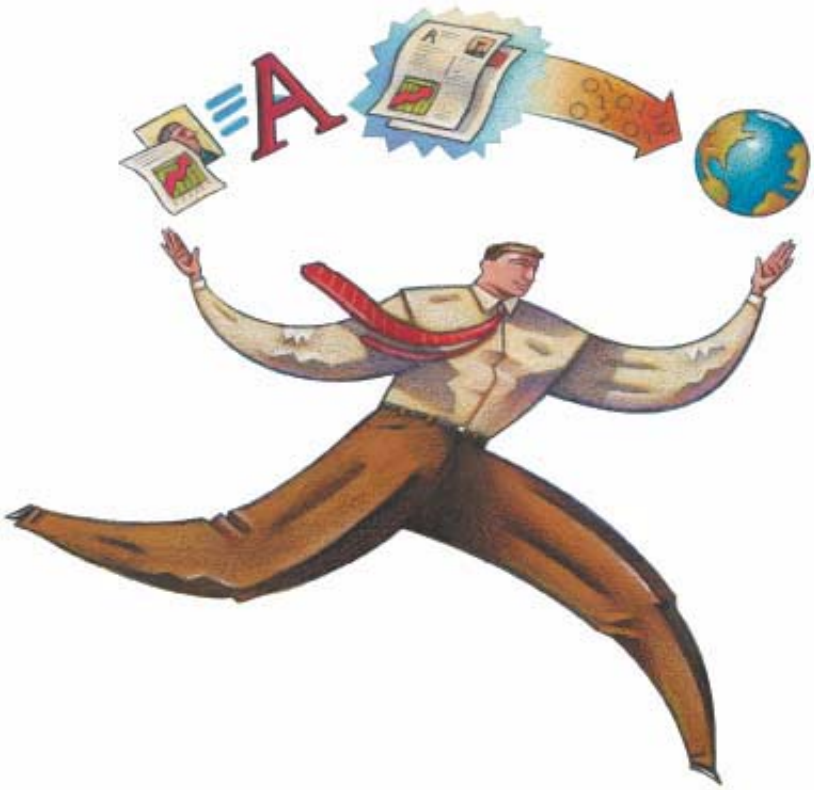
Adobe PDF/XML Architecture - Working Samples

Version 1.0



ADOBE SYSTEMS INCORPORATED
Corporate Headquarters
345 Park Avenue
San Jose, CA 95110-2704
(408) 536-6000
http://partners.adobe.com

August, 2003

© 2003 Adobe Systems Incorporated. All rights reserved.

NOTICE: All information contained herein is the property of Adobe Systems Incorporated. No part of this publication (whether in hardcopy or electronic form) may be reproduced or transmitted, in any form or by any means, electronic, mechanical, photocopying, recording, or otherwise, without the prior written consent of the Adobe Systems Incorporated.

PostScript is a registered trademark of Adobe Systems Incorporated. All instances of the name PostScript in the text are references to the PostScript language as defined by Adobe Systems Incorporated unless otherwise stated. The name PostScript also is used as a product trademark for Adobe Systems' implementation of the PostScript language interpreter.

Except as otherwise stated, any reference to a "PostScript printing device," "PostScript display device," or similar item refers to a printing device, display device or item (respectively) that contains PostScript technology created or licensed by Adobe Systems Incorporated and not to devices or items that purport to be merely compatible with the PostScript language.

Adobe, the Adobe logo, Acrobat, the Acrobat logo, Acrobat Capture, Acrobat Catalog, Acrobat Exchange, Acrobat Reader, Acrobat Search, Distiller, PostScript, and the PostScript logo are trademarks of Adobe Systems Incorporated.

Apple, Macintosh, and Power Macintosh are trademarks of Apple Computer, Inc., registered in the United States and other countries. PowerPC is a registered trademark of IBM Corporation in the United States. ActiveX, Microsoft, Windows, and Windows NT are either registered trademarks or trademarks of Microsoft Corporation in the United States and other countries. UNIX is a registered trademark of The Open Group. All other trademarks are the property of their respective owners.

