

Adobe Acrobat 7.0.5




Acrobat Online Collaboration: Setup and Administration

July 27, 2005



Adobe Solutions Network — <http://partners.adobe.com>



Copyright 2005 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, Distiller, PostScript, the PostScript logo and Reader are either registered trademarks or trademarks of Adobe Systems Incorporated in the United States and/or other countries.

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/or other countries. Verity is a registered trademark of Verity, Incorporated. UNIX is a registered trademark of The Open Group. Verity is a trademark of Verity, Inc. Lextek is a trademark of Lextek International. Linux is a registered trademark of Linus Torvalds. 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.



Contents

Chapter 1	Preface	7
	Introduction	7
	Description	7
	Audience	7
	Organization of This Document	7
	Related Documents	7
	Online Help	8
	Conventions Used in This Book	9
Chapter 2	Acrobat Online Collaboration	11
	Online Collaboration Repositories.	11
	How FDF Files Are Used	13
	How JavaScript Is Used.	13
	Simple Object Access Protocol (SOAP) Support in Acrobat	14
	Extensibility.	14
	Compatibility.	14
	How Comments are Handled for Network Folder and WebDAV Repositories.	15
Chapter 3	Choosing a Repository Type	17
	Assessing Requirements and Resources	17
	The Network Folder Repository	18
	Why Use a Network Folder Repository?	18
	The Web Discussions Repository.	19
	Why Use a Web Discussions Repository?.	19
	The Database Repository	20
	Why Use a Database Repository?	21
	The WebDAV Repository.	22
	Why Use a WebDAV Repository?.	22
	The SOAP Repository	23
	Why Use SOAP?	23
	Repository Type Comparison Table	24
Chapter 4	Designing a Custom Collaboration Interface	27

Online Collaboration Interfaces 27
 Questions to Consider 28

Chapter 5 Configuration and Administration Issues 31

Configuring Your Online Collaboration Preferences 31
 Manual Configuration Setting 32
 Automatic Configuration Setting 32
 Embedding JavaScript in a Document to Set the Server Type Preference 33
Administrative Issues 34
 User Identity 34
 Security 34
 Data Structure 35
PDF Open Parameters 36
Configuring Enterprise Service Discovery for Adobe Acrobat 36
 Choose a Name for the Service. 36
 Determine the Service Type. 37
 Determine the Default DNS Domain 37
 Determine the DNS Service Name. 37
 Add a PTR DNS Record for the Service 38
 Add a SRV DNS Record Entry for the Service 38
 Add a TXT DNS Record for the Service 39
 Test the DNS Service 40
 Using Tracker to Manage Document Reviews 41

Chapter 6 Configuring a Network Folder Repository 43

The Network Folder Repository 43
 Prerequisites 43
 Manually Setting the Network Folder Preference 44
 Automatically Setting the Network Folder Preference 45
Network Folder Platform Issues 46
 Why Network Paths Can be a Problem 46
 Network Mapping for Windows 47
 Network Mapping for Macintosh 48

Chapter 7 Configuring a Web Discussions Repository 49

The Microsoft Web Discussions Repository 49



	Prerequisites	49
	Manually Setting the Discussion Server Preference	49
	Automatically Setting the Web Discussions Preference	51
	Data Storage	52
Chapter 8	Configuring a Database Repository	53
	The Database Repository	53
	Acrobat Database Connectivity (ADBC)	53
	Supported Databases	54
	Prerequisites	54
	Manually Setting the Database Repository Preference	54
	Automatically Setting the Database Preference	56
Chapter 9	Configuring a WebDAV Repository	57
	The WebDAV Repository	57
	Supported Versions	57
	Prerequisites	57
	Manually Setting the WebDAV Preference	58
	Automatically Setting the WebDAV Preference	59
	Data Storage	60
	Bypassing The Proxy Server by Using an Exception	61
	Creating a New Internet Profile on the Macintosh	62
Chapter 10	Configuring a SOAP Repository	63
	SOAP Overview	63
	Acrobat JavaScript SOAP Support	64
	Prerequisites for Configuration	64
	Manually Setting the SOAP Repository Preference	64
	Automatically Setting the SOAP Repository Preference	66
	XML Serialization and Deserialization	66
Chapter 11	Offline Commenting	69
	Working Offline	69
	Offline Data Storage	69
Chapter 12	Troubleshooting	71
	Troubleshooting WebDAV Repositories	71

Network Troubleshooting	75
Client/Server Configuration and Permissions Troubleshooting	76

1

Preface

Introduction

Acrobat® 7.0 Professional and Acrobat 7.0 Standard offer a variety of solutions for collaborative review using a browser and a Web-based comment repository. This document describes the different types of comment repositories which can be used, and explains how to set up and administer the collaboration process.

Description

Because there are several types of comment repositories from which to choose, it is important that you choose the right one. This document will help you decide which comment repository type is best for your needs, and explains how to set up, configure, and administer that repository type.

Audience

This document is intended mainly for system administrators and support personnel who need to configure and support an Acrobat online collaboration system. It also provides useful information for application developers and solution providers who would like to use Acrobat as the basis for a custom collaboration system.

Organization of This Document

This document is organized as follows:

- Chapters 2 and 3 provide a general description of each repository type and discuss how to choose the one most suitable for your needs.
- Chapter 4 outlines the benefits of designing your own custom user interface.
- Chapters 5 through 12 describe how to set up, administer, and troubleshoot each of the repository types.

Related Documents

The following resources provide further information about the Acrobat and PDF Library APIs, and how to use them.

- *PDF Reference*

This document provides a description of the PDF file format, as well as suggestions for producing efficient PDF files. It is intended for application developers who wish to produce PDF files directly, or who wish to gain a detailed understanding of the PDF file format.

Preface

Introduction

- *Acrobat JavaScript Scripting Reference*
This document is the companion reference to the *Acrobat JavaScript Scripting Guide*. It provides detailed descriptions of all the Acrobat JavaScript objects.
- *Acrobat JavaScript Scripting Guide*
This document is the companion reference to the *Acrobat JavaScript Scripting Reference*. It provides an overview of how you can use Acrobat JavaScript to develop and enhance standard workflows.
- *Acrobat and PDF Library API Overview*
This document provides a conceptual overview of the PDF Library and Acrobat core and extended APIs.
- *Acrobat and PDF Library API Reference*
This document provides detailed descriptions of the PDF Library and Acrobat core and extended APIs.

Online Help

- <http://partners.adobe.com/>—This is the Adobe Solutions Network, and offers technical training for Acrobat JavaScript and Online Collaboration. See *Adobe Technical Training: Review and Commenting* and *Adobe Technical Training: Acrobat JavaScript*. It also includes the samples and snippets included in the Acrobat and PDF Library SDKs.
- <http://partners.adobe.com/asn/acrobat/>—A listing of Acrobat resources for developers.
- <http://www.w3.org/TR/SOAP>—W3C: *Simple Object Access Protocol (SOAP) Specification*
- <http://www.w3.org/TR/wsdl>—W3C: *Web Services Description Language (WSDL)*

Conventions Used in This Book

The Acrobat and PDF Library documentation uses text styles according to the following conventions.

Font	Used for	Examples
monospaced	Paths and filenames	C:\templates\mytmpl.fm
	Code examples set off from plain text	These are variable declarations: AVMenu commandMenu,helpMenu;
monospaced bold	Code items within plain text	The GetExtensionID method ...
	Parameter names and literal values in reference documents	The enumeration terminates if proc returns false .
monospaced italic	Pseudocode	ACCB1 void ACCB2 ExeProc(void) { <i>do something</i> }
	Placeholders in code examples	AFSimple_Calculate(<i>cFunction</i> , <i>cFields</i>)
blue	Live links to Web pages	The Adobe Solutions Network URL is: http://partners.adobe.com/asn/
	Live links to sections within this document	See Using the SDK .
	Live links to code items within this document	Test whether an ASAatom exists.
bold	PostScript language and PDF operators, keywords, dictionary key names	The setpagedevice operator
	User interface names	The File menu
italic	Document titles that are not live links	<i>Acrobat Core API Overview</i>
	New terms	<i>User space</i> specifies coordinates for...
	PostScript variables	<i>filename</i> deletefile

Preface

Conventions Used in This Book

2

Acrobat Online Collaboration

This chapter provides a brief description of each repository type and describes the process of uploading, storing, and retrieving comments. It also explains how FDF (Forms Data Format) files work and how Acrobat JavaScript can be used to enhance the review workflow.

Online Collaboration Repositories

Adobe Acrobat provides a platform for the development of a variety of collaborative review systems to fit various needs. A complete, integrated Acrobat and Adobe PDF online collaboration solution includes the following major elements:

- Adobe PDF documents created from source documents authored in various word processing, page layout, and other applications.
- A Web browser environment, running the Acrobat 7.0 plug-in, which allows you to view a PDF document on a network folder or a Web server, add comments to it, and work online with comments from the whole workgroup.
- A central, online comment repository for collecting and storing reviewer comments separate from the associated PDF document being reviewed.
- Optionally, depending on the complexity of the solution required, a user interface to the review and comment workflow, potentially including automated routing and notification, implemented by the solution developer using server-side applications. See [Designing a Custom Collaboration Interface](#) for more information.
- A *Review Management* feature, available in Acrobat 6.0 and higher, which allows you to invite new reviewers, send reminders, and go back online after working offline.

This online collaboration process allows multiple users to comment on a single document and to view comments from others in real time. Also, they can comment on a document at their own convenience and upload their comments at a later time with the offline option. For all repository types, commenting can only occur when the user opens the PDF file within a Web browser. This capability is supported on both Windows and Macintosh machines for Acrobat 7.0.

In considering repository types, be aware that you are not limited to a single repository type for every division of your company. For example, the technical writing department might choose a WebDAV repository to allow outside contractors to participate, while the sales and legal departments might both require a database repository.

The following are the types of browser-based collaboration systems which can be assembled using Acrobat:

- **Network Folder** — A document can be stored on either a network folder or on a Web server, and the resulting comments are stored in a network folder.
- **Web Discussions** — Users can share comments on a document in ways similar to those of a threaded discussion, since users can create topics and reply to others. Comments are stored on a Web server running IIS and Microsoft® Web Discussions (Windows only).
- **Database** — Comments are stored in a database table. The database must be SQL Server or Oracle. This repository type is only available on Windows systems.
- **WebDAV** — Comments are stored on a Web server running WebDAV (World Wide Web Distributed Authoring and Versioning). WebDAV is a set of extensions to the HTTP 1.1 protocol, and allows remote users to cooperatively edit and manage files on Web servers via the Internet.
- **SOAP** — Acrobat 7.0 includes a plug-in that allows for the exchange of messages to and from a remote server using the Simple Object Access Protocol.
- **Custom** — With this repository type, the server can host the configuration code, which can be changed as the online collaboration system evolves. With this approach, an Acrobat JavaScript driver is stored on the collaboration server, and located by the client with a specified URL. This repository type ensures that the file is downloaded only if it has been modified.

A SOAP-based solution is implemented using JavaScript and a server side HTTP service to manage the reading and writing of the annotation data, so in most cases it does require some custom development. Developers can design a custom collaboration system which uses the SOAP protocol to exchange information. See [Configuring a SOAP Repository](#) for more information.

NOTE: Another option is the email-based collaboration feature, but since it does not require a comment repository, it is not discussed in this document.

Each of the standard repository types is discussed further in [Choosing a Repository Type](#).

How FDF Files Are Used

FDF is an Acrobat-specific format used to store form data and other information from a PDF file. An FDF file that is created during a comment upload contains only the reviewer's comments and a reference to the reviewed PDF file. Consequently, the FDF file is typically more compact than the corresponding PDF file, and does not need to be stored under the same parent directory or even on the same machine. For more information about FDF, please refer to the *PDF Reference* in the Acrobat SDK.

FDF files can also contain JavaScript code which can be used to automatically configure the server preferences. For example, a user can navigate to an FDF file on a server and double-click on it. Acrobat will then execute the JavaScript code in the file, which will automatically configure the server preferences for the user.

Chapters 6 through 10 of this document explain how to configure each of the repository types.

NOTE: Another option is to embed JavaScript configuration code in the PDF file to be reviewed, but it is not recommended because it makes it more difficult to repurpose the document for other uses.

How JavaScript Is Used

Acrobat allows you to use Acrobat JavaScript to manipulate PDF files, respond to user actions, control the behavior of a document, and a variety of other actions (see the *Acrobat JavaScript Scripting Reference* and *Acrobat JavaScript Scripting Guide* for more information). When a PDF or FDF file is opened, Acrobat will execute any authorized embedded JavaScript code.

For online collaboration, Acrobat JavaScript provides several important capabilities:

- *Interface to a database* — Acrobat JavaScript provides objects to interact with a database via ADBC (Acrobat Database Connectivity). Using JavaScript, you can read values from a database, update or insert new data, and delete information.
- *Set the comment repository preference* — If you plan on using Acrobat's collaborative review capabilities, you must specify the type of comment repository to be used, as well as its location. Those preferences can be set manually, or via Acrobat JavaScript placed in either the PDF document or its associated FDF file.
- *Interface to a web service* — Acrobat JavaScript provides objects to interact with a web service via SOAP, which is an XML-based transport protocol used for exchanging information. Because it is the developer's choice as to which operations are performed by a given remote procedure, the Acrobat JavaScript SOAP support allows for a more customized collaboration workflow.

For more information, see [Embedding JavaScript in a Document to Set the Server Type Preference](#).

Simple Object Access Protocol (SOAP) Support in Acrobat

SOAP (Simple Object Access Protocol) version 1.1 support is available in Acrobat 6.0 and higher. Acrobat 7.0 provides support for both the SOAP 1.1 and 1.2 protocols by allowing JavaScript clients to perform SOAP request/response transactions via HTTP.

