Interactive PDFs: A Foundation for Form-based Applications

Overview

Adobe® PDFs--Portable Document Format files—have become the de facto standard for creating and distributing high fidelity (faithful to the original) electronic documents. In many organizations the adoption of PDFs has been the first critical step towards streamlining the entire process of creating, distributing, and processing complex forms and documents—a need that is especially severe in regulated industries and government agencies where forms are often the lifeblood of the organization.

Developers have long strived to move past the initial phase of fully electronic document distribution so the entire transaction (from filling in the form to submitting it with authenticated digital signatures) can be done without involving any paper. The technical challenges involved in reaching this goal have been daunting as developers have struggled with how to faithfully present high fidelity interactive forms, manage digital signatures, and integrate improvised solutions into existing applications and business processes. Adobe's enterprise-class document creation solutions solve these challenges.

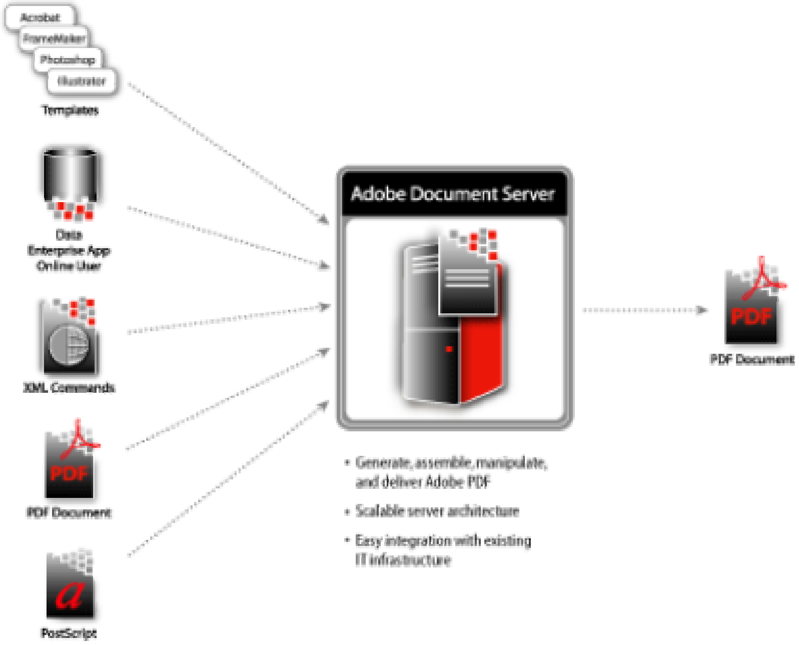
Due to the increasingly mobile nature of many users, additional challenges arise when implementing support for fully electronic document processing. The rise of wireless enabled mobile platforms—such as Intel® Centrino™ mobile technology laptops—means organizations can no longer assume that users will be consistently online throughout the process of completing a digital form. Meeting this challenge requires applications that continue to function in the face of occasional connectivity—a task well suited to interactive PDF forms running in Adobe Reader® on Intel processor-based computers. For example, mobile workers can download an interactive business form, update the information in the form while in the field, and submit the updated document when back online again.

The Changing Role of PDF

Adobe is at the forefront of a revolution in how electronic documents are perceived. No longer are documents and forms merely static entities that can only be read, printed, or filed. With Adobe's enterprise solutions developers have the ability to transform key business documents and forms from passive one-way entities to active two-way application interfaces. The resultant interactive PDFs not only accurately reflect corporate and legal presentation requirements but also allow for the information within the document to be manipulated, extracted, and updated by other automated systems.

"The role of Adobe PDF is changing," says Sydney Sloan, Group Product Marketing Manager, Adobe Systems, "from that of a presentation vehicle to an interactive two-way interface between the user and core enterprise systems. This transformation is changing the way developers use PDFs as they are becoming viewed as suitable replacements for significant portions of application user interfaces." Where in the past developers would generate a crude HTML form, they can now deliver a polished PDF-based interface that can both present information and gather data for input to core systems—with the additional benefit of implicitly supporting occasionally connected mobile users.