This publication and the information herein is furnished AS IS, is subject to change without notice, and should not be construed as a commitment by Adobe Systems Incorporated. Adobe Systems Incorporated assumes no responsibility or liability for any errors or inaccuracies, makes no warranty of any kind (express, implied, or statutory) with respect to this publication, and expressly disclaims any and all warranties of merchantability, fitness for particular purposes, and noninfringement of third party rights.

The information in this document is furnished for informational use only, is subject to change without notice, and should not be construed as a commitment by Adobe Systems Incorporated.

Adobe Systems Incorporated assumes no responsibility or liability for any errors or inaccuracies that may appear in this document. The software described in this document is furnished under license and may only be used or copied in accordance with the terms of such license.

Adobe, the Adobe logo, Acrobat, Acrobat Capture, and Distiller are trademarks of Adobe Systems Incorporated. Microsoft and Windows are registered trademarks and ActiveX is a trademark of Microsoft in the U.S. and other countries. Macintosh is a trademark of Apple Computer, Inc. registered in the U.S. and other countries. PowerPC is a trademark of International Business Machines Corporation. UNIX is a registered trademark in the U.S. and other countries, licensed exclusively through X/Open Co. Ltd. All other products or name brands are trademarks of their respective holders.

Contents

Highlights 1

Prepare system 1

Sample Set: eGrants 1

- GrantApplication.pdf. 1
- sample_data.xml 3
- GrantApplication-adsre.pdf 3

Sample Set: TSP 4

- tsp301.pdf 4
- tsp301.xml 5
- tsp301.xdp 5
- tsp301b1.pdf. 6

Contents

A Working with the Sample Files

Highlights

- Demonstrate PDF forms created with Adobe XML/PDF form designer

Prepare system

1. You will need Acrobat 6 or Adobe Reader 6 installed to work with these samples
2. Select PaperClip to the right.
   This is a ZIP that includes all the samples referenced below.

Sample Set: eGrants

Instructions for downloaded sample files (See “Prepare system”)

GrantApplication.pdf - PDF Form

sample_data.xml - XML form data file

GrantApplication-adsre.pdf - PDF Form, rights-enabled for Adobe Reader

NOTE: The following files are also included in the samples for reference, but are not described in detail in this documentation:

GrantApplication.xml - XML form data file conforming to the eGrants XML schema

E-Grants_Global_Schema.xsd - XML Schema for eGrants

E_Grants_SF424_Schema.xsd - XML Schema for eGrants

DUNSValidation.txt - example of JavaScript code used to connect to SOAP web service.

GrantApplication.pdf

This PDF is an illustration of a more sophisticated use of PDF/XML forms. This form was developed for the Federal Enterprise Architecture (FEA) Web Services Pilot Project. As part of the FEA, the US Government has identified 24 eGovernment initiatives. Of these initiatives, the eGrants initiatives was selected for an early pilot program to demonstrate the use of eForms and Web Services for eGovernment.

Many different government agencies are involved in making grants of various kinds. For example, a scientist might apply to the National Science Foundation (NFS) for a grant to

study black holes and a doctor might apply to the National Institute of Health (NIH) for a grant to study vaccines for AIDS. All of these grant programs have similar requirements. Consequently, the goal of the eGrants initiative is to develop systems for managing the grant process that could be shared among all of the agencies that give grants. As a result, the government would achieve cost savings and efficiency by reducing unnecessary duplication of systems. The FEA invited vendors to participate in demonstration program to show how their technology meets the requirements for eForms and web services for eGrants. In a traditional grant application, a principal investigator would fill out a paper grant application which would then be submitted to the appropriate agency which would then route the application to employees and other reviewers for assessment, consideration, and approval or rejection. This process would require that data from the form would have to be entered into internal agency systems so that the grant process can be managed and administered. In order to help automate this process, the government contracted with a consulting firm to create and XML schema that captures the business data involved in the grant process. Sample documents 6 and 7 show the eGrants XML schemas that were developed. The government then invited vendors to show how their technology could leverage the eGrants schemas for eForms and web services to automate the grant business process. Using the new Adobe design tools, Adobe generated the GrantApplication.pdf sample file which combines the pdf files of the paper grant application with the eGrants schema. As a result, a person applying for a grant can fill in the pdf GrantApplication and send it to the agency to which they are applying for a grant. Once the agency receives the completed pdf grant application, they can automatically extract the XML data for integration with their enterprise systems. To see how the xml data can be exported from the sample document, follow the directly below:

1. Open GrantApplication.pdf in Acrobat 6.
2. Fill in some data on the form.
3. Select **Advanced** > **Forms** > **Export Forms Data**. In the Export Forms Data As dialog box, select **XML files (.xml)** in the *Save as type:* pull-down menu.