The SOAP support also allows clients to convert WSDL service descriptions into convenience objects that make SOAP transactions more convenient, and permits SOAP transactions to occur asynchronously. Since SOAP is a standardized XML format, there are a wide variety of server-side tools available.

For more information, see [Configuring a SOAP Repository](#).

Extensibility

You can extend the Acrobat Online Collaboration feature in several ways:

- You can write JavaScript code to customize your review process and comment handling.
- Acrobat's Online Collaboration feature can be integrated into a larger document management system.
- You can use Acrobat JavaScript to create new types of comments.
- Through the use of the Acrobat JavaScript **SOAP** object, you can customize the manner in which Acrobat handles comments.

Compatibility

To ensure that all reviewers can use the same set of comment types, it is advised that they use Acrobat 7.0. This would help eliminate the possibility that a reviewer using an earlier version of Acrobat would not be able to see comments created using a later version.

How Comments are Handled for Network Folder and WebDAV Repositories

The network folder and WebDAV comment repositories both handle comments in a similar fashion. The example below illustrates the similarities and differences between both types of repositories.

Example

An author has uploaded a PDF document to a Web server, called **2004Brochure.pdf**, for review. He notifies reviewers to use the URL:

```
http://yosemite.juggler.com/webdav/2004Brochure.pdf
```

The company's IT department has configured everyone's copy of Acrobat to use a WebDAV repository and to use the server location:

```
http://bryce.juggler.com/repository/
```

to store comments. No one has yet reviewed the document.

A reviewer named Michelle Barnes is the first to download the document for review. She marks up the PDF file and clicks **Send and Receive Comments** in her browser. Acrobat uses Michelle Barnes' Windows log-in name, which is *mbarnes*, to construct the name of the FDF file. Acrobat exports Michelle's comments into **mbarnes.fdf**.

Next, Acrobat uploads Michelle's comment file to the right location. Acrobat applies an MD5 hash algorithm to the fully qualified URL of the document:

```
http://yosemite.juggler.com/webdav/2004Brochure.pdf
```

through an MD5-based hash algorithm which results in the value:

```
jY_d0VPgIZG6_jUknGUJbzA
```

This hash algorithm is used to ensure that the resulting folder name is legitimate and manageable since a URL may contain characters that cannot be used in a folder name. This hashed folder name is where Acrobat will upload all FDF files that contain comments for the file **2004Brochure.pdf**. Since this folder does not exist, Acrobat creates it under:

```
http://bryce.juggler.com/repository/
```

and then uploads **mbarnes.fdf**. For more information on how Acrobat handles a URL, see [Data Structure](#).

Later, a second reviewer, Jim Davidson, downloads the document for review. Acrobat retrieves

```
http://yosemite.juggler.com/webdav/2004Brochure.pdf
```

and checks the comment folder for any FDF files. Acrobat determines that there is one FDF file relevant to that document, **mbarnes.fdf**, in the comment folder. Acrobat imports the comments into **2004Brochure.pdf** and displays the result. Jim now sees the PDF file with Michelle's comments.

When Jim finishes his review, he clicks **Send and Receive Comments**. Acrobat uses Jim's log-in name of *JDavidson* to create the name of the FDF file that will hold his comments. Acrobat exports only Jim's comments into **JDavidson.fdf**. Michelle's comments, and those of all other reviewers, are excluded from **JDavidson.fdf**.

Acrobat Online Collaboration

Online Collaboration Repositories

One major difference between the network folder and WebDAV repositories is their treatment of log-in names. The network folder repository may be case-sensitive. Consequently, users with identical log-in names that differ only by case, such as *mbarnes* for Michelle Barnes versus *MBarnes* for Martin Barnes, can comment on the same document in a network folder repository since their resulting comment FDF files will be different (“**mbarnes.fdf**” versus “**MBarnes.fdf**”).

This is not the situation with WebDAV. All reviewers must have unique log-in names that are not differentiated solely by case. For more information, see [User Identity](#).

The network folder repository is the only one in which the author can provide a full path to a document instead of a URL (such as `\\rainier\marketing\brochures`). However, it is important to note that all reviewers must use the same path, and ideally they would all be on the same platform. Otherwise the MD5-based hash value will be different and reviewer comments will be stored in several different folders. When this occurs, reviewers will not be able to automatically see all review comments made on the document by other reviewers (see [Why Network Paths Can be a Problem](#)).

For example, suppose an author sends an email notification to reviewers that the document, **2004ToySpecs.pdf**, is located in `\\Rainier\Engineering\Documents`.

Reviewer Samantha Wanke has mapped network drive R: to `\\Rainier\Engineering` while reviewer Denny Brown has mapped network drive G: to the same path. Samantha and Denny will never see each other’s comments because the different drive mappings result in different hashed values. These problems can be avoided either by having the IT department set up consistent drive mappings on all user machines or by providing a URL to the document location.

3

Choosing a Repository Type

This chapter describes the principal features and advantages of each standard repository type. It is essential that you understand the differences so that you apply your resources toward the best solution for your needs.

In addition, Table 1.1 below provides a comparison of each type according to such criteria as cost, ease of setup, support for external users, and security.

Assessing Requirements and Resources

In order to create a viable comment repository solution, a solution developer must first identify those departments within the company that will be participating in the document review and commenting process. Each of the repository types has requirements pertaining to the operating system and other software needed on the client machines (the author's and reviewers' machines), as well as the servers.

The following list of questions will help you determine the best repository type:

- Are users located in close physical proximity or are they spread over a wide geographical area? Will users outside the corporate firewall be involved in document reviews?
- Do users have PCs, Macintosh[®] machines, or both?
- What operating systems and software are installed on the users' machines?
- What kind of security is needed for the documents?
- Does the company have in-house database expertise?
- Are there any servers available for use as the collaboration server?
- What are the company's requirements for user interaction with the comment repository?
- Does the company have control over the software application versions in use by employees and outside reviewers?
- What are the budget constraints?

Your company may use any combination of repository types. However, for a given PDF document, you should only use one repository type; this will help ensure that the author and reviewers will be able to see each others' comments.

Choosing a Repository Type

The Network Folder Repository

The Network Folder Repository

For the network folder repository type, both the document to be reviewed and the resulting comments can be stored either in a network folder or on a Web server.

Since Acrobat uses a hashing algorithm to name the comment store folder to be used for a document located in a network folder, it is recommended that only one platform (either Macintosh or Windows) be used for all clients. This will help ensure that users will be able to view comments from other users. If you need to use more than one platform, then it is recommended that you use a web server (see [Network Folder Platform Issues](#)).

While a network folder repository is very easy to set up, it is best suited for internal users only. If you would like to provide access to external users, the only solution would be to set up an exposed server to which they can connect.

The network folder repository handles comments similarly to the way in which they are handled by a WebDAV repository (see [How Comments are Handled for Network Folder and WebDAV Repositories](#)).

Why Use a Network Folder Repository?

The advantages of using a network folder repository include:

- A shared network folder is the simplest and least expensive comment repository.
- It does not require any special third-party software or databases; only a file or Web server is needed.

Some disadvantages include:

- If a network folder is used, all clients should use the same platform. If multiple platforms are used (e.g. PCs and Macintoshes), a Web server should be used.
- Remote users can be difficult to support due to security concerns.

All users must have write access to the parent folder that will store all document review comments. User access to documents and folders can be restricted by modifying their protections.

The Web Discussions Repository

The Web Discussion server is a Windows-only feature requiring Internet Explorer 4.0 or higher (this is not a concern for Windows 98 and later, but Windows NT 4.0 users must be aware of this requirement).

With Web Discussions, you can create discussions, add new messages to an existing discussion, and create new discussion threads. Thus, a Web Discussions repository is essentially a threaded discussion, managed by the server, that is applied to your online collaboration workflows. It is used to store web-based messages about a PDF file, but those comments cannot be saved to the PDF file. The threaded discussions are maintained on either the Web server that has Microsoft SharePoint Team Services installed or a remote SQL Server machine.

NOTE: *SharePoint Team Services* is the updated name for the FrontPage Server Extensions, and does *not* refer to the SharePoint Portal Server.

This client configuration is by far the simplest with respect to Acrobat, but it does require configuration of Microsoft's Internet Explorer.

Why Use a Web Discussions Repository?

If your organization is already using Web Discussions, you can leverage this resource by choosing Web Discussions as the medium for the comment repository. A Web Discussions repository allows users to share comments on a document in a similar way to that of a threaded discussion. Reviewers can create topics and reply to other reviewers.

A Web Discussions repository is easy to set up, and is included in SharePoint Team Services (which also contains FrontPage, a tool for creating Web sites).

SharePoint provides you with a ready-made Web site on which to set up a comment repository, and eliminates concerns regarding permissions and directories. However, a Web Discussions repository is slower than a WebDAV repository, is a Windows-based solution, has limited extensibility, poses difficulties for remote users, and is the least secure of all the repository types.

The Database Repository

Web-based applications must implement their own interface to a database using CGI (Common Gateway Interface) scripts, ASP (Active Server Pages) pages, servlets, or other programming applications.

Acrobat provides ADBC (Acrobat Database Connectivity), which provides a simplified means of utilizing ODBC calls to connect to a database and exchange information with it, and represents an appropriate solution under the following circumstances:

- All clients use a Windows operating system
- All clients are internal to the company, although some exceptions may be possible
- All client computers can access a database via a network connection

ADBC is an Acrobat JavaScript-based protocol that allows an Acrobat or XML form to directly access ODBC-compliant databases, such as Oracle, Informix, MS-SQL Server, Access, FoxPro and Excel. A developer first uses ODBC to define a data source. Once the data source has been defined, the developer must use Acrobat JavaScript objects to establish a connection to, and interact with, the desired database or databases. Once this is established, data may be exchanged and manipulated using standard SQL queries.

ADBC does not perform locking or transaction management. If you are using Access or Excel, executing any SQL statement in Acrobat locks the tables involved and prevents them from being modified in design view until the user exits Acrobat.

There are four JavaScript objects used with Acrobat Database Connectivity:

- **ADBC** — provides methods through which you can obtain a list of accessible databases and form a connection with one of them.
- **Connection** — encapsulates a session with the database. A **Connection** object provides methods with which you may create **Statement** objects, obtain information about the list of tables in the data source as well as columns within a given table, and close the connection.
- **Statement**— executes SQL statements. A **Statement** object provides methods with which you may access any row or column within the database table and execute an SQL statement on that data.
- **row** — A **row** object comes into existence when you access the data returned from a **Select** query or via accessor methods available within the **Statement** object. The **row** object's properties are dependent on the data returned from the database, and are the column names retrieved by the **Select** query.

Database repositories are generally considered to be very secure, but their security does depend on the database's security settings as well as system settings. Note that if configuration is established using FDF files, those files would contain server and user information in a form which has limited security. A safer approach would be to configure manually; if the UserID and Password are not specified in the FDF file, the user will be prompted for those items when they first attempt to access the database for writing.

For more information, see [How Comments are Handled for Network Folder and WebDAV Repositories](#).

Why Use a Database Repository?

Database repositories are ideal for collecting information from a PDF form submitted by a user and processing the resultant data. Typically, one would store the form data in a database, where it can later be retrieved, processed, and reviewed.

The database repository type has the potential to be the most secure type of comment repository, but as noted above, the system must be set up correctly to ensure the desired degree of security. Database repositories offer more features that enable you to design customized security for a particular reviewer or for groups of reviewers. For example, both Oracle and SQL Server allow you to extend available functions with custom code. Thus, to effectively set up and administer this type of repository, a company would need database administrators who are knowledgeable with either Oracle or SQL Server.

Although accommodating remote users can present a challenge, it is possible for them to access a database repository as long as there is a reliable mechanism for getting through the company's firewall.

Finally, it should be noted that this repository type can only be used with Windows.

Choosing a Repository Type

The WebDAV Repository

The WebDAV (Web Distributed Authoring and Versioning) commenting feature uses a server which is accessed using the WebDAV extensions to the HTTP 1.1 protocol. Acrobat uses WebDAV commands for adding FDF comment file information to the server collaboration store.

NOTE: With Microsoft Windows 2000, Acrobat uses a Microsoft COM object to access the WebDAV server. This COM object is not case-sensitive and has difficulty distinguishing between such user names as *JPUBLIC* and *jpublic*.

While the client configuration is rather simple, the configuration can be problematic because of IT configuration and security concerns. If the proxy server supports the HTTP 1.1 protocol and the port is specified, remote users will be able to get in through a company firewall. If HTTP 1.1 is not supported, a user can bypass the firewall by using an exception list for the proxy (see [Bypassing The Proxy Server by Using an Exception](#)).

For more information, see [How Comments are Handled for Network Folder and WebDAV Repositories](#).

Why Use a WebDAV Repository?

A significant advantage of WebDAV is that it makes it easier to support remote users. WebDAV requires proper setup of a directory that will serve as the parent directory for all document review comments. For servers using Apache 1.3.14, WebDAV additionally requires the installation of **mod_dav**, which is an open-source Apache module that provides distributed authoring and versioning (DAV) capabilities for an Apache Web server. DAV capabilities are built into Microsoft IIS 5.0 and higher as well as Apache 2.0 and higher, so in these cases no additional software is required.

WebDAV has the fastest response time of the repository types that allow remote user access, except when there is a large number of users. WebDAV, as implemented in Acrobat, requires that the entire FDF file be transferred each time comments are added, updated, or removed. In addition, one connection per user is established when downloading. Thus, file size and the number of users should both be considerations when determining the likely performance of this type of repository.

As with most repository types, a WebDAV repository can be set up in a secure manner. HTTPS can be used, and both authentication and digest authentication is supported for Acrobat 6.0 and higher. Note that some firewalls and proxy servers do not support WebDAV.

A WebDAV solution also offers you more flexibility in tailoring a comment repository interface that closely supports the business practices of the company.