Dynamically Creating PDFs



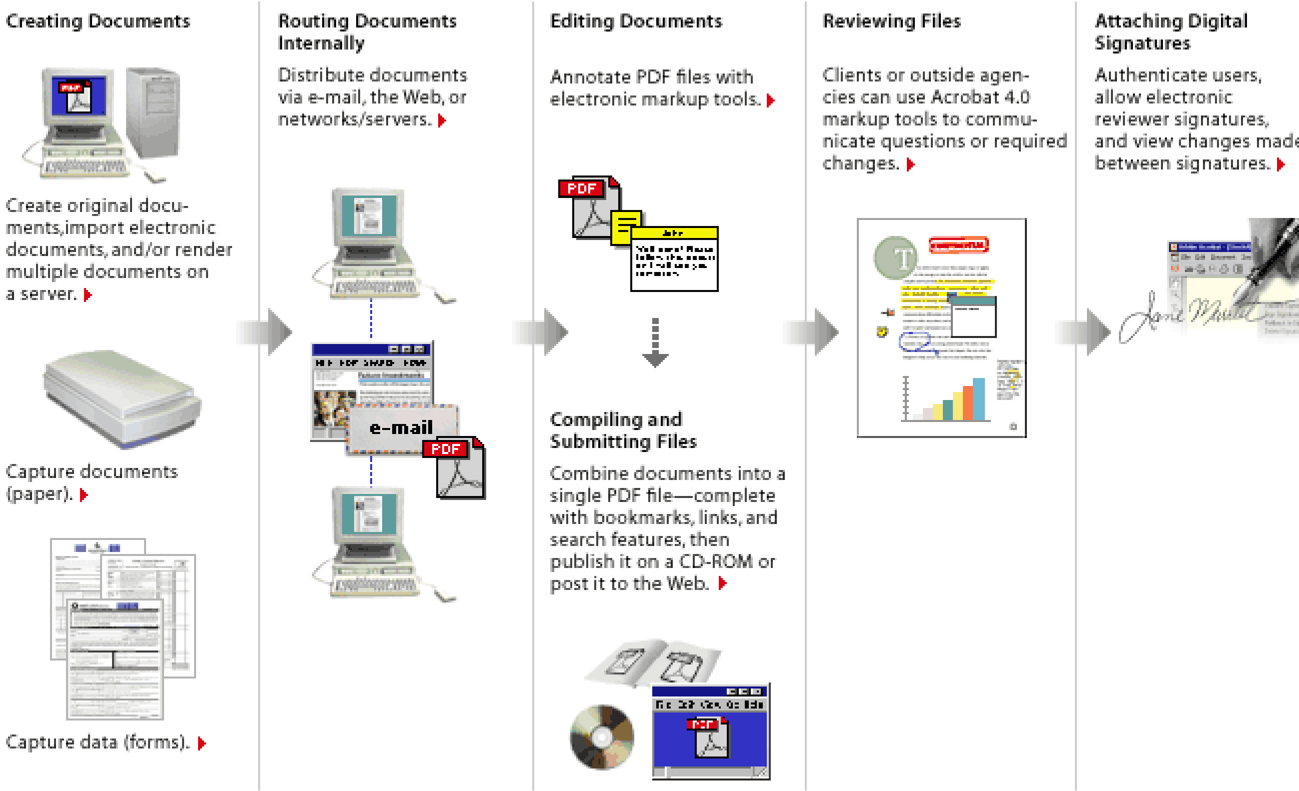
Although the open standards nature of PDF has made it is possible for developers to generate PDF files using a number of third-party libraries, the fastest path to success can be achieved by selecting Adobe Document Server as the cornerstone of a dynamic PDF generation solution. Document Server provides a rich feature set that enables the automated production of interactive PDF documents and forms incorporating content from a variety of sources, including PostScript files, PDF forms, Adobe authoring tools, and XML data (see sidebar for more information on Adobe's use of XML).

