Appligent

A Quick Introduction to Acrobat Forms Technology

People have been using Adobe Acrobat to convert paper forms to electronic ones for many years now, but recent developments from Adobe have created confusion about the best way to create and use PDF forms. In this white paper, we will explain the different methodologies Adobe uses with forms—from the backend database to form design interface, and how this can be problematic for those who build PDF solutions.

Multiple Standards for Forms

Adobe currently supports two methods for integrating data and PDF forms. The original method dates back to PDF 1.2 from the mid 1990's and is known as Acrobat Forms or AcroForms. Essentially this technique involves adding the form fields as an overlay on top of the image of a form. Adobe later introduced XFA Forms (sometimes called Designer Forms) with PDF 1.5 and Acrobat 6 in 2003. Both XFA Forms and AcroForms are supported in Acrobat 6 and 7.

Acrobat Forms (AcroForms)

AcroForms are PDF files that contain form fields. Data can be entered into these fields (manually or through an automated process) by the end users or the author of the form. Internally AcroForms are annotations or fields applied to a PDF document. AcroForms are easily filled using formatted ASCII or FDF files that contain key:value pairs that define the field names and associated values that are used to populate a form, as shown in the following code snippet:

```
%FDF-1.2

1 0 obj
<< /FDF
<< /Fields [
<< /T (field-name-1) /V (value of field-name-1) >>
<< /T (field-name-2) /V (value of field-name-2) >>
]>>

endobj
trailer
<< /Root 1 0 R >>
%%EOF
```

The AcroForm itself can be created with Adobe Acrobat 4.x, 5.x, 6.x or 7.x or through many other forms design packages such as Amgraf's OneForm Designer Plus. Users can interact with an AcroForm by using Adobe Acrobat 4.x, 5.x, 6.x or 7.x or by using the free "Reader" application for one of those Adobe Acrobat versions. Adobe also provides a free utility, Fdftoolkit, to help developers build FDF files. In addition, third party developer's tools like Appligent's FDFMerge and PowerMerge allow a programmer to create a system that populates (fills) high volumes of AcroForms as part of an automated process.

In 2003 Adobe introduced an XML-based version of FDF. XML FDF (or XFDF) was to be FDF expressed in the popular XML (Extensible Markup Language) format. Unfortunately Adobe chose to move its XML forms technology in another direction for reasons never

22 East Baltimore Avenue • Lansdowne, PA 19050 • www.appligent.com
P: 610.284.4006 • F: 610.284.4233 • email: sales@appligent.com

fully explained. XFDF was never fully implemented and lacks support for some features of FDF like forms templates and the Fdftoolkit was never updated to support XFDF. Even so, XFDF support is included in Acrobat 6 (PDF 1.5) and above and represents a natural evolution of FDF technology. Like FDF, XFDF is supported by many third-party software vendors.

Adobe XFA Forms (LiveCycle/Designer)

XFA Forms (XML Forms Architecture) represents a significant change in direction for Adobe from the popular FDF and XFDF methodologies. XFA Forms have their roots in a former e-forms company called JetForm (later renamed Accelio) and acquired by Adobe in 2002. XFA concepts were first introduced in PDF 1.5 (Acrobat 6) and expanded in PDF 1.6, (Acrobat 7). Unlike Adobe's earlier forms technology, XFA Forms utilize XML throughout. While this can be viewed as a technically superior approach as it leverages XML as the backbone for all types of structured documents, there are distinct drawbacks to be considered:

- Adobe XFA Forms are not compatible with AcroForms, and they cannot be modified in Acrobat.
- Creating XFA forms requires Adobe LiveCycle Forms Designer which ships with Acrobat 7, and, on the backend there are no commercial or open-source alternatives to the proprietary Adobe Forms Server for processing XFA Forms.
- Existing Acrobat AcroForms cannot be automatically converted to XFA Forms. Typographic fidelity may need to be sacrificed when manually redrawing the forms with LiveCycle Designer.
- Adobe XFA Form documents are not compatible with versions of Acrobat or Reader prior to 6.0. There are no reliable statistics on the versions of Acrobat and Reader currently in use in the world. Anecdotal evidence based on sales and customer support calls indicates plenty of end users are still running version 5 or 4.
- AcroForms JavaScript is not supported with XFA Forms. A different, incompatible JavaScript syntax is used which leads to increased programming costs.
- XFA Forms are not part of the PDF/A (Archiving) standard which is based on PDF Version 1.4. Applications for government or other institutions that must comply with digital archive standards may be problematic.
- There are no commercial or open-source alternatives to using Adobe products for XFA Forms as there are with AcroForms. If you run into problems with XFA Forms you will have no one else to call besides Adobe.

XForms vs. XFA Forms

XFA Forms should not be confused with Xforms, the W3C standard for XML-based forms. Adobe's XFA Forms is a closed standard that competes with the fully open W3C Xforms standard. While both are XML-based, the XForms standard only specifies the data and not the appearance of the form. XFA Forms specify both the form's appearance as well as the data. XForms are not currently supported by Acrobat.

PDF Forms End-User Licensing Models

One of the best things about PDF is the *free* Adobe Reader. Hundreds of millions of copies have been installed around the world to help make PDF the best choice for document sharing. *But nothing else from Adobe is free.* In fact, its server-based components are quite expensive. That's why the market has welcomed competitive third-party plug-ins and alternative server-based tools that support PDF for all kinds of applications, including fillable, database-connected AcroForms.

When it comes to forms, Adobe licenses rights, not solutions. Many independent software developers have made successful businesses by filling this gap. With AcroForms, there are lots of product choices for forms implementation and deployment that are powerful, easy to understand, and economical. Open standards for PDF and XML data exchange are available from many sources, and end-user licensing models can be negotiated to efficiently support small workgroups or large audiences. With third-party suppliers, backend forms servers can be open-source without the need for onerous license fees, or click charges, or per-form charges, or per-user fees.

Conclusion

When investing in technology, enterprises must do contingency planning, and enterprise PDF is no exception. Look carefully at the big picture before embracing XFA Forms. PDF forms (via AcroForms) continue to offer a viable answer for anyone who must deliver an e-Forms solution and needs to maintain one hundred percent fidelity to the original typographic document. System planners should carefully analyze the audience for their forms-based applications and consider if they can control the users desktop configuration. Simply put, can you be sure all of your potential users have Acrobat 7 installed, and if they don't can you force them to upgrade? If not, AcroForms are the best solution. If so, further ask whether relying on a single source provider for this technology makes sense strategically. Is the proposed technology scalable and affordable? If problems arise what will be your plan B? Don't fall victim to marketing hype and be sure to do your due diligence. You'll find AcroForms may be the best solution to your problem.

For more information on AcroForms solutions visit http://www.appligent.com.