The SOAP Repository

The Acrobat SOAP support can be the basis for creating a customized collaboration workflow. Acrobat JavaScript 6.0 or higher provides objects with which remote procedure calls can be sent from clients to a server, but there is no user interface to help the clients handle such operations as comment uploading.

Why Use SOAP?

There are several advantages to using SOAP as the basis for a custom collaboration system:

- All transactions are based on XML, thus ensuring greater interoperability with XML-compliant applications.
- SOAP repositories can make it easier to support remote users since SOAP's reliance on HTTP makes it easier for SOAP envelopes to pass through firewalls.
- Single comments or other types of updates can be added to the comments store without having to transmit the entire FDF file each time. This can save time and reduce network traffic especially for large documents, thus providing a comparative advantage over the WebDAV approach.
- It is possible to use a wide variety of data store types for saving the annotation data submitted to the server.
- Only one HTTP connection is required to retrieve all of the comments; this may be an important performance consideration for a high latency connection.
- SOAP-based development is well supported because of the widespread availability of standard tools, and offers the advantages of independence from platforms and development environments. In addition, the Acrobat JavaScript SOAP support provides several convenient methods that simplify development tasks by eliminating the need to work directly with XML.
- A SOAP repository can be reasonably secure, depending on how the system is designed and implemented (SOAP can also be used with HTTPS).

In comparison with WebDAV, SOAP repositories are typically less efficient in terms of the space occupied on the server, and the serialization/deserialization process associated with XML parsing can consume more processing cycles than other methods. For more information on the serialization/deserialization process, see [“Configuring a SOAP Repository”](#).

There are significant advantages to developing a custom SOAP repository that addresses your specific requirements. For example, you can design the ideal collaboration user interface along with custom routing and approval processes. For more information, see [Designing a Custom Collaboration Interface](#). It does, however, require more development effort than setting up a standard collaboration solution.

Choosing a Repository Type

Repository Type Comparison Table

Repository Type Comparison Table

Table 1.1 below provides an overview of the features, costs, and benefits of each of the repository types.

TABLE 1.1 Comparison of Repository and Protocol Types

	Network Folder	Web Discussions	WebDAV	Database	SOAP
Cost	Least expensive	Expensive; IIS and Windows Server OS is required	Can be inexpensive if Linux [®] or Apache is used	Excel or Access functionality could be added as part of a custom repository system	Low
Third party software needed?	No	Yes – Microsoft IIS must be running	See footnote ^a	Need a database system	Yes; many open source options are available
Client OS	Can use Mac or PC clients, but not both. For Acrobat 7.0 on Mac, OS X 10.3 ("Panther") or higher is required.	Microsoft Windows	Mac/Win	MS Windows	Mac/Win
Other Client Software	No	MS Office 2000 or MS XP w/SP1	No	ODBC or SQL driver, or Oracle client	No
Supports remote users?	Not well	Depends if the Web site is internal or external to the firewall	Yes, provided that the Web proxy supports it	In some cases	Yes, depending upon internal and external access needs and firewall/proxy issues
Performance	Good	Slower than WebDAV	Fastest if only a few users; for a larger number, having one HTTP-connection per user tends to slow the performance		Good

TABLE 1.1 Comparison of Repository and Protocol Types (Continued)

	Network Folder	Web Discussions	WebDAV	Database	SOAP
Security	See footnote ^b	The least secure	Good	Possibly the best, if the system is set up correctly and if the database itself is secure	Good if set up correctly
Setup	Easy	Easy	Depends on network size and security (firewall/proxy server); difficulty can vary	Labor-intensive because of account management	Requires custom development
Web server needed?	No, but a Web server can be used	Yes ^c	Yes	No	Yes
Web server OS	See footnote ^d	FrontPage Server Extensions from Office 2000 or the SharePoint Team Services from Office XP	Win NT, 2000, or 2003; Win 95 or greater; Most versions of UNIX	Possible but not necessary	Supported on most operating systems
Web server software	See footnote ^e	MS IIS 4 or MS Office	Apache 1.3.14 and mod_dav, Apache 2.0 or higher, or MS IIS 5.0 or higher	NA	Only if you prefer to use a Web server
Database server needed?	No	No	No	Yes	Depends on the application
Database software	NA	MS SQL	NA	Can use MS SQL Server 2000 or Oracle 7, 8i, or 9i	Depends on the application
Comments	Easiest to set up	Windows only	More flexible than Web Discussions	Windows only	Deals with firewalls better than WebDAV

a. For Windows, IIS and Windows Server 2000 or higher is required. For UNIX[®]: just the ModDAV on Apache (which is free and included with Apache 2.0 and higher). Apache Tomcat works on Windows, Macintosh OS X, Linux, and Solaris.

b. For Windows: Basic and Integrated Windows authentication can be set, but not Digest authentication; For Macintosh: both Basic and Digest authentication are supported.

c. The FrontPage Server Extensions and the SharePoint Team Services should both be loaded on IIS.

d. Possible, though unnecessary. You can have a file on the Web server and the comment store on a network folder. If you prefer not to involve a Web server, then use browser-based commenting.

Choosing a Repository Type

Repository Type Comparison Table

e. If a Web server is used to store your PDF files, you can use Windows or UNIX.

4

Designing a Custom Collaboration Interface

Your organization may wish to customize the features in its online collaboration system. The options for customization can range from designing an added collaboration interface to a large-scale integration of the collaboration system with a document management system. While the specifics of how to fully customize and integrate a collaboration solution is beyond the scope of this document, this chapter does suggest some useful customization solutions.

Online Collaboration Interfaces

A simple solution would be to take advantage of the Acrobat online collaboration functionality, which can be used with minimal effort. Solutions based on a network folder repository make it possible to create a collaboration server quickly. The authors are responsible for placing the review document in the proper location and notifying reviewers of that location.

If you prefer to implement a comprehensive collaborative solution, you can build a collaboration interface to facilitate the authors' and reviewers' interaction with the comment repository, thereby streamlining the document review process. The sophistication of the collaboration interface depends on the organization's requirements for its business processes. Since the creation of a collaboration interface involves custom programming and integration, it may be necessary for your organization to contact a solutions provider to assist in the development effort.

A collaboration interface can provide an easily navigable organization of documents, restrictions on document access, and enforcement of the document review workflow. If many documents are to be reviewed, they can be categorized by department or topic area. A collaboration interface can limit access to sensitive design, research, planning, or legal documents, based on an approved list of reviewers. It can also assist authors by notifying reviewers when documents are available for review and automatically sending reminders to tardy reviewers.

Note that document security must be handled before the author submits the document for review.

Questions to Consider

Before choosing the type of review and comment interface, there are several important questions to consider:

1. *Will the collaboration interface be organized around documents or around users?*

One possible approach is to focus the collaboration interface on document presentation. Users see a list of all documents available for review. These can be organized into different categories, or projects can be created to group the documents appropriately. The collaboration interface can restrict access to sensitive documents through password protection. This approach enables users to comment on a variety of documents related to their expertise or interests.

Another possible approach is to concentrate the presentation of information around users. When users interact with the collaboration interface, they see only the documents that they have been assigned. Thus, a user-centric approach helps reduce the time it takes reviewers to find the documents they wish to review.

2. *How much support should the collaboration interface provide for document authors?*

In the collaborative review process, document authors are responsible for transferring their documents to designated locations for review. They would then need to inform reviewers of those locations.

The collaboration interface can be designed to facilitate the author's work. It can automatically transfer the document to the correct location, which can be a network folder or a Web server. If a company uses multiple repositories, which would imply that the reviewer's comment repository preferences must be reset, the collaboration interface can provide an FDF file for reviewers to download that automatically sets the repository preferences. A collaboration interface can also notify reviewers of the document's URL and other pertinent information by sending them email from a list provided by the author.

3. *How much support should the collaboration interface provide for the document review and approval workflow?*

The document review cycle has a finite duration. In most companies, authors spend time and energy trying to gather feedback from reviewers whose responsiveness may vary. A collaboration interface can track those reviewers whose responses are pending, and send reminder notices to them and their managers. It can also terminate the review cycle by making the document unavailable at the end of the review period.

If the company uses a sequential review process, the collaboration interface can drive the workflow. The solution implementor can design the collaboration interface to prompt the author for an ordered list of reviewers and the amount of time allotted to each to complete the review. The interface can send out notifications and reminders, keep track of who actually commented on the document, and notify the author of the review process status.

4. *Will the company want formal acknowledgement from reviewers?*

Many companies require that reviewers initial documents after reviewing them. Acrobat supports the use of digital signatures. However, this feature is treated separately from comments, since only comments are stored in the comment repository when a user clicks **Send and Receive Comments**. Consequently, reviewers must fill in their signature in a separate PDF file, referred to here as the approval PDF document.

An approval PDF document can be handled several ways, depending on the company's document approval workflow.

- Each reviewer could sign an individual approval PDF document and email it to the repository administrator or the author.
- The approval PDF document could be designed with a **Submit** button that uploads the signed PDF file to the Web server. Acrobat has built-in functionality that makes it possible to submit the entire PDF file to a Web server. The Web server would then save the approval PDF document to an appropriate location.
- The reviewers could download a blank approval PDF document from the Web server and insert the approval PDF page or pages into their version of the reviewed document. Once the signed approval pages have been added to the reviewed documents, they can then be sent to the collaboration interface or repository administrator for archival.
- A single approval PDF document can be made available for all reviewers to sign. A checkout system can ensure that only one reviewer at a time actually signs the approval document.

Designing a Custom Collaboration Interface

Online Collaboration Interfaces

5

Configuration and Administration Issues

This chapter describes the various options for setting server preferences, and addresses such issues as user identity, security, and data structure. All of these factors are critical to the proper configuration of both the client and server, and to the ongoing administration of your online collaboration system. The specific procedures for configuring the preferences are described in the chapters dedicated to each specific repository type. In addition, this chapter also provides descriptions of two important features: PDF Open parameters specified in URLs and automated enterprise service discovery.

Configuring Your Online Collaboration Preferences

For most repository types, you can set the configuration preferences in one of four ways:

- Manually — users can set the server preferences in Acrobat using the instructions provided in the subsequent chapters.
- Automatically — an FDF file containing JavaScript can be used to set the server preferences. See [Automatic Configuration Setting](#) for more information.
- Embedding JavaScript in the PDF file — JavaScript code is placed in the PDF file to be reviewed. When a reviewer opens the file, the code is executed and the preferences are set. Note that this approach makes it more difficult to repurpose the PDF file, and is thus discouraged.
- Acrobat Enterprise Installer — the system administrator can set the installer to configure every user's system for the desired server preferences upon installation of Acrobat.

The first three of these methods are discussed below, and specific instructions for manually and automatically setting preferences are given in the chapters dedicated to each repository type. Instructions for setting up the Enterprise Installer to set server preferences are included with the software.

Manual Configuration Setting

If your company or department only needs one type of repository, it is best to configure it manually, since end users will not need to continually change the settings. The system administrator can assign comment repository settings when configuring a workstation for a new employee. This will eliminate the possibility that users may accidentally configure the repository preferences incorrectly.

Comment repository preferences in Acrobat have global scope. In other words, the preferences apply to every PDF file that a user views. If the comment repository is located on a server that requires a user name and password, the login prompt appears for every PDF file a user opens in the Web browser, even for PDF files located on Web sites outside the company. To address this issue, you may set up the repository preferences on a per-document basis using either FDF files or document-level JavaScripts as described in the following sections.

Each user must set the comments server preferences before their collaboration session can begin. This can be done by selecting **Edit > Preferences > Reviewing**. The server preferences will remain in effect until changed.

Automatic Configuration Setting

You can create a single FDF file to automatically configure the preferences for all users who open the file in their browsers.

FDF refers to Acrobat's *Forms Data Format*, which Acrobat uses to receive information from a server and fill in form fields. An FDF file specifies the PDF document to which it applies, and can also specify the type of comment repository to be used as well as the location of the comments. The FDF file can also communicate with the Acrobat JavaScript engine, which is used to automatically set the users' server preferences.

For example, a user can be directed to an FDF file that has been coded with the preference information, and open it with their Web browser. When Acrobat processes this FDF file, it automatically retrieves the associated PDF file, sets the repository preferences, and imports any reviewer comments.

Most authors will not have the desire or expertise to create an FDF file that specifies comment repository preferences for an individual PDF document. Consequently, this particular approach works best if the repository administrator assumes responsibility for creating the appropriate FDF file when an author adds a document to the repository. Fortunately, this process can be automated using an intelligent front-end, developed by solution implementors, with which authors interact. Sample FDF files are provided in the chapters dedicated to each repository type.

Why Use an FDF File to Set Up Comment Repository Preferences?

Assigning the comment repository in an FDF file precludes the possibility of any user error in setting up the comment repository. It is a flexible approach that allows you to specify that the comment repository applies to all PDF files or to a particular PDF file. For examples of how to set up a global repository preference, see the Acrobat Technical Training: Acrobat Collaboration Training at:

<http://partners.adobe.com/asn/>

Alternatively, a network administrator can create an FDF file for each repository type, ensuring that the server preferences are automatically set for the user. This is useful for people who frequently comment on multiple documents.

For ease of use, the administrator can place the FDF file at a readily accessible location, such as a frequently used Web site. Once this FDF file is opened in the browser, the user is taken to a confirmation page which says that they have successfully set their collaboration server settings.

Embedding JavaScript in a Document to Set the Server Type Preference

As an alternative to creating an FDF file, and to avoid having to reset the server type and location for each collaboration session, the server preferences can be set in a document-level JavaScript which runs automatically whenever the document is opened. To accomplish this, select **Advanced > JavaScript > Document JavaScripts**, create a name for your script, click **Add**, and delete the function script that Adobe has inserted.

