields that were already rendered. You can also add new fields and additional control.

How Digital Signatures secures a document

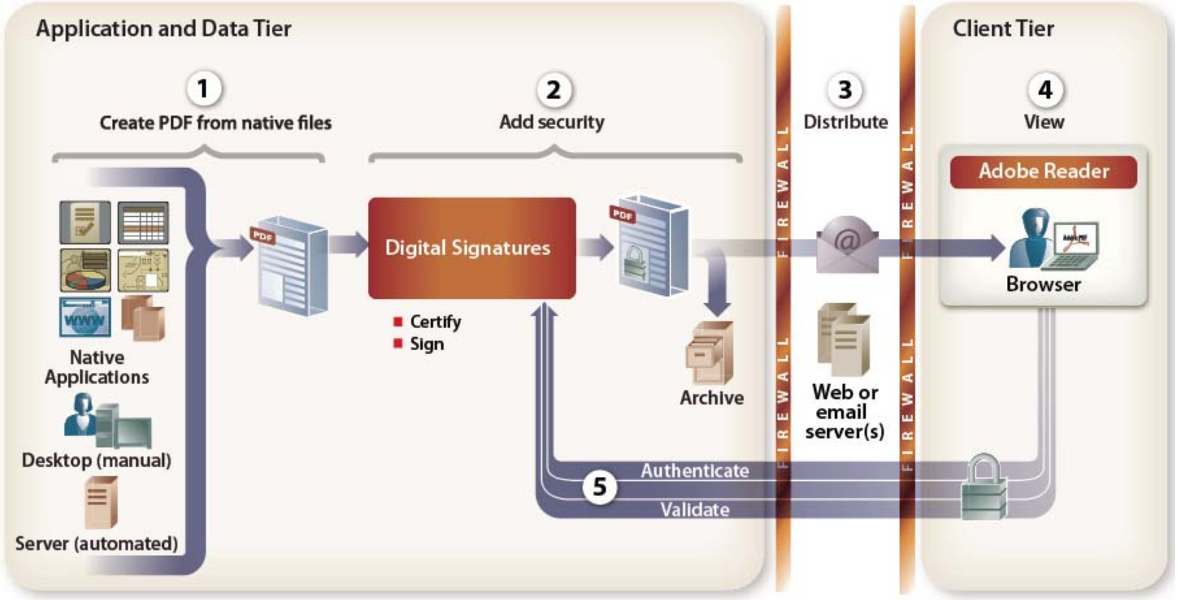
In a typical Digital Signatures process, an application secures the document using the following steps:

- Retrieves a PDF document from a specified repository
- Applies a digital signature by using a credential (private key) in a specified keystore
- Passes a document back to the process for archiving, delivery, and other actions, as appropriate.

In another example, a custom application created by using the Java API uses these steps:

- Gets a series of documents
- Applies a digital signature to all of them, and passes a document back to the process for archiving, delivery, and other actions, as appropriate.

The following illustration shows an example of a typical Digital Signatures process.



1 PDF files are created using any type of PDF production method:

- Automatically from a server using Forms,
- Manually on a desktop using Acrobat
- Using the PDF creation capabilities of third-party applications.

2 A client application retrieves a PDF document from a specified repository. The PDF is passed to Digital Signatures.

3 Digital Signatures secures the PDF document by certifying or signing it. It can archive the document in its secure state, as well as pass it to a web or email server for distribution.

4 A web server or email server distributes the secure document by posting it on a website or sending it by email to recipients.

5 The recipient uses a public key (digital certificate) to validate the signature. If required, the recipient can add information to the PDF document. If the document is a form, the recipient can fill it and then resign it for submission to the sender.

6 Digital Signatures receives the submitted PDF document and validates the signature to ensure the integrity and authenticity of the document.

Services included with Digital Signatures

Digital Signatures includes these services:

- Signature
- PDF Utilities
- XMP Utilities

For more information about the services included in this solution component, see Services Reference for LiveCycle .

Encryption

You can encrypt a PDF document with a password or a certificate that renders the document unreadable to anyone who does not have the password or a private key to open the file. The private key needed to open a certificate-encrypted file, must correspond to the public key used to encrypt the document.

XML Encryption and Decryption

*** New for 10 ***

You can use the encryption service to protect XML data in XFA forms. You can encrypt the entire XML from the root node, or provide an XPath expression that points to the exact node or tag to be encrypted. To decrypt the document, the user must have a private key that can be used to unlock the content in a PDF document. You must know the user credential of the person for whom the XML is encrypted, and must provide the same credentials to the Encryption service. The Encryption services looks up the Truststore, where user credentials are private keys are paired and stored.

