FDF Data Exchange Specification

February 8, 2007

Adobe Solutions Network — http://developer.adobe.com

FDF Data Exchange Specification

Adobe Systems Incorporated, 345 Park Avenue, San Jose, California 95110, USA.

FDF Data Exchange Specification

1.1 Overview

This specification describes the PDF language extensions that enable FDF (Forms Data Format) files to be used to send and share various types of data between users, system administrators, and servers. The use of FDF for transmitting form data and annotations is described in the *PDF Reference*.

FDF is typically used to encapsulate information such as X.509 certificates, directory settings, timestamp server settings, and embedded PDF files for network transmission. The format is based on the language syntax and encoding rules documented in the *PDF Reference*. The FDF mime type is registered by Acrobat® such that FDF files are always referred to Acrobat for processing. This referral is done by HTML browsers, email software, and by Windows and Apple operating systems.

This specification supports the following:

- Exchange of Self-Sign security certificates and requests for certificates, introduced in Acrobat 5.0.
- Signed FDF files, introduced in Acrobat 6.0 (signing of FDF is optional).
- Exchange of contact information, including certificates, introduced in Acrobat 6.0,
- Exchange of directory settings information, introduced in Acrobat 6.0,
- Exchange of Adobe Policy Server settings, introduced in Acrobat 7.0
- Exchange of Timestamp Server settings, introduced in Acrobat 7.0
- Exchange of ASSP Server settings, introduced in Acrobat 8.0

1.1.1 Compatibility

The following is a list of known compatibility issues; the information should be verified by anyone using it to build systems that rely on its accuracy.

General

PPK dictionaries require Acrobat 5.0.

Signatures require Acrobat 6.0 in order to be processed and are ignored in earlier versions.

Acrobat 4.05

For Acrobat 4.05 and earlier, an error message will display when an FDF file with a PPK dictionary is encountered.

Acrobat 5.x

If the **V** attribute in the PPK dictionary has a value 0x00030000 or greater, Acrobat 5.0 will display an alert with the text shown below. This is a bug in Acrobat 5.0, as it should have been a more informative error message.

If **V** is set to a value that can be processed by Acrobat 5.x, other errors may appear.

Certificates using RSA encryption key lengths greater than 1024 bits will result in the following error message in Acrobat 5.1:

> *(1821) Internal cryptographic error with certificate libraries.*

PPK dictionaries require Acrobat 5.0.

Acrobat 6.0

If Acrobat 6.0 encounters a document with a PPK dictionary **V** value greater than or equal to 0x0004000, the following alert text will be displayed:

> *Error encountered with Acrobat FDF Data Exchange file.*
> *Error. The version of this FDF Data Exchange File requires a later version of this application.*

1.1.2 References

PDF — *PDF Reference, sixth edition, Adobe Portable Document Format, Version 1.7*, Adobe Systems Incorporated

FDF — FDF Toolkit in the Acrobat SDK:
http://partners.adobe.com/public/developer/acrobat/devcenter.html

2.1 FDF Catalog

The FDF catalog dictionary is used for signed FDF files, which were introduced in Acrobat 6 (PDF 1.5). The signature syntax is described in the *PDF Reference*.

TABLE 2.1 Additional entries in the FDF catalog dictionary

KEY	TYPE	SEMANTICS
Sig	dictionary	(*Optional, always indirect; PDF 1.5*) A reference to a signature dictionary that contains a signature of the FDF file contents. The syntax for this signature dictionary is defined in the *PDF Reference*.
Ff	integer	(*Optional*) A set of flags with default value 0. The value for bits 0 and 1 are: **Value** **Action** 00 Acrobat will not delete the FDF file after processing (however a web or email browser may still delete the file). 01 Acrobat will automatically delete the FDF file after processing. 10 Acrobat will prompt the user to determine whether to delete the FDF file after processing (however a web or email browser may still delete the file).
M	date	(*Optional; PDF 1.5*) The modification date of the file.

3.1 FDF Object

FDF dictionaries can contain a PPK dictionary to allow certificate exchange (introduced in Acrobat 5.0).

TABLE 3.1 Additional entries in the FDF dictionary

KEY	TYPE	SEMANTICS
PPK	dictionary	(*Optional, must be an indirect reference; PDF 1.4*) A PPK dictionary (see section 4.1, "PPK Dictionary" below).