Developers faced with the task of bringing together complex documents from multiple PDFs will find Document Server up to the task. It streamlines the document creation process as individual pages from separate PDF files can be added and deleted, or merged while automatically keeping page numbers and embedded links up to date. Changes in document structure are instantly reflected in the table of contents (linked for ease of navigation). Personalized headers and footers can also be placed across long, compound documents. And with support for Scalable Vector Graphics, dynamic charts and graphs can be incorporated into customized documents.

Document Server makes it possible to create personalized PDF documents and forms by flowing XML data retrieved from an enterprise application or database into templates using standard XSL-FO formatting—Extensible Stylesheet Language Formatting Objects is a Web standard that combines XML data with layout information for the dynamic composition of multipage PDF documents containing text, scalable graphics, and images. Because XSL-FO uses the same styling properties as Web pages to create precisely formatted multipage documents, it is the perfect tool for generating PDF documents from Web-based applications.

Document Server is based on a highly scalable, server-based architecture that can run across multiple CPUs, meeting the performance needs of enterprise-scale deployments. Adobe recommends deployments take place on large memory multi-processor servers—systems based on the high performance Intel Xeon™ MP architecture with at least 1GB of memory per CPU would be perfect for this task. Document Server is also designed to take advantage of the benefits offered by Intel's Hyper-Threading Technology as its threaded implementation allows for improved performance due to additional concurrency and better utilization of on-chip execution resources.

Adding Interactivity



Creating Documents

Create original documents,import electronic documents, and/or render multiple documents on a server.

Capture documents (paper).

Capture data (forms).

Routing Documents Internally

Distribute documents via e-mail, the Web, or networks/servers.

Editing Documents

Annotate PDF files with electronic markup tools.

Compiling and Submitting Files

Combine documents into a single PDF file—complete with bookmarks, links, and search features, then publish it on a CD-ROM or post it to the Web.

Reviewing Files

Clients or outside agencies can use Acrobat 4.0 markup tools to communicate questions or required changes.

Attaching Digital Signatures

Authenticate users, allow electronic reviewer signatures, and view changes made between signatures.

The transformation of PDFs from a one-way presentation vehicle into a two-way application interface is completed with the addition of interactive PDF form elements. Traditional Web applications require developers be involved in the process of implementing user interface pages--changing the location of a form field often involves modification of an existing JSP, ASP, or PHP page. Developers that choose to present forms as interactive PDFs will find themselves freed from mundane page maintenance tasks as the responsibility can be taken on by those designing the form.

Form designers create interactive PDF forms using Adobe Acrobat, then enable additional usage rights by using the Document Server for Reader Extensions product. When someone opens a rights-enabled file in Adobe Reader (version 5.1 or later), a dialog box appears stating which additional features are available to use. Users can save interactive forms locally, fill them out on or offline, distribute them to others for review and comment, add digital signatures (using integrated solutions from Adobe partners VeriSign, Entrust, and Valyd), and submit them via e-mail or the Web directly from within Adobe Reader. Rights can be assigned to individual documents using a browser interface, or batch assigned to large volumes of forms via an application programming interface (API).

Interactive documents and forms can be taken to the next level of functionality with the addition of embedded JavaScript code. With this capability a user's selection of form values can be checked for validity on the fly. Developers can also use JavaScript to further extend the capabilities of interactive forms. For example, additional form fields could be created as needed. To avoid subtle differences in browser JavaScript implementations, a full JavaScript interpreter is included in Acrobat and Adobe Reader.

Simple Integration

Once an interactive document or form is completed the information contained within will often be extracted and used to drive further business processes. Adobe has provided developers a variety of ways to make this happen. The simplest option is configuring the document to behave like a HTML form. This method requires the least programming effort but does not support advanced PDF features such as digital signatures and annotations (they cannot be represented in HTML).