Chapter 9: Communications management

Output

Output provides support for a variety of output formats and supports the output design features provided in the Form Design perspective in Workbench or Designer.

Key features

Using Output, you can create applications with the following functionality:

- Generate final form documents by populating Designer files with XML data.
- Output forms in a range of formats, including non-interactive PDF, PDF/A, PostScript, PCL, and label printer print streams.
- Control laser printer features, select paper trays, do duplex printing, and staple printed documents.
- Process single documents, document packages, and document batches from single or multiple Designer ES3 templates that may use fragments.
- Assemble PDF documents and convert PDF documents from native documents.
- Dynamically assemble XDP files and place fragments at insertion points in XDP files for demanding document assembly tasks.
- Convert PDF packages that may contain any combination of PDF forms and documents to PostScript.
- Edit and create XDC files in the repository and on a file system using the device profile editor.
- Convert PDF documents to PDF/A-1b and PDF/A-2b using the DocConverter service.
- (* **New for 10** *) Flatten interactive forms, without losing the state of the form (retain modifications to an interactive form that have been effected by user-driven events).
- Central Migration Bridge (deprecated) service permits the use of form templates, data, and data transformations from Adobe Central Output server in the LiveCycle context. You must have Central 5.7 installed on the same server as LiveCycle. Central Migration Bridge (deprecated) is available to existing Central customers to make the migration process easier to stage in phases and complete.

  Note: *The Central Migration Bridge service, introduced in LiveCycle ES2 to assist Adobe Central Pro Output Server customers in migrating in a staged manner to the Output service, continues to be available in this release, however customers should be aware that it will be removed in the next major release.*

Document design for Output

Designer form design features include tools for designing fixed and flowable document layouts for Output. The layout and output choices you make when you design forms and documents are implemented and performed by Output.

Using Designer, form authors can use a variety of layout specifications and features in form designs and templates:

- Create forms whose layouts adjust when merged with XML data. This ensures that rendered (or printed) forms accommodate the merged data content and volume by including or excluding design elements, growing to create space for data, and paginating automatically.
- Define duplex formatting and the media (paper tray) in a form template to logically link document layout to printer capabilities.

- Access fragments in the repository and incorporate them into form templates for greater control and management of document designs.
- Spell-check documents and generate sample test data while designing forms. Using the sample test data, you can then generate a PDF preview or print directly to a target printer.

These features were designed for use with Output. Forms and documents created with these layout features can be rendered or printed in a variety of formats without compromising the design intent.

Supported document formats

Using Output, processes can merge XML data with forms or documents created using Designer to produce documents in a variety of formats:

- PDF final form documents for viewing and printing in Adobe Reader or Acrobat
- PDF archive documents in PDF/A-1b and PDF/A-2b
- PostScript and PCL documents for direct printing to monochrome and color laser printers
- ZPL for direct printing of documents to Zebra label printers
- IPL for direct printing of documents to Intermec label printers
- DPL for direct printing of documents to Datamax label printers
- TPCL for direct printing of documents to Tec Toshiba label printers

Using Output, you can also generate documents in a wide range of languages, including those that use Western and Eastern European, Asian, and Middle Eastern character sets.

Controlling print features

Using Output, you can leverage and control the specific capabilities of the printers you use to generate printed documents. Using XDC Editor, you can edit existing XML Forms Architecture Device Configuration (XDC files), create new ones, and map media to specific paper trays.

You can generate documents that access the following features of laser printers:

- Duplex printing
- Specifying which input and output paper trays to use
- Resident fonts
- Stapling and copying.

You can generate PDF documents that automatically open a print dialog box in Adobe Reader or Acrobat. You can also generate PDF documents that are preconfigured to specify duplex printing and the number of copies to be printed on a default or specified printer.

Generating flexible documents

Output provides flexible document generation. You can generate the following types of document:

- A single document from a specific Designer template
- A batch of documents from a single template
- A batch of documents, each from separate files by using a single Designer template.

Using this flexible generation capability, you can also leverage fragments in document output. You can generate a document package that uses multiple Designer templates. For example, a mortgage application can contain numerous sections. Some of them are customized for the client by using specific templates for interactive forms and fragments. Others are standardized by using dynamic PDF documents. The final generated output is a single cohesive document.

With Output, you can also produce companion metadata files for document integration and statistics. For example, metadata from incoming XML such as submitted fax numbers or account numbers can be integrated into forms for output. Output also supports metadata that is used for generating documents, such as page numbering and the specific documents included in a batch.