Insert code similar to the following example for a WebDAV repository:

```
Collab.defaultStore = "DAVFDF";  
Collab.setStoreSettings("DAVFDF", "http://example.org/webdav/");
```

The first line associates the folder with the WebDAV server type, and the second line makes WebDAV the default server type. Note the use of a device-independent path to the folder holding the comments. For more information on device-independent paths, see the *PDF Reference*. For more complete descriptions of the Acrobat JavaScript support for Online Collaboration, see the *Acrobat JavaScript Scripting Reference* and *Acrobat JavaScript Scripting Guide*.

Configuration issues for each server type are described in the following chapters. There are, however, several elements of online collaboration which are consistent across server types, and are discussed below.

NOTE: While embedded JavaScript code can be used to do server configuration, it is not generally recommended because it makes it more difficult to repurpose a PDF document.

Administrative Issues

For an online collaboration system to work properly, the issues of user identity, security, and data structure must be addressed. They are discussed in the following sections.

User Identity

Since several users may comment on a single document, it is essential that each of the comments is attributed to the correct person. To ensure that the comments are attributed correctly, Acrobat automatically inserts the name that appears as the **Login Name**, shown on the **Identity** panel within the **Preferences** dialog box, into the title bar of the annotation pop-up note.

On Microsoft Windows systems, Acrobat uses the Windows login name. On Apple Macintosh systems, Acrobat uses either the login name or file sharing name, depending on which exists. User names are generally case sensitive, which means that *JPUBLIC* is a different user than *jpublic*. See [Configuring a WebDAV Repository](#) for the only exception to this rule.

NOTE: If the user has bypassed the Windows login screen, the default user name is “GUEST.” This may cause confusion when the user uploads their comments. See the *Security* section below for more information.

In the network folder and WebDAV repository types, when the login name contains something other than letters or numbers, Acrobat will substitute the hexadecimal representation of that character within the login name. For example, comments for a user with the login name of `John Public` would be stored on the server in a file called `John_0020Public.fdf` because the space in the name is translated as a hexadecimal ‘0x0020’.

Acrobat does not use a checkout or locking mechanism for comment files. Therefore, to prevent confusion and possible loss of information, enterprises should not allow multiple users to share the same user name.

NOTE: On Macintosh operating system versions 9.0 and 9.1, there is a limit of 31 characters for a file name. This restriction does not apply to Macintosh OS X. However, it could apply to a Windows server hosting a file using Apple File Services. Hence it is important to consider the possibility that users with many spaces or non-alphanumeric characters in their user name may exceed the maximum allowable characters for a file name on certain systems.

Security

Acrobat controls access by allowing comments to be uploaded only with the login name, and only if the login name has permission to create files on the server. In the case of a default login name, as mentioned in the *Identity* section, the comments would be uploaded as `GUEST.fdf`. If there were more than one person logged in as “GUEST” the comments of each user would be saved to the same file.

Acrobat does not verify that the user has write privileges to either the root or hash directories. If the user does not have write privileges, or the directory does not exist, then Acrobat will display an error message stating that the comments could not be saved and uploaded.

Data Structure

When the user opens a PDF document in the browser, Acrobat reads the URL from the network layer, and then uses Domain Name Servers (DNS) to translate the name to an IP address. The DNS conversion guarantees that the comments will be stored in the correct repository regardless of whether the users specify the same URL to access the collaboration document. Acrobat then converts the IP address into an MD5 hash in order to create the name for the comments folder. For example:

```
http://www.adobe.com/docs/document.pdf
```

would be converted to

```
http://152.30.132.12/docs/document.pdf
```

which would subsequently hash into 'UAtOJErFgHsL2xbLvPPJgC'. More information on MD5 hash can be found in RFC 1321 or in many places on the Web, such as:

<http://www.ietf.org/rfc/rfc1321.txt>

NOTE: It is not possible to reverse the MD5 hash to obtain the URL.

Do not change the location of the collaboration document until the collaboration period has expired, because the folder names are based on the URL of the original document. If the location of the collaboration document is changed and more comments are added, the new comments will be saved to a different location from that used for the initial comments.

When working with multiple versions of a document, name each version differently; otherwise the comments will not align properly. In the example above, the MD5 hash for

```
http://152.30.132.12/docs/document.pdf
```

after the URL is put into a canonical form, is:

```
UAtOJErFgHsL2xbLvPPJgC
```

while the MD5 hash for:

```
http://152.30.132.12/docs/document2.pdf
```

is:

```
1N2AwJKLt6BxpzjHp4_W7B
```

If you were to revise `document.pdf` and place the updated document in the same location without changing the file name to reflect the version, the comments saved under that hash would not correlate to the new document. This could result in such problems as highlighting that exists where there are no longer any words, or line drawings for a diagram that has moved to a different page.

PDF Open Parameters

Adobe defines parameters that allow you to open a PDF document with a command or URL that specifies exactly what to display (a named destination or specific page), how to display it (using such characteristics as a specific view, scrollbars, bookmarks, comments, or highlighting), set the collaboration store, manipulating the Acrobat Inbox, and open or remove items in the Review Tracker.

For more information, see *PDF Open Parameters*.

Configuring Enterprise Service Discovery for Adobe Acrobat

Adobe Acrobat 7.0 provides the ability to locate services automatically using DNS Service Discovery. Services that are advertised using Multicast DNS can be located without manual configuration if the service and client are located within the same network subnet. To discover services throughout a corporate network, however, services are located using Unicast DNS queries and the existing DNS infrastructure.

The following procedures outline how to advertise a service throughout an enterprise by configuring DNS. It is assumed that you have experience working with DNS.

Choose a Name for the Service

Choosing an appropriate name is crucial for advertising the service. The name will be displayed to the user as part of the Acrobat user interface, and should convey sufficient meaning for the users to decide whether it is appropriate for their needs. The name must be unique and should indicate the purpose of the service (e.g. "Accounting Collaboration Service"), and possibly its location (e.g. "Engineering Collaboration Service"). The name may contain spaces and other special characters. When the Service name is entered into DNS, the special characters must be escaped by replacing them with their decimal UTF-8 character code. For example, a space character would be expressed as \032. Thus, the examples given above would be entered in DNS as follows:

Accounting Collaboration Service: Accounting\032Collaboration\032Service

Engineering Collaboration Service: Engineering\032Collaboration\032Service

Determine the Service Type

Acrobat searches for different types of services in different parts of the user interface. The service type should correspond to the service that you intend to publish. The service types and descriptions are described below in [Table 1.2](#):

TABLE 1.2 *Service Types*

Service Type	Description
<code>_acrobatSRV._tcp</code>	An Acrobat Service
<code>_acrobatRMACR._tcp</code>	An Acrobat Review, Markup, and Approval Repository

Determine the Default DNS Domain

Adobe Acrobat searches for services in the default DNS domain. On Windows, this can be determined by executing the **ipconfig /all** shell command. The default DNS domain corresponds to the "Connection-specific DNS Suffix" value.

Determine the DNS Service Name

The *DNS Service Name* is the name of the service as it will appear in the DNS entries, and can be created from the *Service Name*, *Service Type*, and *Default DNS domain*. The DNS Service Name is of the following form:

<Service Name>.<Service Type>.<Default DNS Domain>.

If the Service Name contains special (non-alphanumeric) characters, those characters must be escaped. For example, a space must be replaced with \032. The period (.) appearing after the Default DNS Domain, as shown above, must be included. For example, if the Service Name is "Example Service Name" for an Acrobat Service, and the Default DNS Domain is "corp.example.com", the DNS Service Name would appear as follows:

`Example\032Service\032Name._acrobatSRV._tcp.corp.example.com.`

Configuration and Administration Issues

Configuring Enterprise Service Discovery for Adobe Acrobat

Add a PTR DNS Record for the Service

The first DNS entry needed to advertise the service is a PTR record that maps the Service Type to a particular Service Instance. In BIND the entry would appear as follows:

<Service Type>.<Default DNS Domain>. IN PTR <Service Name>

The period (.) appearing after the Default DNS Domain, as shown above, must be included. Using the same example, the entry would appear as follows:

```
_acrobatSRV._tcp.corp.example.com. IN PTR Example\032Service\032Name
```

Add a SRV DNS Record Entry for the Service

The SRV record allows the Service Name to be bound to a specific IP address and port corresponding to the service. In BIND the entry would appear as follows:

<Service Name> IN SRV 0 0 <port #> <Service Target>

If the service above were running on port 8080 of the machine **exampleService**, the entry would appear as follows:

```
Example\032Service\032Name IN SRV 0 0 8080  
exampleService.corp.example.com
```

Add a TXT DNS Record for the Service

The TXT record allows the Service Name to be bound to a set of named parameters that are specific to the service you are publishing, and are described below in [Table 1.3](#):

TABLE 1.3 *TXT Parameters*

Service Type	TXT Parameters
<code>_acrobatSRV._tcp</code>	<p>txtvers: The version number of the service. This is 1 for Acrobat 7.0.</p> <p>path: The path portion of the URL for the service. For example, if the URL is http://example.org/services/ExampleService then the path parameter would be set to services/ExampleService.</p> <p>feedType: The Service Type (either "RSS" or "JS"). "RSS" indicates that it is an RSS Feed to be deployed in the Acrobat Tracker, and "JS" indicates that it is an Initiation Wizard.</p>
<code>_acrobatRMACR._tcp</code>	<p>txtvers: The version number of the service. This is 1 for Acrobat 7.0.</p> <p>setting: The collaboration store setting. This is dependent on the type of repository being used.</p> <p>type: The collaboration service type. The possible values are "DAVFDF" (WebDAV), "FSFDF" (Network Folder), "ADBC" (Database Connection), "CONFIG" (Custom Server Type).</p>

In BIND the entry would appear as follows:

<Service Name> IN TXT "<p1 name>=<p1 value>" "<p2 name>=<p2 value>" ... "<pN name> = <pN value>"

For example, if the service above is an Acrobat Service with a feedType of RSS and a path of RSS/example.rss, then the entry would appear as follows:

```
Example\032Service\032Name IN TXT "txtvers=1" "feedType=RSS"
"path=RSS/example.rss"
```

Configuration and Administration Issues

Configuring Enterprise Service Discovery for Adobe Acrobat

Test the DNS Service

The service should now be advertised using DNS. To test that it has been advertised properly, you can use **nslookup**.

To find the services for a specific service type, use the following syntax:

nslookup -q=ptr <Service Type>.<Default DNS Domain>

For example, to search for an Acrobat service in the domain corp.example.com, use the following syntax:

```
nslookup -q=ptr _acrobatSRV._tcp.corp.example.com
```

This will return the set of service names that are advertised as Acrobat services. To obtain the SRV and TXT records for a specific service name, use the following syntax:

nslookup -q=any <Service Name>.<Default DNS Domain>

For example, to obtain the SRV and TXT records for a service named "My Example Service", use the following syntax:

```
nslookup -q=any My\032\Example\032Service.corp.example.com
```


Using Tracker to Manage Document Reviews

Tracker, which is available in both Acrobat Professional and Acrobat Standard versions 6.0 and higher, can be used to monitor PDF document reviews by publishing information using RSS (Really Simple Syndication) 2.0, which is an XML-based format used for describing lists of items. It can also be used to produce a visual client for a Web service application.

The Tracker's URL API can be used to manage subscriptions through the usage of application integration. This is accomplished with a specialized **acrobat:** notation, which uses XML extensions to customize the user interface. This syntax may be used to create links which add, update, or remove subscriptions to document reviews, open a PDF document in the browser, or display the Tracker window, as shown in the following examples:

The following URL shows the Tracker:

```
acrobat:Inbox?show
```

The following URL adds a feed to the Tracker:

```
acrobat:Inbox?addFeed=http://www.example.org/rss/example_rss.xml
```

The following URL refreshes the same feed in the Tracker:

```
acrobat:Inbox?refresh=http://www.example.org/rss/example_rss.xml
```

The following URL selects the same feed in the Tracker:

```
acrobat:Inbox?select=http://www.example.org/rss/example_rss.xml
```

The following URL converts the same feed to PDF. If it is not currently subscribed, it will be converted without adding it to the Tracker:

```
acrobat:Inbox?convert=http://www.example.org/rss/example_rss.xml
```

Configuration and Administration Issues

Configuring Enterprise Service Discovery for Adobe Acrobat

6

Configuring a Network Folder Repository

The Network Folder Repository

The network folder repository is a simple mechanism for collaborative reviewing using a folder on a network. This is the easiest of the repository types to use because it does not require a Web server, database, or any CGI code. However, you may use a Web server if you specify the store location in the **Edit > Preferences > Reviewing** dialog box.

With a network folder repository, the collaboration document is placed somewhere on the network, and the reviewers are notified. As each reviewer opens and comments on the document, Acrobat creates an MD5 hash of the document URL, and the comments are saved in the hashed sub-folder as an individual FDF file. Note that the PDF file and its accompanying FDF files do not need to be stored under the same parent directory, or even on the same machine.

Because of the way network paths are specified on each platform, network folder reviews can be difficult to implement in cross-platform environments. For example, Windows and Macintosh users may not be able to see each other's comments. This problem is most evident when specifying the location of the document to be reviewed. The location of the comment repository is a less complex issue; in this type of situation, a Web server would help resolve the problem of establishing the document location. For more information, see [Network Folder Platform Issues](#).

Prerequisites

The following is the network folder repository requirement for the server:

- A network folder must be chosen to store the comments. This folder must be accessible and allow write access to all the individuals commenting on the document.

and the following are network folder repository requirements for the client:

- All users must have Apple Macintosh OS 9.0 or later, Microsoft Windows 98, Windows NT 4.0, Windows Millennium Edition, Windows 2000 or higher, or Windows XP.