4.1 PPK Dictionary

The PPK dictionary contains certificates for exchange with another entity, or requests for certificates from another entity. The PPK dictionary was introduced in Acrobat 5.0.

TABLE 4.1 Entries in the PPK dictionary; PDF 1.4

KEY	TYPE	SEMANTICS
Type	name	(*Optional*) Always **PPK**.
V	integer	(*Required*) The version of this FDF file format. The value contains the major version number in the upper 16 bits, and the minor version number in the lower 16 bits. A change of major version numbers indicates a change that is not supported by old code. The values for major and minor version numbers are as follows: **Major** **PDF language version** 0x0002 PDF 1.4 (Acrobat 5.0) features 0x0003 PDF 1.5 0x0004 PDF 1.6 0x0005 PDF 1.7 **Minor** **Acrobat version** 0x0002 Acrobat 5.0 0x0005 Acrobat 6.0 0x0008 Acrobat 7.0 0x0009 Acrobat 8.0 The minor version allows Adobe to create beta versions of software with pre-release FDF uses that can later be rejected if they prove to be a format that will not be supported. The minor version is routinely incremented whenever the FDF generation code is changed.
R	integer	(*Optional*) The minimum revision of the Acrobat FDF PPK dictionary-processing code that is required to process this file. The number contains the major version number in the upper 16 bits, and the minor version number in the lower 16 bits. ***Note:*** *This attribute is deprecated, and the value of* **V** *should be used in its place; however it is required and used in Acrobat 5.0.*
Name	string	(*Optional; PDF 1.5*) The unauthenticated name of the person that generated this FDF file. If the FDF file is signed, the name should be extracted from the signature.
EMail	string	(*Optional; PDF 1.5*) A string containing the email address of the sender. If the FDF file is signed, the email address should be extracted from the signing certificate, if it is available in this certificate.
Request	dictionary	(*Optional*) A request dictionary that contains information regarding a request for information.

KEY	TYPE	SEMANTICS
Import	Array	(*Optional*) An array of import dictionaries that contains contacts that can be imported.
DirSettings	Array	(*Optional; PDF 1.5*) An array of directory dictionaries, each containing configuration information for a directory, for example an LDAP directory.
APS	Array	(*Optional; PDF 1.6*) An array of directory dictionaries, each containing the configuration information for a single Adobe Policy Server URL.
TimeStamp	Array	(*Optional; PDF 1.6*) An array of directory dictionaries, each containing the configuration information for a single timestamp server.
ASSP	Array	(*Optional; PDF 1.7*) An array of directory dictionaries, each containing the configuration information for a single ASSP server. ASSP servers are used for roaming credential signature services.
Ff	integer	(*Optional*) A set of flags with default value 0. The values for bit 0 are: **Value** **Action** 0 Acrobat will always delete the FDF after it has been processed. 1 Acrobat will not delete the FDF file after processing (however a Web or email browser may still delete the file).
M	Date	*(Optional*) Modification date of the file.
PacketID	String	(*Optional*) An identifier that, if provided, will be returned along with a certificate that is requested to be sent to a URL or email address. This acts as a cookie for the web server.

Example 4.1 PPK dictionary containing a request and import dictionary

```
3 0 obj
   <<
      /Type /PPK
      /V 0x00020002
      /R 0x00020000
      /Import 4 0 R
      /Request 5 0 R
      /M (D:20001010235959)
      /PacketID (0001)
      /Ff 1
   >>
endobj
```

5.1 Import Dictionary

The import dictionary contains a contact that can be imported into the recipient's address book.