The most popular method of retrieving data from an interactive document is by using the Adobe-specific Forms Data Format (FDF) protocol. Developers using this service will have access to the full range of PDF features including digital signatures, uploaded images, and annotations. FDF also allows incremental changes to the contents of PDF documents. By using FDF entire pages can be added to PDF documents on the fly. It is also possible to make wholesale modifications to a page by using FDF to overlay templates on top of an existing page (Acrobat updates the page while preserving the users data).

Applications that rely on XML parsers to process information can make use of the XML Forms Data Format (XFDF) protocol. XFDF is similar to FDF in that it is a compact format that associates form fields with the data contained in them. Unlike FDF the XFDF protocol cannot be used to modify the appearance or functionality of a form. It is also not possible to represent digital signatures or attachments using XFDF.

Application integration with Document Server is made simple by its straightforward design—all functionality is available in an XML-based command language which is exposed as a Web service. Developers can interact directly with the Web service using HTTP/SOAP requests or use one of the provided native APIs. Native language interfaces are provided for Java, Perl, and COM programmers.

Flexible Formatting

The hallmark of PDF has always been its ability to accurately display documents regardless of an output mediums resolution. This flexibility allows the same PDF file to be viewed on a normal display or printed with such resolution as to be indistinguishable from the original. As PDFs have successfully scaled up, they can now also scale down with equal efficiency. The task of viewing a document on devices with different form factors—such as the Intel XScale™ powered Pocket PC—is simplified by Adobe Reader support for handheld devices and Tagged PDF files.

The Tagged PDF format allows a document author to specify the structure of the document using markup tags created in tools such as Adobe PageMaker and Microsoft Word. The Reader client uses these tags to automatically reflow the content for maximum readability on smaller displays. It also supports viewing the document in a full screen mode that shows the PDF file as originally intended—scrolling is required as the smaller display naturally restricts viewing to a portion of the original document. This level of adaptability allows the same content to be appropriately presented on handheld devices and traditional environments without requiring developer involvement.

Summary

The challenges of building a home-grown electronic forms distribution and processing solution warrants developers take a good look at Adobe's enterprise document creation solutions. Where in the past developers viewed PDFs as high fidelity presentation vehicles, they are now becoming aware that interactive two-way PDFs can form the nucleus of a new generation of forms-based applications. The efficiencies offered by these solutions allow developers to rapidly implement fully electronic form and document processing--which benefits the organization by streamlining cumbersome paper-based workflows and providing the personalized and immediate experience user's desire.

<sidebar>

Adobe PDF and XML

Adobe has embraced the Extensible Markup Language (XML) as a key architectural element within its product line. XML is a set of rules for defining semantic tags that break a document into parts and identify the different parts of the document. It is a meta-markup language that that defines a syntax used to define other domain-specific, semantic, structured markup languages.

Where PDF is focused on describing the appearance of information, XML serves a complementary purpose by defining the structure of that information. The combination of these technologies allows for the dynamic composition of polished PDF documents while still preserving the structure of the information embedded in the document.

Adobe uses XML as a key integration technology to link together Adobe and third-party applications. By using XML Adobe can easily pull information together from disparate sources for inclusion into dynamically generated PDFs.

Why XML?

- XML frees data from proprietary codes
- General format for any purpose
- Emphasis on interoperability and easy processing
- Widely known and understood technology
- Basic tools and libraries are freely available

Why PDF?

- PDF frees pages from display problems
- Designed for presentation of high-fidelity forms and documents
- Emphasis on efficient random access, security, and printability
- Can be used online and offline
- Free Adobe Reader and support from multiple authoring tools

Adobe uses XML
</sidebar>

Author

Allan Mc Naughton is the principal analyst at **Triangle Technology Advisors LLC** [link to http://www.triangletechadvisors.com], a firm specializing in the composition of technology white papers. He can be reached at allan@triangletechadvisors.com. [Do not link email]

Image sources:

ePaper document lifecycle
http://www.adobe.com/epaper/features/workflow/images/epaperwkflw.gif

Document Server data flow
http://www.adobe.com/products/server/documentserver/images/ads_workflow_web.gif