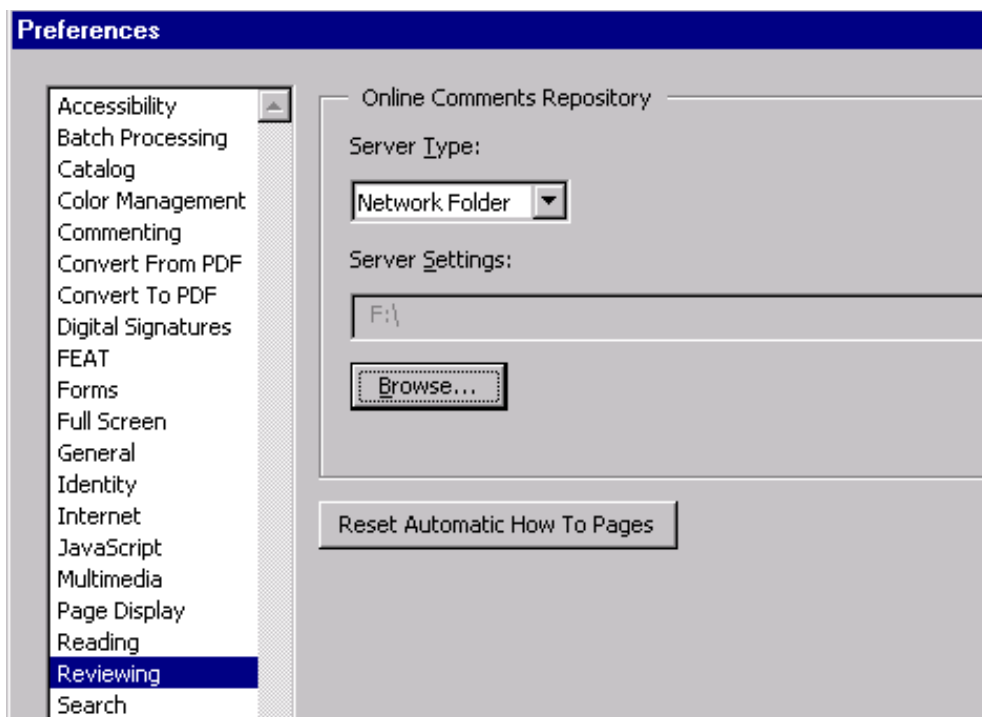
NOTE: For the Macintosh operating systems, Adobe Acrobat 5.0 is required. For browser-based collaboration on the Macintosh, Acrobat 7.0 is required. In either case, Acrobat 6.0 cannot be used for a Macintosh.

- The document must be placed in a location accessible to all reviewers.
- Optional: A local drive may also be used to store the comments, although in this case, comments can only be created and viewed by one person unless access to that disk is shared with the network.

Manually Setting the Network Folder Preference

The following scenario can serve as an example. Suppose that John Public and Jane Doe will be commenting on a document called `http://www.adobe.com/docs/document.pdf`, and that the comments will be held in a folder called `F:\Adobe\collab\annots\`. To select the server:

1. Start Acrobat, and select **Edit > Preferences > Reviewing**.
2. Change the Server Type to *Network Folder*, then click **Browse...** to identify the network folder where the comments are to be stored.



Example

When John Public and Jane Doe create comments and upload them to the server, Acrobat creates a hash folder and saves their comments as:

```
F:\Adobe\collab\annots\48dsn30fhbfw\jpublic.fdf
```

and:

```
F:\Adobe\collab\annots\48dsn30fhbfw\jdoe.fdf
```

NOTE: Though this example uses the letter name of the drive, Acrobat also supports UNC paths. The equivalent UNC path would appear as follows:

```
\\diskfarm\collab\annots\48dsn30fhbfw
```

Automatically Setting the Network Folder Preference

The Network Folder preference can be set automatically by using an FDF file containing JavaScript code.

The following script sets the repository to `\\fileserver\shared\collab`, and the store type for a network folder repository is **FSFDF**. Note that the `Collab.setStore` method's second argument is a device-independent path expressed with UNC path representation. Refer to the *PDF Reference* for more information on device-independent paths.

A sample JavaScript is given below:

```
%FDF-1.2
%âãÿÓ
1 0 obj
<<
  /FDF
  <<
    /F (http://reviewserver/pdfs/settingsupdated.pdf)
    /JavaScript
  <<
    /After 2 0 R
  >>
  >>
endobj
2 0 obj
<<
>>
stream
  Collab.defaultStore = "FSFDF";
  Collab.setStoreSettings("FSFDF", "/fileserver/shared/Collab/");
endstream
endobj
trailer
<<
  /Root 1 0 R
>>
%%EOF
```

In this sample script, `http://reviewserver/pdfs/settingsupdated.pdf` is the PDF document that opens in the browser after the collaboration settings have been set in Acrobat. To adapt this script to your needs, modify the PDF file name and the store settings.

Network Folder Platform Issues

Specifying a network path to the location where a document is placed for review can be problematic because of the way networks function. Windows and Macintosh systems map network drives in different ways, and there are even variations within each environment. This section explains how the problem may occur, and how to solve this type of problem through the usage of a Web server.

Why Network Paths Can be a Problem

On Windows, a file name could appear as shown below:

```
\\mainserver\users\jsmith\example.pdf
```

or:

```
D:\jsmith\example.pdf
```

However, on a Macintosh system, its appearance might be specified in different ways depending on the operating system used. The syntax in the example below is used in Macintosh OS 9.0, but would appear differently with OS X Apple File Protocol and OS X SMB:

```
Server Zone:mainserver:Users:jsmith:example.pdf
```

Since the syntax used in each case is different, Acrobat will create different hashed-name folders. A successful online collaboration workflow depends on a single name that is accessible from any platform. In such situations it is recommended that you use HTTP-based URLs, which are case-sensitive and rarely contain platform-dependent syntax. If you must use a network folder to hold the PDF, then it will only work consistently for one platform, and you may need to take steps to ensure that there are no variations within that platform that could result in different file name formats.

NOTE: Using a Web server as the location of the document to be reviewed can help reduce hashing problems associated with using a network folder as a comment repository.

Network Mapping for Windows

There are four ways in which Windows users can access a PDF file:

- Browsing to a file on a network server via **My Network Places**
- Using the **Command Prompt** window to access a folder
- Mapping a drive to the network server
- Retrieving the file from a Web server

Each of these methods has configuration issues which are discussed below:

1. Browsing **My Network Places**. Although this method may work for small workgroups or networks, it will not necessarily work correctly with large workgroups or networks. The reason is that the file path used in **My Network Places** may not always be the same for all users. This can happen even though the visual path to the server may appear the same for all users. For example, let's say that we have three users (Machines A, B and C) located in different areas of the globe, all commenting on a PDF file located on a Toronto, Canada file server. If Machine A is in Hamburg, Germany and its user browses to the Toronto file server, the IP routing underneath the visual display in Windows will not be the same IP route used by Machine B in Buenos Aires, Argentina, or Machine C in Sidney, Australia.
2. Using the **Command Prompt** window. Using the **Command Prompt** window has the same underlying issue as using **My Network Places**. For example, even though the visual path `\\MyTorontoFileServer\MyPDFFolder\MyPDF.pdf` may appear to be identical to multiple users, each machine may use a different underlying IP route.
3. Mapping a Network Drive. The network folder must be configured properly. First, the drive letter must be exactly the same for all users. For example, if Machine A is mapped to drive letter E, then all machines commenting on a PDF file located on that server must be also be mapped to drive letter E.

Next, the drive letter must also be mapped to exactly the same folder on the server for all of the machines, because the path statement must be identical for all users. For example, if Machine A is mapped to the E: Drive and the PDF file is directly located under the E: Drive (MyPDFFolder), its path statement would appear as follows:

```
\\MyFileServer\AllPDFFiles\MyPDFFolder\MyPDF.pdf  
E:\MyPDF.pdf
```

However, if Machine B is also the E: Drive, but its mapping is one folder level lower than the PDF file location (AllPDFFiles), its path statement would appear as follows:

```
\\MyFileServer\AllPDFFiles\MyPDFFolder\MyPDF.pdf  
E:\MyPDFFolder\MyPDF.pdf
```

The differing path statements in this example would result in two different hash folders.

Configuring a Network Folder Repository

Network Folder Platform Issues

4. *Web Server.* This is usually the best location for the PDF file. However, in this scheme the network folder is still used to store the comments. The reason why this works so well is that the URL or URI is a platform-independent path and therefore does not cause the problems associated with network browsing, using the **Command Prompt** window, or mapping a network drive.

Since this method still uses the file server location for storing comments, the procedures used for mapping a network drive still apply.

Network Mapping for Macintosh

NOTE: The following network mapping applies to Acrobat 5.0 and 7.0 clients.

For the Macintosh, the path statement must be the same for every Macintosh that accesses the drive.

For example, suppose Machine A uses the following path:

```
afp//SW2/MyMac/MyPDFFiles/MyPDFFolder/MyPDF.pdf
```

Then Machine B must also use the same path.

On an AppleTalk network, the same machine can appear in more than one AppleTalk Zone, so in this case the path could also appear as follows:

```
afp//SW3/MyMac/MyPDFFiles/MyPDFFolder/MyPDF.pdf
```

In addition, the user can share out multiple levels of folders on the same drive, meaning that the drive itself may be shared out more than once. For example, if the path **SW2/MyMac/MyPDFFiles** exists and has been mounted, it would appear with the volume name **MyPDFFiles**. The users could still browse to the same file using different paths.

As with the case for Windows, the best location for a PDF file that is to be reviewed is a Web server, but you would still use the network folder to store comments.

7

Configuring a Web Discussions Repository

The Microsoft Web Discussions Repository

Using Microsoft Web Discussions, users can create discussions, add messages to a discussion, and create new discussion threads. Web Discussions is essentially a threaded discussion managed by a server, and can be used to support comment and review workflows. This type of repository allows users to submit messages about a PDF file. Note that those comments cannot be saved to the PDF file; this type of workflow is purely Web-based. The threaded discussions are maintained on either a Web server that has Office Server Extensions (OSE) installed, or on a remote SQL Server.

NOTE: This repository type is only available to Windows users.

Prerequisites

The following are server requirements for configuring the Web Discussions repository:

- Microsoft Office Server Extensions or SharePoint Team Services must be installed on the server. These are available with several Microsoft server products, including Microsoft Office 2000, Office XP, or Office 2002.
- Either the Microsoft Management Console (MMC) or the Personal Web Manager (PWM for Windows NT 4.0) must be installed on the server.

The following are Web Discussions repository requirements for the client:

- Users must have Microsoft Office 2000, Office XP, or Office 2002
- Users must have Windows 98, Windows NT 4.0 (Service Pack 3 or higher), Windows Millennium Edition, Windows 2000 or higher, or Windows XP.
- Users must have Internet Explorer 5.0 or higher (or 4.0 and higher for NT 4 users) installed on their desktop computer.
- All reviewers must have access to the server to be used as the discussions server.

Manually Setting the Discussion Server Preference

If, for example, reviewers will be commenting on a document named:

`http://www.adobe.com/docs/doc.pdf`

the resulting comments will be held on the discussion server and associated with the URL:

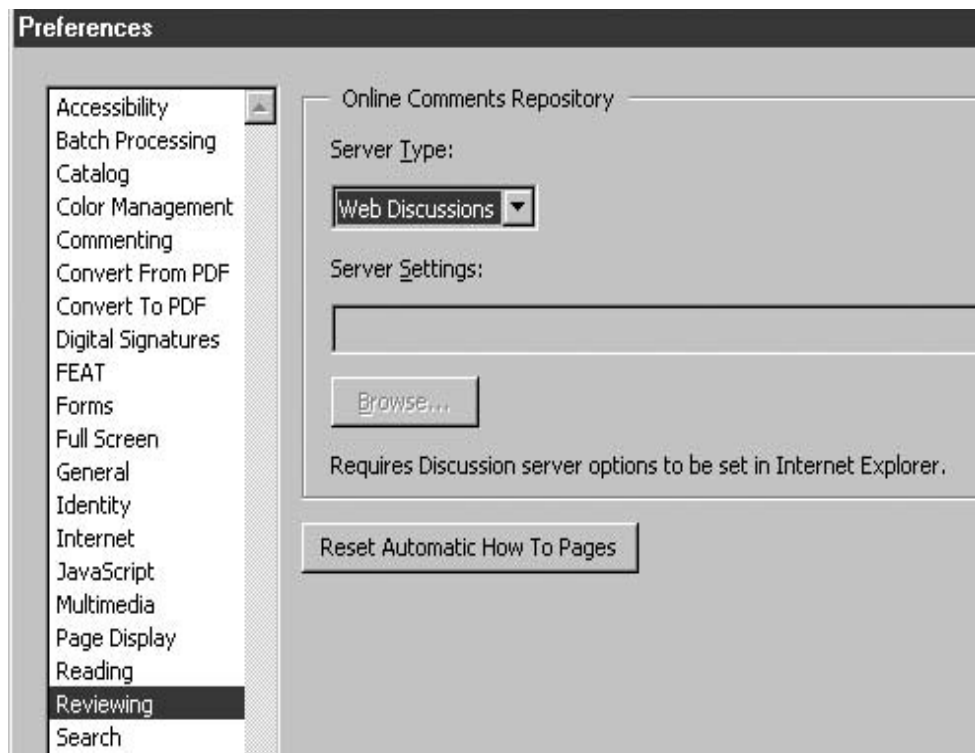
`http://www.adobe.com/docs/doc.pdf`

The procedure for setting the repository preference would be:

1. Start Acrobat, and select **Edit > Preferences > Reviewing**.
2. In the Server Type pop-up menu, select **Web Discussions**, and click **OK**.

Configuring a Web Discussions Repository

The Microsoft Web Discussions Repository



Next, you must configure Microsoft Internet Explorer:

3. Launch Internet Explorer to select the Web Discussions Server.
4. If the Web Discussion comments are not immediately visible in the Web browser, go to **View > Explorer Bar > Discuss**. Web Discussions will then appear at the bottom of the Web browser window or a Web Discussion Server dialog box will appear.
5. If the Add or Edit Discussion Servers dialog box appears, add the IP address of the Discussion Server. As stated before, the host name for the server can be used as well, however the IP address is the preferred method in case DNS fails.
6. Below the Discussion Server name you can add the **friendly name**; this is an option in which you may specify the host name. Note that this entry does not affect how the host name is resolved.