TABLE 5.1 Common entries in an import dictionary; PDF 1.4

KEY	TYPE	SEMANTICS
Type	Name	(*Optional*) Always **Import**.
ContactInfo	string	(*Optional*) A string containing information that can be used to contact the originator of the FDF file. This includes information such as the phone number or email address of the originator of the FDF file. This can be used, for example, by software that imports certificates from the **CMS** attribute to verify the fingerprint (MD5 or SHA-1 hash) of the certificate. ***Note:*** *Use of the* **ContactInfo** *attribute is not a fully adequate mechanism for trusting a certificate, particularly a signature-enabled certificate. It is best if an independent, out-of-band mechanism be used to verify the certificate source. Receiving both the certificate and the human contact protocol (e.g. phone number) in the same FDF file amounts to self-validation and trust based on this pairing should be executed with care and understanding.*
cn	string	(*Optional; PDF 1.5*) The common name for the contact.
EMail	string	(*Optional; PDF 1.5*) A string containing the email address of the contact.
o	string	(*Optional; PDF 1.5)* A string containing the name of the organization, for example, "Acme Inc.," to which the contact belongs.
Contact	string	(*Optional; PDF 1.5)* A string containing the name of the contact, as it might be listed in an address book (for example, "John Smith <jsmith@example.com>"). The presence of this entry implies that this dictionary is for an address book entry, rather than a certificate with no associated address book information.
Group	string	(*Optional; PDF 1.5)* A string containing the name of a group, for example, "Managers," to which the contact belongs. This can be used when sharing address book information for groups that contain multiple contacts.
Trust	number	(*Optional; PDF 1.5)* An array of integers indicating the trust flags associated with this certificate. Each entry contains the trust associated with the certificate at the same index in the **Certs** array. The number of entries corresponds to the number of entries in the **Certs** array. The flags can be combined, although the kPSSigTrustIdentity flag must be defined or the other flags will be ignored and the certificate can only be used for chain building (not signatures). The trust flags are defined as follows:

KEY	TYPE	SEMANTICS

		`/* Trusted for signing (creating recipient signatures). This is assumed set by default in Acrobat 7 and above. */`
		`kPSSigTrustSigning 0x0001`
		
		`/* Trusted for authoring documents (creating author signatures) */`
		`kPSSigTrustAuthenticDocuments 0x0002`
		
		`/* Trusted for authoring documents with dynamic (multimedia) content */`
		`kPSSigTrustDynamicContent 0x0004`
		
		`/* Trusted for authoring documents with full access JavaScript */`
		`kPSSigTrustJavaScript 0x0010`
		
		`/* Trusted for identity. If this flag is not present other flags are ignored and certificate is only used for chain building */`
		`kPSSigTrustIdentity 0x0020`
		
		`/* Trusted as an anchor: no checks are done for certificates above this certificate */`
		`kPSSigTrustAnchor 0x0040`

Policy	dictionary	(*Optional; PDF 1.6)* An array of dictionaries containing certificate policy information. Each entry contains the certificate policy associated with the certificate at the same index in the **Certs** array. The number of entries corresponds to the number of entries in the **Certs** array. Each sub-dictionary contains two key/attribute pairs, where the keys are **PolicyOID** and **PolicyUFName**. The attribute for **PolicyOID** is an array of text OIDs in dotted notation. The attribute for **PolicyUFName** is a text description of the policies.
Certs	array of strings	(*Optional; PDF 1.5*) An array of certificates that may be imported. This attribute provides an alternative to the **CMS** attribute for embedding certificates; it supports security handlers that do not support CMS parsing. If **Certs** is present, the **CMS** attribute need not be present unless it is desired that Acrobat versions prior to 6.0 be able to process the file. If both **Certs** and **CMS** are present, they should contain the same list of certificates, and security handlers are free to choose either certificate list. There are no rules regarding this selection.
CMS	string	(*Optional*) A Certificate Message Syntax (CMS) object that contains a certificate or list of certificates that may be imported. CMS is equivalent to PKCS#7 and in this situation is used as a simple container for the certificate(s). See "PKCS#7 Signatures" in *PDF Reference*, version 1.7.

Example 5.2 An import dictionary

```
4 0 obj
   <<
     /Type /Import
     /ContactInfo (phone +1 408 536 6000)
     /CMS <3082019b06092a864886f70d0 .... aab43100>
   >>
endobj
```

6.1 Directory Dictionary

The directory dictionary contains configuration information that can be used to configure servers such as a directory server (e.g. LDAP), Adobe Policy Server, ASSP server, or timestamp server.