Document conversion and assembly

Output supports document generation beyond simple print control and PDF rendering. It also supports PDF document assembly and conversion. For example, you can convert PDF documents to TIFF files for archiving or to PostScript files for direct server-based printing. You can convert interactive PDF forms (fillable forms) to final version PDF forms (filled information is frozen, and the document is no longer interactive).

The DocConverter service converts documents to PDF/A:

- Transforms PDF forms, XML forms (created in Designer), and PDF forms created in Acrobat to PDF/A-1b and PDF/A-2b.
- Converts signed or unsigned PDF documents (Digital Signatures required).
- Validates the compliance of a PDF/A file and converts it if necessary.

LiveCycle fully supports the DocConverter service and does not require call to Acrobat to convert or report on compliance.

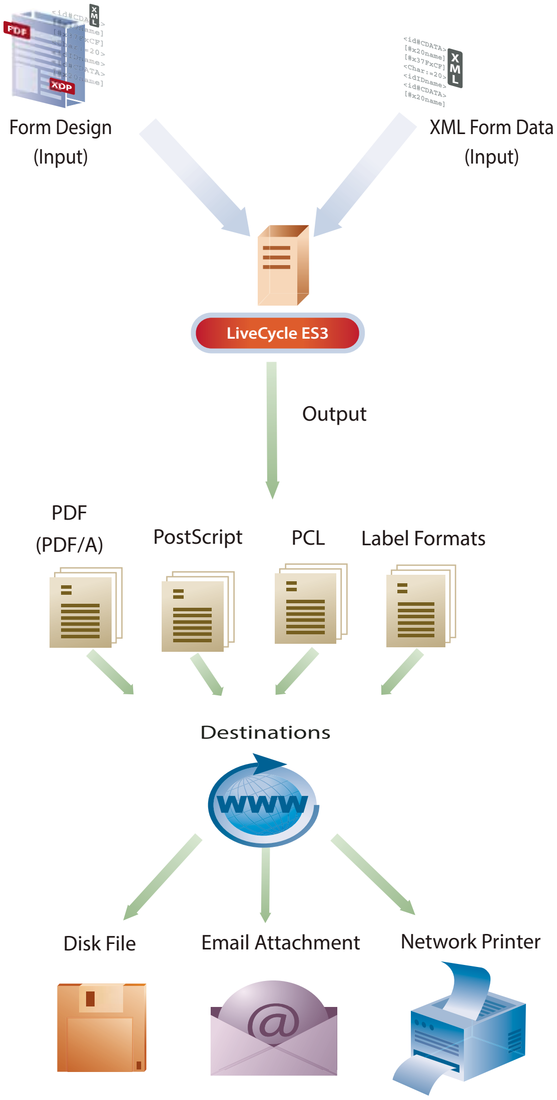
You can use Output to manipulate and assemble the multiple source PDFs into a single seamless document.

Document and process automation

You can develop processes that automate generating documents that include barcodes that contain variable data or print directly to label printers. Barcode data can be a range of standard codes, including 2D symbologies such as PDF417, Datamatrix, or QR. When you output to a Zebra label printer, you can use standard or RFID labels. When you output to any of the label formats, you utilize accurate resident barcodes and efficient resident fonts.

How Output works

Here is an example of how Output works.



A form author creates the form in Designer. The form conforms to the XML schema that the Output service uses as the XML data input file to create the output.

1 The developer identifies the form, XML data input file, and output format that the Output service uses to create the output. The developer uses one of these methods to identify the form, input, and output:

- Adds the Output service in a process created in Workbench
- Uses the APIs provided with the LiveCycle SDK

2 The developer deploys the application to the LiveCycle server.

3 The Output service is invoked. An XML input file is provided to the Output service. The service extracts the form design from the repository and merges the XML data input file with the form design to create the output. The output is either a print stream (PostScript, PCL, or label format) or a PDF document.

4 The output is sent to the designated destination as an email attachment (using the Email service as part of a process). It is then sent to a network printer, or is saved as a file on disk.

Services included with Output

Output includes these services that are part of LiveCycle:

- Assembler
- Central Migration Bridge (deprecated)
- Convert PDF
- DocConverter
- Encryption
- Output
- PDF Utilities
- Print PDF Package
- XMP Utilities

For more information about the services included in this solution component, see Services Reference for LiveCycle .

Production Print

Production Print is a dedicated, high-performance document composition run-time and development environment. It merges XML, ASCII, and other data types with Designer forms to generate personalized documents in a range of print and electronic formats, including Advanced Function Printing (AFP) and IJPDS, to support high-volume production and enveloping requirements.

Key features

Development and Run-time environment

Production Print runs in a dedicated environment; it is not a standard LiveCycle module that depends on Workbench and Foundation, however, it provides integration points to both.