Automatically Setting the Web Discussions Preference

The store type for Web Discussions is WD. You may create an FDF file to configure the online comment server settings.

A sample script for setting the preferences is given below:

```
%FDF-1.2
%âãÏÓ
1 0 obj
<<
  /FDF
  <<
    /F (http://reviewserver/pdfs/settingsupdated.pdf)
    /JavaScript
    <<
      /After 2 0 R
    >>
  >>
endobj
2 0 obj
<<
>>
stream
  if(app.platform == "WIN")
    Collab.defaultStore = "WD";
  else
    app.alert("The Web Discussions annot store in only available
on Windows!");
endstream
endobj
trailer
<<
  /Root 1 0 R
>>
%%EOF
```

NOTE: For information on the Acrobat JavaScript **Collab** object, see the *Acrobat JavaScript Scripting Reference* and *Acrobat JavaScript Scripting Guide*.

In this sample script, <http://reviewserver/pdfs/settingsupdated.pdf> is the document that opens in the browser after the collaboration settings have been configured in Acrobat. You can modify the PDF file name and the store settings to adapt the script to your needs.

Configuring a Web Discussions Repository

The Microsoft Web Discussions Repository

Data Storage

The comment information is stored in strings, each of which is considered a discussion item. Acrobat extracts these comments from the Web Discussions server and presents them as a series of discussion threads.

Due to restrictions inherent to this repository type, comments containing binary attachments, such as sound or file attachments, are subject to the 64 KB size limitation of a discussion item. In such cases, Acrobat converts the binary stream into a hex stream, and, if necessary, splits it into smaller pieces to comply with the size requirement. Since Web Discussions does not guarantee the order in which the messages are returned, if the stream is separated into more than one discussion item, Acrobat ensures that the order of the stream remains intact by posting the first part of the comment as a top-level discussion thread, and all successive parts as replies to the previous ones.

Since users may not wish to see all of the comments associated with a page, Acrobat stores comments slightly off to one side.

Suppose the URL for the document is `http://www.adobe.com/docs/document.pdf`. In order for the user to see the messages that Acrobat stores on the Web Discussions server, the user would navigate to the following URL:

`http://www.adobe.com/docs/document.pdf/ACData`

Though this folder does not actually exist, Internet Explorer's discussion window will now show all the comments that Acrobat has stored on the Web Discussion server.

8

Configuring a Database Repository

The Database Repository

The Database repository type allows users to work directly with an SQL server database, making it a true enterprise solution for online collaboration. This repository type uses Acrobat DataBase Connectivity (ADBC), which facilitates interaction between Acrobat JavaScript and a database by providing a convenient interface to an Open Database Connectivity (ODBC) connection.

For more information on ADBC and details for client configuration, see the *Acrobat JavaScript Scripting Reference* and the *Acrobat JavaScript Scripting Guide*. For more information on ODBC, see Microsoft's technical documentation at <http://www.microsoft.com/data/odbc/>.

NOTE: The database repository type is not currently available for Macintosh users. Also, the SOAP plug-in for Acrobat 6.0 or higher provides support for SOAP-based communications in a database repository collaboration system (see [Configuring a SOAP Repository](#)).

Acrobat Database Connectivity (ADBC)

Through the usage of the Acrobat JavaScript **ADBC** object, you can take advantage of many useful properties and methods that make it possible to connect to a database, retrieve tables, and execute queries. In addition, the **ADBC** object can be used to obtain information about the databases available on the system, the tables contained within each database, and the data types used within any given table. This makes it possible to populate a PDF file with information, save field values in an Acrobat or XML form to a database, and add or update data as needed.

All of this can be accomplished without the use of a Web server or CGI script. The queries and manipulations can take place on a desktop PC/workstation. When combined with the powerful batch processing capability available in Acrobat 6.0 or higher, large-scale database tasks involving PDF files can be performed in a automated manner.

NOTE: Currently, ADBC is a Windows-only feature and requires Open Database Connectivity (ODBC) provided by Microsoft.

Configuring a Database Repository

The Database Repository

Supported Databases

Acrobat currently supports Microsoft SQL Server 2000, and Oracle 7, 8i, and 9i.

Prerequisites

The following are database repository requirements for the client:

- Users must have Microsoft Office 2000, 2003, or XP, and Microsoft Windows 98, Windows NT 4.0, Windows Millennium Edition, Windows 2000 or higher, or Windows XP.
- When using a Microsoft SQL server, an SQL driver must be present on the client machine. This can be added through the installation of Office 2000 or higher, Office XP, or MS SQL client software. When using an Oracle database, the Oracle client software must be installed on the client, and the **TNSNAMES.ORA** file must be installed in `%ORACLE_HOME\network\admin`.

Manually Setting the Database Repository Preference

Your department should configure and name the database to best suit your particular needs.

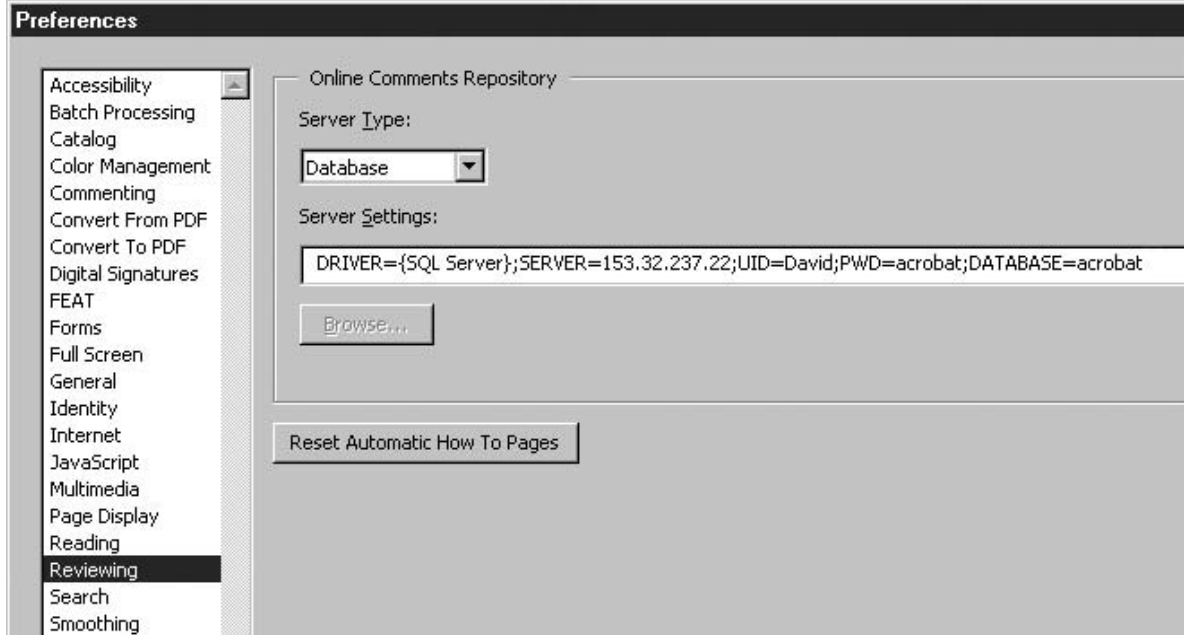
If users are to manually set the **Server Preferences**, the steps needed are similar to those used for the other preferences, as shown in the following example:

John Public and Jane Doe will be commenting on a document called

<http://www.adobe.com/docs/document.pdf>, and the resulting comments will be stored in a database called `CollabDB`.

To select the server:

1. Start Acrobat, and select **Edit > Preferences > Reviewing**.
2. Change the **Server Type** to *Database*, and type in the string that points to the database where the comments will be stored, as shown in the figure below:



NOTE: If the UID and PWD parameters in the dialog box shown above are left blank you will be prompted for a user id and password.

When the user creates comments and uploads them to the server, Acrobat creates a table in the database, using the MD5 hash of the URL, and saves each comment as a row in that table. Each row contains the following information:

- AUTHOR - the identity of the reviewer
- PAGE - the page number on which the comment is located
- NAME - an Acrobat-generated name
- CONTENTS - a JavaScript description of the comment
- DATA - binary data

Sound and file attachments are stored as binary large objects (BLOBs), whose size cannot exceed 2 GB per comment.

Configuring a Database Repository

The Database Repository

Automatically Setting the Database Preference

An FDF file can be created to set the online comment server settings. The following script sets the default string, using the store type of DB for database:

```
%FDF-1.2
%âãïó
1 0 obj
<<
    /FDF
    <<
        /F (http://reviewserver/pdfs/settingsupdated.pdf)
        /JavaScript
        <<
            /After 2 0 R
        >>
    >>
>>
endobj
2 0 obj
<<
>>
stream
    if (app.platform == "WIN")
        {Collab.defaultStore = "DB"; Collab.setStoreSettings("DB",
"DRIVER={SQL Server};SERVER=qui;UID=sa;PWD=;DATABASE=acrocollab");}
    else
        app.alert("The ADBC annot store is only available on
Windows!");
endstream
endobj
trailer
<<
    /Root 1 0 R
>>
%%EOF
```

NOTE: For information on `Collab.defaultStore` and `Collab.setStoreSettings`, see the *Acrobat JavaScript Scripting Reference* and the *Acrobat JavaScript Scripting Guide*.

In this sample script, **`http://reviewserver/pdfs/settingsupdated.pdf`** is the PDF file that opens in the browser after the collaboration settings have been set in Acrobat. To adapt this script to your needs, modify the PDF file name and the store settings.

9

Configuring a WebDAV Repository

The WebDAV Repository

The WebDAV (Web Distributed Authoring and Versioning) protocol can be used to send POST and GET commands for adding FDF comment file information to the collaboration server store. If a Windows server is used, it is the only type of repository that allows users whose machines have Microsoft 2000 or higher to store their comments on a Microsoft FrontPage Extended server or an FTP server.

To configure the client, specify the client type as **WebDAV** and the URL/URI for the comments store. While this client configuration is fairly simple, it can be problematic because of IT configuration and security concerns.

The IT configuration can be an issue if the company firewall does not support or permit the usage of the WebDAV protocol. Even though a firewall will normally block external access to the WebDAV server, you may configure WebDAV for internal usage if you bypass the proxy server by using the exceptions list (see [Bypassing The Proxy Server by Using an Exception](#)). Another option is to use a server that is open to external users.

NOTE: See <http://www.webdav.org> for more information on setting up your Web server to support WebDAV.

Supported Versions

Acrobat currently supports WebDAV RFC2518, along with Apache 1.3.14, 2.0 and higher, as well as IIS 5.0 and 6.0.

Prerequisites

The following are server requirements to configure the WebDAV repository:

- Apache 1.3.14 with mod_dav 1.0, Apache 2.0 or higher, or Microsoft IIS 5.0 or higher.
- IS department involvement is needed to ensure the collaboration environment is set up correctly.
- A site to store the comments is required. This location must be accessible to all the individuals commenting on the document.

The following are WebDAV repository requirements for the client:

- Users must have Apple Macintosh OS 9.0 or higher for Acrobat 5.0, OS 10.3 or higher and Panther are required for Acrobat 7.0, Windows 98, Windows NT 4.0, Windows Millennium Edition, Windows 2000 or higher, or Windows XP.
- To use a FrontPage extended repository or an FTP repository, users must have Microsoft Windows 2000 or higher, or Windows XP.
- To use any of the supported Apache or Microsoft IIS servers, the client may have any of the platforms listed above.

Configuring a WebDAV Repository

The WebDAV Repository

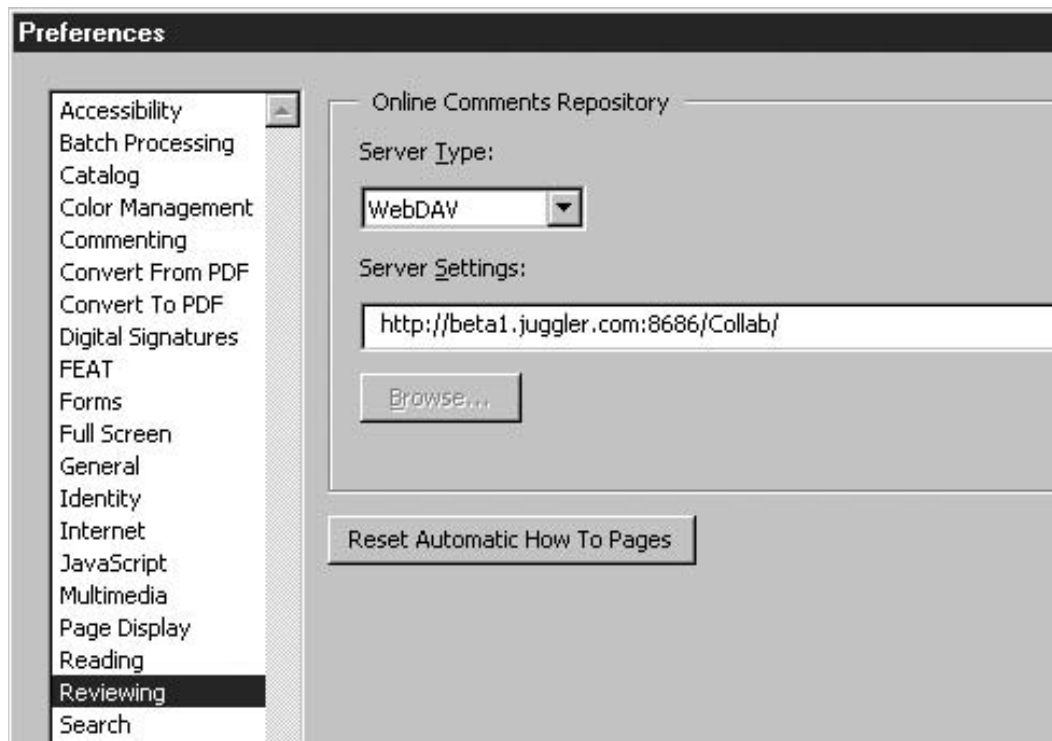
Manually Setting the WebDAV Preference

To configure the WebDAV repository, you must first determine the root directory for the comments, and then add DAV access. For example, to add DAV access to an Apache server with `mod_dav` installed, add the following code to `httpd.conf`:

```
<Location /Collab>  
  Dav On  
</Location>
```

To select the server:

1. Start Acrobat, and select **Edit > Preferences > Reviewing**.
2. Change the Server Type to **WebDAV**, and enter the location where the comments are to be stored as shown below:



On the client side, suppose John Public and Jane Doe are reviewing a document called `http://www.adobe.com/docs/document.pdf`, and that the comments will be stored at `http://beta1.adobe.com868/Collab`.