This PDF also shows how PDF/XML forms can be integrated with web services technologies to further automate forms based processes. GrantApplication.pdf includes a call to a web service using SOAP 1.1 for the validation of DUNS numbers.

To view the JavaScript code for the web service:

Select Advanced > Edit All JavaScripts. This will open the JavaScript Editor window displaying the code for the web service invocation. For your convenience, DUNSValidation.txt is an example of this code.

During the demonstration for the eGrants project, the DUNS validation web service was running on a local machine, but the web service could just as easily be invoked anywhere on the Internet.

The ability to include web services in Adobe PDF/XML forms is another example of the ability to embed sophisticated business logic in pdf documents.

sample_data.xml

This XML file shows how xml data can be used to pre-populate an eform. For example, if an enterprise already has an existing relationship with a customer, their enterprise will already know certain information about the customer and can enter that information directly in the form so that the customer does not have to re-enter data that the enterprise already has. In the grant application example, a researcher may need to renew a previously approved grant. In this case, the agency already has the researchers information and can pre-populate that information into the renewal form.

1. Open GrantApplication.pdf in Acrobat 6.
2. Select **Advanced** > **Forms** > **Import Forms Data**. Select sample_data.xml

GrantApplication-adsre.pdf

Again, all of the capabilities for PDF/XML forms in Acrobat are also available in the free Adobe Reader 6.0 by adding additional usage rights to the form with Adobe Document Server for Reader Extensions. This ability is especially applicable to a government application where the government cannot impose any cost on the citizen for government services which also must be available on whatever technology platform the citizen has. Since the Adobe Reader is free and available on all major platforms, Adobe meets these requirements for government eForms.

Sample Set: TSP

Instructions for downloaded sample files (See “Prepare system”)

tsp301.pdf - PDF Form

tsp301.xml- XML form data file

tsp301.xdp - XML form data package file

tsp301b1.pdf - PDF Form, rights-enabled for Adobe Reader

tsp301.pdf

This sample file should be opened using Acrobat 6. This is a typical form used by employees to allocate retirement savings across several different investment funds. This particular sample illustrates a government agency example, but it is equally applicable to private industry.

In traditional use, an employee would open the PDF file, print it out, fill-in the information, and send in the paper form for processing where the data would then be keyed into an enterprise system. This paper based process may take days if not weeks to complete and potentially introduce data entry errors and delays.

Delays, expense, and effort, can be eliminated using new Adobe technology for PDF and XML which allows the data to be entered directly into the form where it can then be integrated directly with enterprise systems without human intervention.

When the form is opened in Acrobat 6, the user can enter the information directly into the form.

NOTE: when the form is first opened in Acrobat 6, a informative dialog box appears which tells the user that the total of all fund percentages must equal 100. This is an example of simple business logic that can be embedded in a PDF/XML form. The business logic can be as simple or as complex as desired or required by the business process of which the form is a part. For example, the business logic could connect to a data base or a web service. Click OK to dismiss the dialog box.

1. Enter a last name, first name, and optionally a middle name or initial.
2. Enter nine continuous digits (eg. 123456789) to represent a social security number.

NOTE: when you tab out of this field, the form automatically applies the correct formatting to the field (eg. 123-45-6789).
The formatting can be whatever was specified by the designer when the form was created. Formatting can also be localized for language and geography.