TABLE 6.1 Entries in a directory dictionary; PDF 1.5

KEY	TYPE	SEMANTICS
Type	Name	(*Optional*) Always **DirSettings**.
Subtype	Name	(*Optional; PDF 1.5*) A unique identifier that identifies the format of the data that is in the **Attr** dictionary. The defined formats are: **Adobe.PPKMS.LDAP.v0** PDF 1.5 **Adobe.APS.v0** PDF 1.6 **Adobe.Timestamp.v0** PDF 1.6 **Adobe.ASSP.v0** PDF 1.7
V	integer	(*Optional; PDF 1.5*) The version number of the data in the Attr dictionary, for the given Subfilter. Defaults to 0. The version number may be updated if there is new data in the attributes dictionary that is not compatible with older parsers that can process a given Subtype.
Filter	Name	(*Optional; PDF 1.5*) The language-independent name of the handler that created the data that is in this dictionary.
Name	String	(*Optional; PDF 1.5*) A language-dependent descriptive name of what this server connects to (e.g. *Adobe Employee LDAP*).
Attr	dictionary	(*Optional*) A dictionary containing the settings for a directory. There are no attributes in this dictionary except those that are defined for **Subtype**.

6.1.1 Attribute dictionary for Adobe.PPKMS.LDAP.v0 settings

TABLE 6.1 Attribute dictionary for Adobe.PPKMS.LDAP.v0 directory settings (PDF 1.5)

KEY	TYPE	SEMANTICS
IPAddress	string	(*Optional*) The IP Address of the LDAP server.
Port	number	(*Optional*) The port number of the LDAP server.

Example 6.1 A directory dictionary contain LDAP settings

```
4 0 obj
   <<
   /Type /DirSettings
      /Subtype /Example.PPKMS.LDAP.v0
      /Name (Example.com LDAP Server)
      /Filter /Example.PPKMS.ADSI
      /Attr
        <<
          /Port 39
          /IPAddress (122.31.255.14)
        >>
   >>
endobj
```

6.1.2 Attribute dictionary for Adobe.APS.v0 settings

TABLE 6.1 Attribute dictionary for Adobe.APS.v0 settings (PDF 1.6)

KEY	TYPE	SEMANTICS
URL	string	(*Optional*) The full URL of an Adobe Policy Server.

Example 6.2 An APS dictionary

```
4 0 obj
   <<
      /Type /DirSettings
      /SubType /Example.APS.v0
      /Name (Example.com QE Policy Server)
      /Filter /Example.APS
      /V 0
      /Attr
        <<
          /URL (https://example.com:443)
        >>
   >>
endobj
```

6.1.3 Attribute dictionary for Adobe.ASSP.v0 settings

TABLE 6.1 Attribute dictionary for Adobe.ASSP.v0 settings (PDF 1.7)

KEY	TYPE	SEMANTICS
URL	string	(*Optional*) The full URL of an ASSP server.

Example 6.3 An APS dictionary

```
4 0 obj
   <<
      /Type /DirSettings
      /SubType /Example.ASSP.v0
      /Name (Example.com Roaming ID Server)
      /Filter /Example.ASSP
      /V 0
      /Attr
        <<
          /URL (http://assp.example.com:4100)
        >>
   >>
endobj
```

TABLE 6.1 Attribute dictionary for Adobe.Timestamp.v0 settings (PDF 1.6)

KEY	TYPE	SEMANTICS
URL	string	(*Optional*) The full URL of a timestamp server.
Auth	name	(*Optional*) Indicates whether the server requires authentication. The only acceptable value is HTTP, indicating that HTTP authentication is required using **userid** and **password**.
Userid	string	(*Optional*) The user id required when authenticating to the server.
Password	string	(*Optional*) The password required when authenticating to the server. Note that this password is not protected by encryption when stored in the FDF file.

Note: *Acrobat does not export a user name and password in plain text for security reasons. However, if needed, the FDF file can be manually edited to add that field, which will facilitate setting up a time stamp server that requires the user to log on.*

Example 6.4 A timestamp server dictionary

```
4 0 obj
   <<
   /Type /DirSettings
   /SubType /Adobe.Timestamp.v0
   /Name (Example.com Timestamp Server)
   /V 0
   /Attr
      <<
        /URL (http://tsa.example.com:8080)
        /Auth /HTTP
        /Userid (jsmith)
        /Password (FOE432fder)
      >>
   >>
endobj
```

7.1 Request Dictionary

The request dictionary contains a request for information that is to be returned by the recipient.

TABLE 7.1 Entries in a request dictionary; PDF 1.4

KEY	TYPE	SEMANTICS
Type	name	(*Optional*) Always **Request**.
Subtype	name	(*Optional*) The type of information being requested. Currently supported values are **CMS**, to request a person's public key certificates, and **DirSettings**, to request directory configuration information. The default is **CMS**.
EMail	string	(*Optional*) The email address to which the request should be returned.
URL	string	(*Optional*) The URL to which the request should be returned. Supported URL protocols include **HTTP** and **mailto**. The format of the reply will be dependent on the **Subtype** of the request (see definition in *Javascript for Acrobat API Reference*).