When they upload their comments to the server, Acrobat creates a hash folder and saves their comments at the following locations:

```
http://beta1.adobe.com868/Collab/UAtOJErFgHsL2xbLvPPJgC/jpublic.fdf
http://beta1.adobe.com868/Collab/UAtOJErFgHsL2xbLvPPJgC/jdoe.fdf
```

Automatically Setting the WebDAV Preference

The server must be set up to allow WebDAV access. See [Manually Setting the WebDAV Preference](#) for more information.

An FDF file can be created to configure the comment server settings. The following script sets the default store to `'ads, port 8383'` with `/Collab/` as the root, and uses the store type for WebDAV, which is `DAVFDF`:

```
%FDF-1.2
%ãÏÓ
1 0 obj
<<
  /FDF
  <<
    /F (http://reviewserver/pdfs/settingsupdated.pdf)
    /JavaScript
    <<
      After 2 0 R
    >>
  >>
endobj
2 0 obj
<<
  >>
stream
  Collab.defaultStore = "DAVFDF";
  Collab.setStoreSettings("DAVFDF", "http://ads:8383/Collab/");
endstream
endobj
trailer
<<
  /Root 1 0 R
>>
%%EOF
```

NOTE: For information on `Collab.defaultStore` and `Collab.setStoreSettings`, see the *Acrobat JavaScript Scripting Reference* and the *Acrobat JavaScript Scripting Guide*.

Configuring a WebDAV Repository

The WebDAV Repository

In this sample script, `http://reviewserver/pdfs/settingsupdated.pdf` is the PDF that opens in the browser after the collaboration settings have been set in Acrobat. To adapt this script to your needs, modify the PDF file name and the store settings.

Data Storage

Acrobat relies upon the underlying mechanism dealing with HTTP connectivity to determine if the server exists. If the DAV server is down or does not exist, the comments cannot be uploaded. Once the users have made the comments and attempt to upload them, Acrobat contacts the DAV server and receives a directory listing of the repository root using the command `PROFIND`. If the hashed name does not exist, Acrobat will attempt to create one using the command `MKCOL`. If the command fails, Acrobat will return an error message stating that the server is unavailable and that the comments have not been saved.

If the hashed directory already exists or the `MKCOL` command was successful, Acrobat will export the current user's comments to an FDF file. For example, if the current user is logged in as *JPublic*, Acrobat exports only those comments that are authored by JPublic to an FDF file named **JPublic.fdf**. This FDF file is then uploaded to the server using the `PUT` command. If the upload is successful, the **Send and Receive Comments** button will be disabled.

This set of commands only uploads the current user's comments to the server. The user must click either **Send and Receive Comments** or **Receive Comments** in order to view comments uploaded by other users. Acrobat does not retrieve the current user's set of comments from the server, since it is assumed that the user is not working at two different machines simultaneously.

Bypassing The Proxy Server by Using an Exception

If your proxy server does not support the WebDAV extension, it is possible to create an exception to the proxy server in order to bypass a firewall. This is accomplished by following the procedures described below:

NOTE: It is advised that you consult with your System/Network Administrator or IS department before changing the Web browser configuration.

Internet Explorer 6.0 for Windows:

1. Go to **Tools > Internet Options**.
2. In the **Internet Options** window, select the **Connections** tab.
3. In the **Local Area Network (LAN)** settings section, click **LAN Settings....**
4. In the Local Area Network (LAN) settings dialog box, within the Proxy server section, click **Advanced....**
5. In the Proxy Settings dialog box, within the **Exceptions** section, type the name of the WebDAV host in the **Do not use proxy server for the addresses beginning with** field.

Safari [1.0.3 - 1.2.3 (v125.9)] for Macintosh OS X 10.2 and higher, 10.3 and higher:

1. Go to **Safari > Preferences**.
2. Click **Advanced**.
3. Click **Safari: Change Settings....** This launches the **System Preferences** in the current profile.
4. In the **Proxies** section, enter the host name and IP address in the **Bypass proxy settings for these Hosts & Domains** field.
5. Click **Apply Now**.
6. Quit the **System Preferences**.
7. Close the **Safari Advanced Preferences** window.

Internet Explorer 5.0 and higher for Macintosh (Acrobat 5.0 only)

1. Go to **Edit > Preferences...**
2. Scroll down the Internet Explorer **Preference** dialog box in the left window pane to the **Network** section and select **Proxies**.
3. In the Internet Explorer Preference dialog box in the lower section that states **List the sites you want to connect to directly, by passing the proxies set above**, put a space or comma between each site, and type the name of the WebDAV host.
4. In the Internet Explorer **Preference** dialog box, click **OK**.
5. In the **Preferences** dialog box, click **OK**.

Configuring a WebDAV Repository

The WebDAV Repository

Creating a New Internet Profile on the Macintosh

Macintosh OS X 10.2 and higher, 10.3 and higher:

1. Go to **Apple > System Preference**.
2. Click on the **Network** icon.
3. Open the **Location** popup menu and select **New Location....**
4. In the **Name your new location:** window, type the name of the new location (such as WebDAV) and click **OK**.
5. In the **TCP/IP** section **Show** popup menu, select the information to be displayed in the user interface. For example, you could select Bluetooth, Built-in Ethernet, Airport, or Internal Modem.
6. In the **Configure** popup menu, select the type of configuration you will be using. For example, you could select DHCP, BootP, Manually, or DHCP with manual IP address. If your network requires it, you may need to provide more information for your client, such as DNS Servers, Client ID, or Search Domains.
7. Click **Apply Now**.
8. Quit the **System Preferences**.

WebDAV Client Configuration

Steps:

1. Launch Adobe Acrobat.
2. Go to **Edit > Preferences > General**.
3. Select **Reviewing**.
4. In the **Server Type** pop-up menu select WebDAV.
5. In the **Server Settings** field, type the URL/URI using the IP address of the Web server, as in the following example: `http://161.43.237.41/WebDAV/`.

NOTE: The ending back slash is imperative in the URL/URI or this address will not work, because this is the URL/URI naming convention for a WebDAV store. The host name can also be used; however, if DNS (Domain Naming Service) fails, the host will be unreachable. This applies to both Macintosh and Windows platforms.

6. In the **Preferences** dialog box, click **OK**.

10

Configuring a SOAP Repository

Acrobat support for the SOAP protocol enables developers to create customized commenting and review workflows using PDF and Acrobat. For more information on SOAP access from Acrobat, see the *Acrobat JavaScript Scripting Reference* and the *Acrobat JavaScript Scripting Guide*:

For more specific information on the SOAP protocol, see the W3C documentation at:

<http://www.w3.org/TR/SOAP>

The SOAP protocol supports comment repositories by providing the messaging framework that makes it possible to connect Acrobat with an HTTP-based service such as a Web service, thus enabling the exchange of annotation data between Acrobat and the server. Unlike the other commenting workflow methods, the developers setting up the workflow may choose how to store the annotation data in a variety of formats, such as a database, text file, or XML file.

SOAP Overview

SOAP is a W3C standard for exchanging structured information in a decentralized, distributed environment using XML. It is an extensible messaging framework containing a message construct that can be exchanged over a variety of underlying protocols. SOAP can also be used to communicate with Web Services described by the Web Services Description Language (WSDL: <http://www.w3.org/TR/wsdl>).

In Acrobat 7.0, which supports the SOAP 1.1 and 1.2 standards, Acrobat JavaScript code may use a WSDL proxy object to generate a SOAP envelope, which can be used to invoke web services on a remote server using either SOAP/RPC or document/literal encoding. There is support for both synchronous and asynchronous connections, and clients can use those connections to perform SOAP request/response transactions via the WSDL proxy object over HTTP or HTTPS.

Acrobat JavaScript SOAP Support

The Acrobat JavaScript **SOAP** object methods enable you to locate network services using DNS Service Discovery, bind a service name using a network address and port, connect to the service, and exchange information with the service through either synchronous or asynchronous communication. For more information on the **SOAP** object attributes and methods, see the *Acrobat JavaScript Scripting Reference* and the *Acrobat JavaScript Scripting Guide*.

Prerequisites for Configuration

To create a SOAP collaboration server, you will need to set up a Web server that supports the SOAP 1.1 or 1.2 protocol. You can configure a Java server using the Java 1.4 SDK and Apache Tomcat 4.0 or higher, or an ASP .NET server running IIS 6.0.

Each client must install Adobe Acrobat Standard or Professional 6.0 or higher, and must have a copy of the collaboration store code.

Manually Setting the SOAP Repository Preference

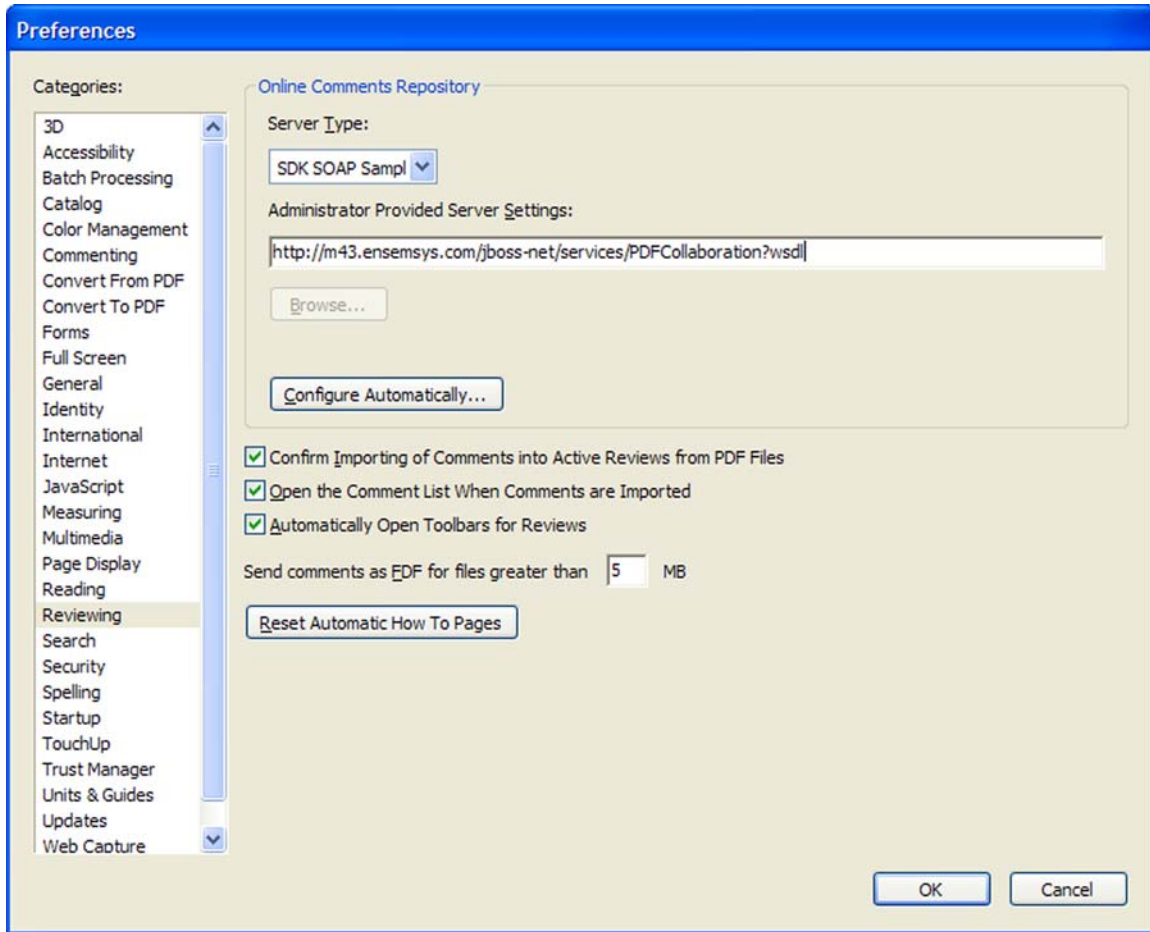
The Acrobat client can be configured through the use of the Acrobat JavaScript **SOAP.addAnnotStore** and **SOAP.setStoreSettings** methods. In addition, a sample JavaScript file (**sdkSOAPCollabSample.js**) that demonstrates the creation and configuration of a SOAP repository can be found in the Acrobat 7.0.5 SDK. This sample provides you with a good basis from which you can customize and extend your own SOAP commenting workflows.

The core functions in this JavaScript file are **update**, **enumerate**, and **complete**, as well as a functions you would define for reading data from the server and writing data to the server. Descriptions of these are provided in the *Acrobat JavaScript Scripting Reference* and the *Acrobat JavaScript Scripting Guide*.