- Dedicated design environment (Design Center) and run-time environment to perform high-volume document composition and finishing using Designer forms.
- Design center hosts Designer; supports fragments and is integrated with the LiveCycle repository.
- Control Center for managing one or more run-time servers.
- Production Print can be invoked via a LiveCycle process using Workbench services supplied with the module.
- Production Print can invoke any LiveCycle process from within its run-time environment.

Using Production Print, you can create applications with the following functionality:

- Format and render Designer forms with XML, ASCII and other data formats.
- Collect, split, sort and group formatted documents using a post-processing repository.
- Generate a variety of output formats including, PCL, PostScript, PDF, PDF/A, AFP and IJPDS.
- Apply finishing processes for 2-up printing, marking documents for automatic insertion, and enveloping.

- Convert AFP files to PDF files.

How Production Print works

1 Within Design Center a new project is designed addressing all of the steps required for a complete process.

2 The developer identifies the data source, XML data, ASCII data, and SAP formats.

3 The developer designs a form using Designer that can be launched within Design Center or the developer associates a form with the process for the data source.

4 Forms can be stored in and accesses from the LiveCycle repository.

5 The developer then establishes the further stages of the process, sorting documents in the post processing repository, rendering to final format (AFP), applying marks (OMR or barcode including 2D Datamatrix) to drive insert, and enveloping machines.

6 The project is saved and put into production using Control Center.

7 In Workbench, a process is designed using the Production Print service to invoke the process designed in Design Center.

8 At run-time, LiveCycle passes a data file to the Production Print run-time. The data file is processed, generating an AFP print file that contains OMR marks for inserts and automatic enveloping.

9 The file is submitted to an AFP roll-feed printer that prints images 2 beside each other.

10 The printed roll is fed through a machine that cuts the roll to sheets.

11 The printed sheets are passed to the insertion machine and processed. Thousands of envelopes are generated and placed into the mail-stream for distribution.

Services included with Production Print

Production Print includes these services that are part of LiveCycle:

- Production Print

  ***Note:** Note that Production Print is not part of the standard LiveCycle media; it has its own media and installation process.*

PDF Generator

Using PDF Generator, you can generate PDF documents from many different file formats. The following file formats are supported:

- Native file formats such as Word, Excel, PowerPoint, Visio, AutoCAD, and Corel WordPerfect
- Open standards such as TIFF image files, PostScript, Open Document Format (ODF), and HTML.

PDF Generator supports output to PostScript and legacy TIFF archival systems, or it can generate searchable PDF documents from image sources.

Using PDF Generator, you can also assemble single PDF documents from several source documents or separate PDF documents into constituent pages. PDF document assembly supports these publishing features:

- Repagination
- Resizing

- Addition of headers and footers
- Insertion or deletion of pages
- Creation of a table of contents.

The included services provide a flexible mechanism for extending the range of the file formats that PDF Generator supports. Also, they add support for converting almost any type of document to PDF.

Key features

PDF Generator provides the following features:

- Converts many native file format to PDF.
- Converts PDF files to many file formats.
- Supports assembling documents, attachments, and groups of documents into PDF packages by using the workflow and job ticketing language.
- Supports creation of industry standard PDF/X-1A, PDF/A-1b, and PDF/A-2b from native file formats and PostScript.
- APIs that provide the ability to add file formats to existing conversion support.
- Multi-threaded conversion of Word, PowerPoint, and OpenOffice file formats.
- Fallback options if a problem occurs with conversion.
- HTML conversion produces PDF files that more closely resemble the original HTML. PDF Generator also provides options for configuring the appearance of the PDF that the HTML-to-PDF conversion produced.
- Can optimize the size of the PDF files it produces.
- Improves performance when converting non-Adobe documents to PDF. Performance improvements vary depending on the document and how it is being converted. However, initial testing shows significant improvement in performance when converting non-Adobe documents to tagged PDF.
- Provides system-readiness tools that can verify that native applications and account information is set up correctly.
- Allows non-administrative users to submit conversion jobs through a LiveCycle web page.
- Sets the security level for PDF files produced.
- Provides control over Adobe PDF settings. Examples of such settings include whether fonts are embedded, image compression, and color conversion.