3. Enter information (eg. 09-25-1989) to represent a birth date. Again, note that when you tab out of the field, the information is automatically formatted as specified by the designer (September 25 1989).
4. Enter fund percentages that add up to 100. If the fund percentages do not add up to 100 then the dialog box will be displayed again when you tab out of the last one.

tsp301.xml

Now it is time to see the XML form data representation that is produced by the form. To export the XML form data yourself, follow the instructions below:

1. Select **Advanced** > **Forms** > **Export Forms Data**. In the Export Forms Data As dialog box, select **XML files (.xml)** in the *Save as type:* pull-down menu.
   The result of this operation will be an .xml file like the sample file tsp301.xml, but containing the data that you typed into the form.
2. View the xml data file. You can use any simple text editor, or your browser, or a specialized XML editor, like XML Spy.

The schema representation for the form data is merely a simple one that was developed to show the XML output from Adobe PDF/XML forms. The schema to which the data conforms could come from a database, a web service, or from an external industry standard vocabulary such as ACORD (Insurance), eGrants (Government Grants), HL7 (Healthcare), RosettaNet/OASIS EPS (Manufacturing), HR-XML (Human Resources), TaxML (Financial Services), XBRL (Financial Statement Reporting), UBL (eCommerce), or any other one. Adobe's technology for XML support arbitrary customer defined XML data sources. Using the new Adobe design tool, the form designer imports the XML schema and binds the elements of the schema to the form fields. The form can even be created automatically be dragging the XML from the schema onto the Form design canvas.

By combining XML with PDF in this manner, PDF/XML documents become the vehicle for integrating human interaction into enterprise data systems. Furthermore, PDF/XML documents can be used both to collect and distribute data.

Next step - import XML data into a PDF template. However, let's modify the data first.

1. Edit the XML file using an XML Editor or a simple text editor. Try changing the name or social security number. Save the XML data file.
2. Select **Advanced** > **Forms** > **Import Forms Data**. Select your edited XML file.
   Notice the changes are reflected in the PDF form

This illustrates the ability to use PDF/XML documents both for round tripping data or for one way distribution of data from an enterprise system. The PDF/XML document can act as a presentation template which is then merged with the xml instance data to make a complete document. Furthermore, the XML data continues to move with the PDF/XML document and can be exported later if needed. PDF/XML documents can also contain XML data that is not part of the form but is used for other aspects of the business process of which the form is a part. Consequently, the PDF document can act as a data container.

tsp301.xdp

As part of the new Adobe XML architecture, Adobe is also introducing an XML packaging mechanism for PDF called XML Data Package (XDP). This sample (tSP301.xdp) is an example of the XDP file for the tsp301.pdf sample file. To create and XDP from the PDF, follow the instructions below:

1. Open the PDF file (tsp301.pdf).
2. Select **File** > **Save As**. In the Save As dialog box, select '**XML Data Package Files (*.xdp)**' in the *Save as type:* pull-down menu.

The XDP specification is a way to save a PDF files in XML format where a number of PDF sub-assemblies are exposed in the xml file. Some of these sub-assemblies include the XML form template definition and the XML form data schema, among others. The PDF file itself is also base-64 encoded in the XDP. As a result, PDF files can participate in XML workflows where an enterprise has existing infrastructure, applications, and tooling for processing XML. Then when the user needs to interact with the XML data, the XDP file can also be opened in Acrobat or Adobe Reader.

tsp301b1.pdf

This is an example of a PDF/XML document that has also been rights enabled with Adobe Document Server for Reader Extensions, providing users of the free Adobe Reader 6 with the ability to save forms and data locally, fill them out online, distribute to others for review and commenting, add digital signatures, and submit them via e-mail or the Web directly from within Adobe Reader.

Most organizations want to accrue the benefits of using PDF/XML forms, but don't want to impose any cost or technology constraints on their users or customers. Adobe solutions allows organizations to accomplish both of these goals because the Adobe Reader is free and is available on most major technology platforms.

To see how the XML capabilities are enabled in this sample, follow the instructions below:

1. Open the PDF file (tsp301b1.pdf) in Adobe Reader 6.
   Since this sample file has been rights enabled, the user can fill in and save the form using the Reader. Similarly, the xml data can also be exported from the Reader.
2. Select **Document** > **Fill-in Form** > **Export Forms Data**. In the Export Forms Data As dialog box, select **XML files (.xml)** in the *Save as type:* pull-down menu.