As shown in the SDK sample, all annotation data for a PDF document can be composed in one text string and stored in one file on the server. The service should have two Remote Procedure Calls (RPCs): **readFile** and **writeFile**. The **readFile** method returns an XML string containing the annotation data that is automatically deserialized by Acrobat, and the **writeFile** method stores the annotation data in a file stored on the server and returns **true** if successful. Both methods accept a parameter that represents the name of the file on the server containing the annotation data, and the **writeFile** method has a second parameter that represents a string containing an array of annotations, which is automatically serialized by Acrobat. For more information on XML serialization and deserialization, see [XML Serialization and Deserialization](#).

With the sample JavaScript file installed in the **Acrobat JavaScripts** folder, a new annotation data store will be created using the `collab.addAnnotStore` method. After starting Acrobat, perform the following steps:

1. Select **Edit > Preferences > Reviewing**.
2. Change the **Server Type** to the display name of your SOAP Repository, and enter the URL for the SOAP HTTP service in the **Server Settings** text box, as shown in the figure below.



Automatically Setting the SOAP Repository Preference

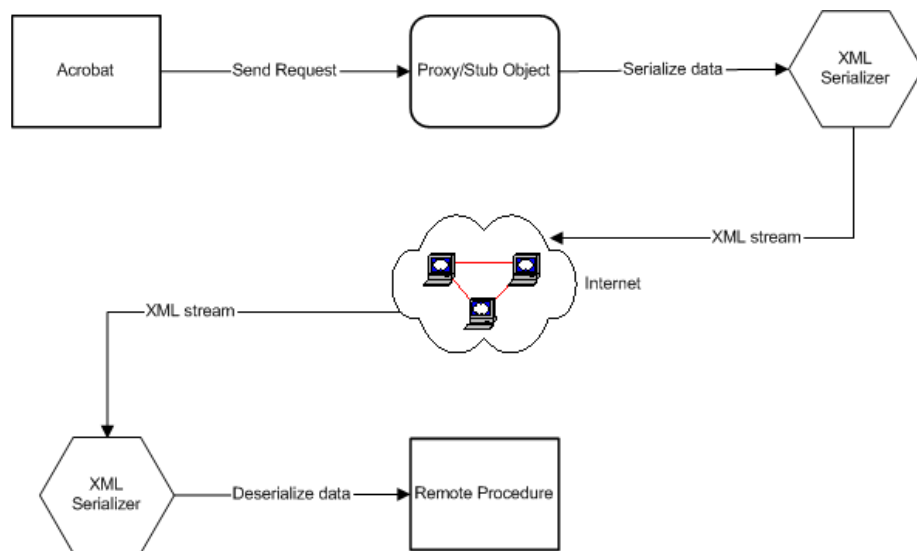
Similar to the WebDav Repository method, an FDF file can be created to set the online comment server settings found in the **Reviewing** panel in Acrobat **Preferences**.

Unlike WebDav, however, Acrobat does not have a predefined store type name for a SOAP repository, since it is a custom solution. It is defined in the JavaScript file stored in the **Acrobat JavaScripts** folder used to create the SOAP repository, and used when creating the FDF file containing the JavaScript to configure the store settings.

The store type is the second parameter in the `Collab.addAnnotStore` method. In the sample code given in the `sdkSOAPCollabSample.js` file, where a new SOAP Repository is being created, **SDK Soap Sample** is the store type.

XML Serialization and Deserialization

The SOAP protocol communicates via text-based XML messages. Therefore, before data can be sent across the Internet using SOAP, it must be converted, or serialized, to an XML stream format. Because XML is an open standard, XML streams can be processed by any application, including Acrobat.



When passing parameters to a remote procedure, Acrobat JavaScript data types are automatically bound to the relevant SOAP data types, and the binding holds true when the remote procedure returns data to the client. This is accomplished through the XML Schema Definition (XSD) language, which provides a standard set of data type definitions that can be exchanged in SOAP envelopes, and is what makes it possible for Acrobat and XML forms to work in an interoperable fashion with Web services. For more information, see the *Acrobat JavaScript Scripting Reference* and the *Acrobat JavaScript Scripting Guide*.

XML serialization is an important consideration in the performance of your workflows, and it is your choice as a developer to decide how the annotation data is stored. For example, you could use a database to store annotation information on the server. This approach would enable you to reduce the amount of XML data exchanged with the SOAP repository by exchanging only subsets of annotation data, known as *lightweight annotations*, with the server. A lightweight annotation contains just enough information to uniquely identify those annotations which must be updated. These contain only the page number, annotation name, author, annotation type, document id, and modification date.

A *heavyweight annotation* has binary data associated with it. Examples of these include **FileAttachment**, **Sound** and **Stamp** annotations. The sample JavaScript file (`sdkSOAPCollabSample.js`), provided with the Acrobat SDK and demonstrating SOAP connectivity between Acrobat and a SOAP HTTP service, currently supports only lightweight annotations. However, you could customize the code so that binary attachments are exchanged with the Web service, either in text format using base64 encoding, or in a more efficient manner using either the SOAP With Attachments (SwA) standard or the Message Transmission Optimization Mechanism (MTOM). This requires that your binary data be represented as a **ReadStream** object. For details, see the *Acrobat JavaScript Scripting Reference* and the *Acrobat JavaScript Scripting Guide*. It is recommended that the exchange of these heavyweight annotation types be minimized, as encoding large binary files can affect the performance of your application.

Configuring a SOAP Repository

Acrobat JavaScript SOAP Support

11

Offline Commenting

Working Offline

When an online collaboration session is either not possible or inconvenient, a reviewer can develop comments offline and upload them to the server at a later time. To work offline:

1. Open the document in a Web browser.
2. On the Commenting toolbar in the browser, click **Save and Work Offline**, and specify where the document should be saved.
3. Using the copy you just saved, open the document in Acrobat and begin working. When you are finished reviewing, save the document locally using exactly the same file name as the one used on the server. You may work on the document offline as many times as you wish, as long as you ensure that it is saved locally after each session.

NOTE: It is critical that users working offline do not change the file name of the document they are reviewing.

To upload your comments to the server:

1. Open your local copy of the collaboration document in Acrobat.
2. Click **Go Back Online** in the **Commenting** toolbar. The file will then open in your Web browser, and close in Acrobat.
3. When the document opens in the Web browser, your comments are automatically uploaded to the collaboration server, and comments from all other users are automatically downloaded from the server and appear in the document.

Offline Data Storage

When saving the collaboration document offline, Acrobat copies all of the comments from all users and stores them in the PDF file on the local disk. When the user clicks **Go Back Online** in the **Commenting** toolbar, only those comments authored by that user are uploaded to the server. Acrobat stores the information related to this procedure in Acrobat **Preferences**. On Microsoft Windows, the information is stored in the Windows registry, and on the Apple Macintosh it is stored in the **Preferences** folder located in the **System** Folder.

Offline Commenting

Offline Data Storage

12

Troubleshooting

Troubleshooting WebDAV Repositories

Problem:

Users are experiencing problems using the WebDAV collaboration server. A warning dialog box appears, stating that the collaboration server could not be contacted and that the user is unable to open the site using Internet Explorer.

Possible Cause:

The server preference settings are incorrect. The user may have entered the WebDAV URL manually and made an error, as shown in the following example:

`http://beta1.adobe.abc:8686/Collab`

when it should be

`http://beta1.adobe.com:8686/Collab/`

NOTE: The trailing slash must be used for it to work properly.

Solution:

The user should ensure that the server preferences have been specified correctly.

Problem:

Users are experiencing problems using the WebDAV collaboration server. A warning dialog box appears and states that the collaboration server could not be contacted. In this case, however, the user is able to browse to the site.

Possible Cause:

Acrobat uses special HTTP commands, called DAV verbs, to communicate with the WebDAV server. In some cases, the proxy server may not forward DAV verbs to the server.

Solution:

If the proxy server does not forward DAV verbs, the network settings must be changed to allow a direct connection to the collaboration server. This can be accomplished in one of three ways:

1. The IS department could create a PAC script which directs HTTP commands for the collaboration server to bypass the proxy, and then set the user's browser to use the PAC script.
2. Edit the user's Internet settings to bypass the proxy for the comments server.
3. Edit the user's Internet settings to use a direct connection to the Internet.

All Repository Types

Problem:

Acrobat is unresponsive after clicking **Upload Comments** or **Upload and Download Comments**.

Possible Cause:

Acrobat cannot connect to the server, possibly because the server is down.

Solution:

This action will eventually time out and Acrobat will display a dialog box stating that the server is unavailable. The server connections should be checked.

Problem:

Comments only appear intermittently. For example, a user participates in a review session in which the comments appear, and in future sessions observes that those comments are not always present.

Possible Cause:

This could be an IP translation problem. Sometimes a URL can translate to different IP addresses, depending on the configuration. For example: for the lookup of the address for **www.example.org**, the DNS server will sometimes return 192.239.92.41 and another time may return 208.185.132.75. Since Acrobat uses the IP address to create the MD5 hash of the URL, a different IP address results in an incorrect MD5 hash.

Solution:

The solution is to use the IP address when specifying the URL. In this case, for users who want to comment on:

```
http://www.example.org/mypdf.pdf
```

it would be best to use

```
http://192.239.92.41/mypdf.pdf
```


Problem:

Users do not see each other's comments, yet they are able to upload comments from their own machines successfully.

Possible Cause:

The problem may be how the users are accessing the PDF on the file server. It is a common practice on Windows to open PDF files in the browser rather than using the **File > Open** command in Acrobat. The user may surf to the file server using a UNC path such as `\\filesERVER\users\jsmith\document.pdf`, or they may mount a volume such as `\\filesERVER\users` to `k:` and open the file using `k:\jsmith\document.pdf`. However, these represent different files in Acrobat because of the different file paths, resulting in different MD5 hashes.

Solution:

For users to effectively comment on documents that exist on file servers, all users must enter the file name using exactly the same syntax, which is case-sensitive. The best solution to this problem is to use a Web page with a link to the PDF.

Problem:

Comments do not appear to be uploaded from an offline copy.

Possible Cause:

The user has opened the offline copy in the browser instead of Acrobat. Offline commenting is only possible within Acrobat.

Solution:

The user should upload the comments using the *Upload Comments* menu item.

Problem:

All server settings are correct and the server is active, yet the user continues to receive the following error message: "Your online comments server is currently unavailable."

Possible Cause:

If using either Apple Macintosh 9.0 or 9.1, the user name may be longer than the maximum of 31 characters allowed for a file name within those operating system versions.

Solution:

The user should shorten their user name to allow the PDF file name to conform to this limitation.

Troubleshooting

Troubleshooting WebDAV Repositories

Problem:

The **Upload Comments** button does not activate when double-clicking on an FDF file that was saved using **Comments > Export Selected** in the Web browser.

Solution:

Make a change to one of the comments and the button will activate.

Problem:

A message appears saying that the collaboration server is not available.

Solution:

1. Check the client configuration. This is your primary responsibility (see below for ways to check on this).
2. Check the user's network connectivity. This is something the user should have already checked.
3. Check the network configuration. For example, see if a proxy server or firewall is being used.
4. Check the client/server configuration.

Problem:

Users are having a problem with the client configuration.

Solution:

Check the following:

- Can other machines reach the server via Acrobat Reviewing?
- Are the machines on the same platform?
 - If not, are they using a network folder repository?
 - If not, are they trying to use a Windows-only feature on a Macintosh machine?
- Is the client pointing to the correct type of server?
- Is the client pointing to the correct collaboration store?
- Is the client configured to the host name or the host IP address?
 - If the client is configured to the host name, then try having the customer switch to the host IP address.
- If using WebDAV, has the customer configured the exceptions list for their Web browser?
- Does the exceptions list use both the host name and the host IP address?

Network Troubleshooting

Problem:

Users are having problems with the network.

Solution:

Check the following:

- Is the user connected to the network?
- If the client is configured to the host name address, then check to see if the user can 'ping' the host name.
- If the client is configured to the host IP address, then check to see if the user can 'ping' the host IP address.
- If the user cannot ping either host name or the host IP address, then it is a networking issue and the user's IT department should be contacted.
- If you would like to continue helping the user, you can run 'tracert' to the host name or host IP address of the server to determine where the connection fails.
- Is the user able to access the collaboration store via the Windows **Command** prompt, **My Network Places**, or the **Chooser**?
 - If not, it may be a connectivity issue, and the user's IT department should be contacted.
- Does the user's company use a proxy server?
- Does the proxy server support WebDAV?
- Have they configured the exceptions list for their Web browser?
- Does the exceptions list use both the host name and the host IP address?
- Is the user trying to access a server outside of their firewall?
 - If so, then it is a networking issue for the user's IT department.

Client/Server Configuration and Permissions Troubleshooting

Problem:

The user is having a problem with the Client/Server configuration and permissions.

Solution:

Check the following:

- Is the user able to access the collaboration store via the Windows **Command** prompt, **My Network Places**, the **Macintosh OS X Go** menu, or the **Macintosh Chooser** from that machine?
 - If not, it may be a permissions issue, and the user's IT department should be contacted.
- Can the user access the collaboration store from other machines via the Windows **Command** prompt, **My Network Places**, the **Macintosh OS X Go** menu, or the **Macintosh Chooser**?
 - If not, it may be a permissions issue, and the user's IT department should be contacted.
 - If so, it may be a 'Trust' issue.
- When using the database comments feature, does the user have a valid account with read, write and execute permissions on the database and table?
 - If not, or if the user does not know, the user's IT department should be contacted.
- The FDF file is being downloaded but will not execute from a Macintosh. Is the server's MIME configuration correctly set for the FDF file type?
- The user cannot access the network folder. Does the user have proper read, write and execute permissions on the server share?
- The user cannot access the WebDAV store on the Web server. Does the user have proper read, write and execute permissions on the server store?
- Does the network use a proxy server or firewall? Does it allow the WebDAV extensions to the HTTP 1.1 protocol?
- The user cannot access the server share. Is the server internal or external to the network firewall?