  ***Note:** Adobe PDF settings were previously known as Distiller® parameters or job options.*

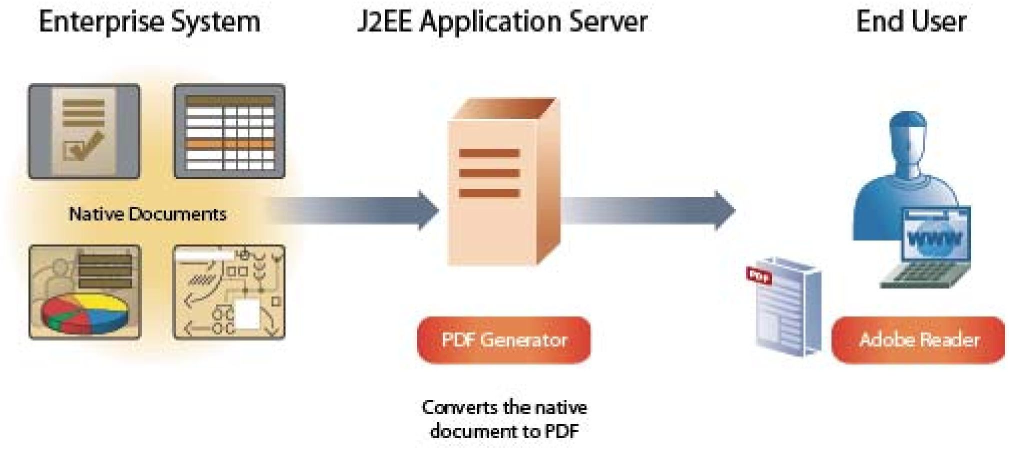
Using PDF Generator in the context of LiveCycle development tools and additional modules, you can perform the following tasks:

- Assemble content from diverse sources and file formats into a single PDF document that includes cohesive pagination, indexing, and a table of contents. You can also preserve, import, or export existing content such as annotations, file attachments, and bookmarks.
- Secure assembled PDF documents with document policies or passwords by using Rights Management or Digital Signatures.
- Route-assembled and secured PDF documents to stakeholders for approval and resubmission.
- Embed PDF documents in automated enterprise document workflows that store information in an ECM system or archive solution. For example, you can use PDF Generator to convert customer letters to PDF before you store them in a document archive system.

- Provide enterprise end users access to a centralized service for PDF creation, through email or web-based interfaces, or drop (watched) folders.

How PDF Generator works

Here is an example of how PDF Generator works.



1. An administrator accesses Administration Console to customize the default PDF settings, security settings, and file type settings that control how PDF Generator converts documents. The administrator also configures PDF Generator to periodically scan a folder, called a *watched folder*, for files to be converted.
2. A client application places a native file for conversion to PDF in the watched folder.
3. LiveCycle periodically scans the watched folder for files to be converted at intervals and specified by the administrator when the watched folder was configured. LiveCycle invokes PDF Generator and provides the file for conversion.
4. PDF Generator uses the default settings that the administrator specified in Administration Console to convert the native file to PDF and moves the resulting PDF document to the Out folder.

Document conversion to PDF/A

The DocConverter service converts documents to PDF/A:

- Transforms PDF forms, XML forms (created in Designer), and PDF forms created in Acrobat to PDF/A-1b or PDF/A-2b.
- Converts signed or unsigned PDF documents (Digital Signatures required).
- Validates the compliance of a PDF/A file and converts it if necessary.

LiveCycle fully supports the DocConverter service and does not require call to Acrobat to convert or report on compliance.

Redaction

The PDF Utilities service provides options for you to redact PDF documents that have been marked up for redaction using the options in Acrobat. You redact portions of the document that might contain sensitive information to be removed from the document. Unlike Acrobat, the redaction feature in LiveCycle redacts only the visible portions that are marked for redaction.

Document types suitable for conversion to PDF

The following file types are suitable for conversion to PDF:

- PostScript and Encapsulated PostScript (EPS) files
- Native file types
- PDF files

Using PDF Generator, workgroups can convert PostScript files to compact, reliable, and more secure PDF files over a network. Conversion from PostScript to PDF is frequently used to convert large volumes of print documents to electronic documents such as invoices and statements. Document can also be converted to online reports such as business intelligence reports. Converting documents to PDF also lets enterprises send their customers a paper version and an electronic version of a document.

PDF Generator can also convert many native file formats to PDF. Such file formats include all Microsoft Office types, including Word and Excel. Support for the Open Document Format (ODF) through the use of OpenOffice increases the scalability of PDF Generator.

Using PDF Generator, you can convert PDF to other file formats, such as PDF files to print-ready (PostScript) format, without manually invoking Acrobat. You can also convert PDF files to TIFF images.

Input file formats to PDF

PDF Generator converts these native file formats to PDF:

- Adobe Photoshop® CS2, CS3, CS4, CS5, CS5.5 (PSD)
- Adobe FrameMaker® 7.2, 8 (FM)
- Adobe PageMaker® 7.0 (PMD, PM6, P65, PM)
- AutoCAD 2005, 2006, 2007 (DWG), 2009 (English only)

  ***Note:** On 64-bit Windows systems, PDF conversion is possible only if Acrobat X Pro is installed.*
- Corel WordPerfect 12 (WPD)
- Image files (JPEG, GIF, BMP, TIFF, PNG)
- Microsoft Office 2000, XP, 2003, 2007, and 2010 (DOC, XLS, PPT, WPD, MPP, RTF, TXT, Microsoft Office open XML Formats). Requires Acrobat X Professional.
- Microsoft Office Visio 2003, 2007,2010 (VSD). Requires Acrobat X Professional.
- Microsoft Project 2003, 2007, 2010 (MPP)
- OpenOffice 3.3 (ODT, ODS, DOP, ODG, ODF)
- Print files (PS, PRN, EPS)
- Video formats, including SWF and FLV (Windows only)
- WordPerfect X4 (English only)
- Web files (HTML)

For more information, see the LiveCycle Administration Console Help (also accessible in the Administration Console).

PDF to output file formats

Using PDF Generator, you can convert PDF files into other file formats:

- Print-ready format (PostScript) without manually invoking Acrobat
- TIFF images
- HTML 4.01 with CSS 1.0
- PDF/A-1b that uses only the DeviceRGB colorspace
- PDF/A-2b
- PDF/E-1 that uses only the DeviceRGB colorspace
- JPG 2000, TIFF, and PNG
- PostScript and Encapsulated PostScript (EPS)

Open standards support

The following output file formats are supported when the input file is other than PDF:

PDF/X-1a: When a document is converted using the Adobe PDF setting called *PDFX1a 2001*, PDF Generator produces PDF documents. These documents must be checked by or conform to PDF/X-1a:2001. PDF/X-1a:2001 is an ISO standard for graphic content exchange. PDF documents can be opened using Acrobat or Acrobat Reader 4.0 and later. PDFX 1a is specified in ISO 15930-1.

PDF/X-3: When a document is converted using the Adobe PDF setting called *PDFX3 2002*, PDF Generator produces PDF documents. These documents must be checked by or conform to PDF/X-3:2002. PDF/X-3:2002 is an ISO standard for graphic content exchange. PDF documents can be opened using Acrobat or Acrobat Reader 4.0 and later. PDFX3 is specified in ISO 15930-1.

PDF/A-1b:2005: When a document is converted using the Adobe PDF setting called *PDFA-1B*, PDF Generator produces PDF documents. These documents conform to the ISO archival standard for the long-term preservation of electronic documents (called *PDF/A-1b*). PDF/A-1b is specified in ISO 19005-1.

Searchable PDF: This feature is especially useful in converting image file formats such as TIFF. The conversion process uses optical character recognition (OCR) to associate character shapes with the characters they represent. When the resultant searchable PDF file is opened in Acrobat, the document looks the same as the original. However, now you can select the characters (using the text tool) and you can search for them.

Services included with PDF Generator

PDF Generator includes these services that are part of LiveCycle:

- Assembler
- DocConverter
- Generate PDF
- Convert PDF
- Distiller
- Encryption
- PDF Utilities

- XMP Utilities

For more information about the services included in this solution component, see Services Reference for LiveCycle .

Correspondence Management

The Correspondence Management Solution provides the following benefits:

Efficient and Productive

It streamlines processing, enabling you to:

- Deploy and manage templates and content in an intuitive interface designed for business users
- Quickly assemble correspondence from both pre-approved and custom-authored content

Managed

It provides higher quality communications, enabling you to:

- Manage business rules and approved content in a centralized content repository
- Support simple to complex approval and review processes with a robust BPM engine
- Create Adobe PDF files for archiving and auditing
- Create organizationally compliant messages using preapproved content through embedded business rules.

Extensive

It improves customer engagement, enabling you to:

- Add interactive technologies to enable a two-way channel, reducing paper submissions
- Protect sensitive information with built-in, end-to-end document security
- Send correspondence via customer's preferred communication channel, whether electronic, mail, mobile, or fax

Chapter 10: Content Management

Connectors for ECM

The Connectors for enterprise content management (ECM) provide these modules:

- LiveCycle Connector for EMC® Documentum®
- LiveCycle Connector for IBM ® FileNet
- LiveCycle Connector for IBM ® Content Manager
- LiveCycle Connector for Microsoft®SharePoint® (2007 and 2010)

Connectors for ECM provide an efficient development environment. Other LiveCycle modules can produce and consume objects and their metadata to and from the connector services, thereby connecting the ECM system to the engagement application. Developers can leverage contents in ECM systems when developing applications and can leverage services that are part of LiveCycle, within the application.

Key features

Connectors for ECM provide the following features:

- Integrated content repository services to other LiveCycle components
- Improved performance and scalability
- Flexible deployment
- Unified invocation methods through expanded user interface choices
- Expanded support for ECM platforms

The Connector for Microsoft SharePoint provides the following features:

- Allows users to invoke LiveCycle processes, such as an approval process from within SharePoint
- Allows users to convert documents to Adobe PDF and manage the rights on a file in PDF or native formats
- Provides the ability to create and initiate SharePoint workflows that use services in LiveCycle
- Enables users to apply usage rights to PDF files to enable additional features in Adobe Reader
- Allows automation of running LiveCycle processes from within SharePoint workflows
- Enables users to manage assigned tasks and claim new tasks from within SharePoint 2010
- Allows integration of LiveCycle forms with SharePoint Server 2010 and effectively use SharePoint as the repository for form data

Integrated content repository services

Customers can develop LiveCycle engagement applications that process and consume contents that are stored in ECM systems. Connectors for ECM provide integrated content repository services, including content library services, version control, and secure access, to the entire LiveCycle product. Also, through the component services, the connector components provide basic content repository services, such as checkin, checkout, and content metadata access. The Connector for Microsoft SharePoint allows you to integrate LiveCycle forms with SharePoint Server 2010 and use SharePoint as the repository for form data.

Workbench developers can easily map values from other LiveCycle modules, such as Forms, Reader Extensions, or PDF Generator, to the attributes of an ECM object. Connectors for ECM make it possible to reuse and scale the same LiveCycle process to handle forms in different folders of the ECM system. In addition, the connectors include service components to support ECM run-time integration.

Improved performance and scalability

Connectors for ECM preserve existing content schemas defined in an ECM system. By protecting the investment made in the ECM infrastructure, customers have choices for designing schemas to achieve higher performance and scalability of their ECM application.

Flexible deployment

Running the connectors on the same application server and operating system as the ECM system is no longer necessary. Connectors for ECM are deployed with the other LiveCycle components. Flexibility exists because the operating system and application server requirements are independent of those required for the ECM server.

Unified invocation methods

Using the expanded user interface choices, developers can develop applications by using a variety of client interfaces. The following interfaces are supported:

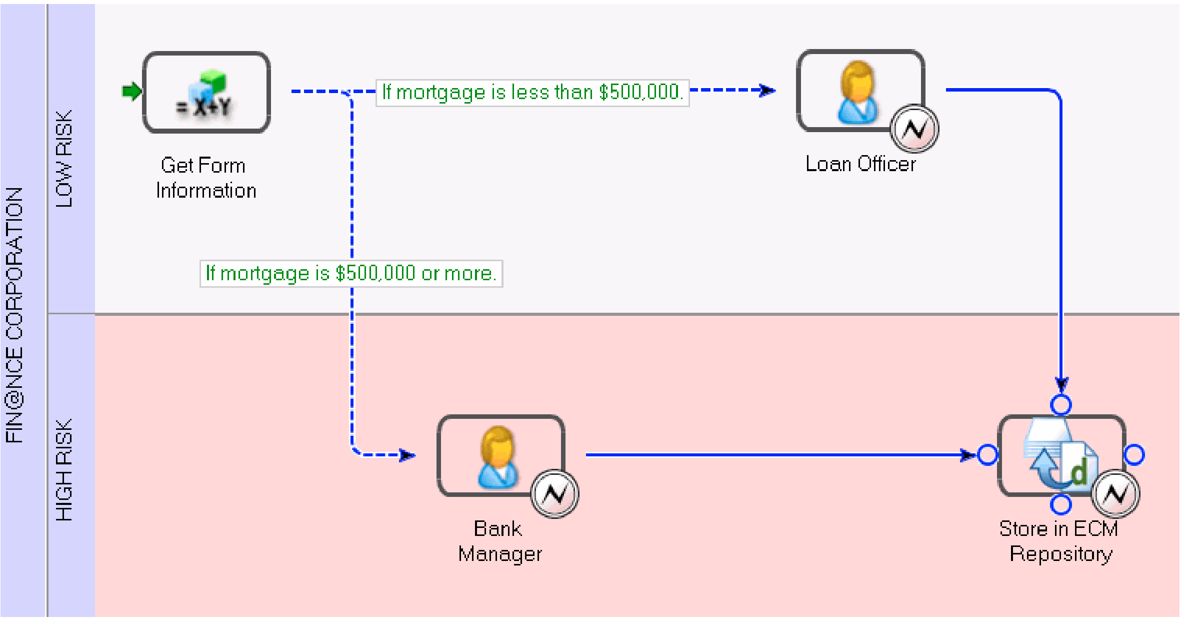
- ECM user interfaces such as Documentum Webtop and FileNet P8 Workplace
- Portals
- Flex or desktop applications
- Workspace

Developers can programmatically invoke the content repository connector services and repository provider services by using web services, LiveCycle Remoting, and Java APIs. They can also invoke the content repository connector services by using watched folders and email invocation methods. These methods are available for calling the services within LiveCycle processes developed in Workbench.

How the Connectors for ECM work

In a typical process that interacts with an ECM system, a LiveCycle application retrieves a document from the ECM repository and stores content in the repository.

The following example shows a mortgage application that retrieves a form from an ECM repository and stores the final document in the ECM repository.



Here is how the mortgage application works in this example:

1. A form author or form developer creates the form design in Workbench, as follows:
   - Goes to the ECM content tree within Workbench
   - Searches the ECM content and then uses the drag-and-drop functionality to add images and fragments to the form design
   - Saves the form design in the ECM system
2. A developer creates the business logic and process for the form. The first step in the process uses a variable that retrieves a form from the ECM repository. Then, the process saves the data in a variable that a user submits from a client application. For example, a user can initiate the process from Workspace and submit the data by clicking a button on the form.

   Based on the amount of the mortgage that the applicant specified, the form data is routed to the appropriate individual for approval. After the form is approved, as part of the last step in the process, the form data is saved in the ECM repository.

Services included with Connectors for ECM

The Connectors for ECM include these services:

- Content Repository Connector for IBM Content Manager
- Content Repository Connector for IBM FileNet
- Content Repository Connector for EMC Documentum
- Process Engine Connector

- Connector for Microsoft SharePoint

For more information about the services included in each ECM solution component, see Services Reference for LiveCycle.

Chapter 11: Services for LiveCycle Modules

This document lists the services that developers can use to create processes in Workbench. Modules are groupings of services that are part of LiveCycle.

When a service is deployed on the LiveCycle server, the service appears below a specific group known as a category in the Services view in Workbench.

Common category

Service	**Licensed for modules**
Assembler	Content Services (deprecated) Forms Output PDF Generator
Barcoded forms	Reader Extensions
Convert PDF	Output PDF Generator
DocConverter	Output PDF Generator
Encryption	Digital Signatures Forms Output PDF Generator Process Management Reader Extensions Rights Management

Service	Licensed for modules
Form Data Integration	Forms Process Management Reader Extensions
PDF Utilities	Content Services (deprecated) Digital Signatures Forms Output PDF Generator Process Management Reader Extensions Rights Management
XMP Utilities	Digital Signatures Forms Output PDF Generator Process Management Reader Extensions Rights Management

Connector for EMC Documentum category

Service	Licensed for modules
Content Repository Connector for EMC Documentum	Connector for EMC Documentum

Connector for IBM Content Manager category

Service	Licensed for modules
Content Repository Connector for IBM Content Manager	Connector for IBM Content Manager

Connector for IBM FileNet category

Service	Licensed for modules
Content Repository Connector for IBM FileNet Process Engine Connector	Connector for IBM FileNet

Connector for SharePoint category

Service	**Licensed for modules**
Connector for Microsoft SharePoint	Connector for Microsoft SharePoint

Content Services category

Service	**Licensed for modules**
Document Management	Content Services (deprecated)

Digital Signatures category

Service	**Licensed for modules**
Signature	Digital Signatures

Forms category

Service	**Licensed for modules**
Forms	Forms

Foundation category

Service	**Licensed for modules**
Decision Point Email Execute Script FTP File Utilities JDBC JMS LDAP Repository Set Value Variable Logger Web Service XSLT Transformation	All modules. Content Services is deprecated.

Output category

Service	**Licensed for modules**
Output	Output

PDF Generator category

Service	**Licensed for modules**
Distiller®	PDF Generator
Generate PDF	PDF Generator

Reader Extensions category

Service	**Licensed for modules**
Reader Extensions	Reader Extensions

Rights Management category

Service	Licensed for modules
Rights Management	Rights Management

The following processes are implemented as services and appear in Workbench based on the module deployed. Modification of the processes associated with the service is not supported.

Output category

Service	Licensed for modules
PrintPDFPackage	Output

Process Management category

Service	Licensed for modules
Complete Task Default Render ES Update 1 Email Notification Form Augmenter Queue Sharing Render Form Guide (deprecated) Render HTML Form Render PDF Form Share Tasks For Shared Queues ShareTask Service Stall Submit Form Guide (deprecated) Submit HTML Form Submit PDF Form User Wait Point Workspace Queue Sharing	Process Management