Example 7.1 Request Dictionary containing a return email address

```
5 0 obj
   <<
     /Type /Request
     /EMail (jidentity@example.com)
   >>
endobj
```

Example 7.2 Request dictionary containing a return URL

```
5 0 obj
   <<
     /Type /Request
     /F
       <<
         /Type /FileSpec
         /FS /URL
         /F (http://www.example.com/cgi/script)
       >>
   >>
endobj
```

8.1 Implementation Notes on FDF Versions

8.1.1 Acrobat 6.0 Authoring (from 6.01 code)

By default, FDF is created with an Acrobat 5.0 compatibility set. The FDF will be set for Acrobat 6.0 compatibility if any of the following occur:

- an exported certificate is not compatible with Acrobat 5.0 (determined by using the `isCertAcrobat5Compatible()` method in the FDF Toolkit).
- directory settings are exported

If the FDF file can be opened with Acrobat 5.0, then

- **R** is set to 0x00020004 (`PUBSEC_FDF_A5_REQUIRED_VERSION`)
- **V** is set to 0x00020004 (`PUBSEC_FDF_CURRENT_VERSION2`)

If the FDF file will require Acrobat 6.0, then

- **R** is set to 0x00020015 (`PUBSEC_FDF_A6_REQUIRED_VERSION`),
- **V** is set to 0x00030005 (`PUBSEC_FDF_CURRENT_VERSION3`)

8.1.2 Acrobat 6.0 Consumption (from 6.01 code)

If the file format version (**V**) is any of the following:

- less than `PUBSEC_FDF_OLDEST_VERSION` (0x00020002)
- equal to 0x00020003 (Acrobat 6.0 beta software)
- in the range 0x00030000 to 0x00030004 (also Acrobat 6.0 beta software)

then the message "Error. The version of this FDF Data Exchange File is no longer supported." is displayed.

If the file format version (**V**) is greater than `PUBSEC_FDF_NEWEST_VERSION` (0x0003FFFF), then the message "Error. The version of this FDF Data Exchange File requires a later version of this application." is displayed.

The version of PubSec code in Acrobat 6.0 was 0x0002015. If the required PubSec code revision (**R**) is greater than 0x00020015 (`PUBSEC_FDF_A6_REQUIRED_VERSION`), then the message "Error. The version of this FDF Data Exchange File requires a later version of this application." is displayed.

8.1.3 Acrobat 7.0 Authoring

By default, FDF is created with an Acrobat 5.0 compatibility set. The FDF will be set for Acrobat 7.0 compatibility if any of the following occur:

- timestamp settings are exported
- Policy Server settings are exported

FDF will be set for Acrobat 6.0 compatibility if any of the following occur:

- an exported certificate is not compatible with Acrobat 5.0 (determined by using the `isCertAcrobat5Compatible()` method in the FDF Toolkit).
- exporting directory settings

If the FDF file can be opened with Acrobat 5.0, then

- **R** is set to 0x00020004 (`PUBSEC_FDF_A5_REQUIRED_VERSION`)
- **V** is set to 0x00020008 (`PUBSEC_FDF_CURRENT_VERSION2`)

If the FDF file will require Acrobat 6.0, then

- **R** is set to 0x00020015 (`PUBSEC_FDF_A6_REQUIRED_VERSION`),
- **V** is set to 0x00030008 (`PUBSEC_FDF_CURRENT_VERSION3`)

Note: *Some beta versions of Acrobat 7.0 set the **V** value to 0x00030006. Files with a **V** value of 0x00030006 should be rejected.*

If the FDF file will require Acrobat 7.0m, then

- **R** is set to 0x00020016 (`PUBSEC_FDF_A7_REQUIRED_VERSION`),
- **V** is set to 0x00040008 (`PUBSEC_FDF_CURRENT_VERSION4`)

8.1.4 Acrobat 7.0 Consumption

If the file format version (**V**) is any of the following:

- less than `PUBSEC_FDF_OLDEST_VERSION` (0x00020002),
- equal to 0x00020003 (Acrobat 6.0 beta software)
- in the range 0x00030000 to 0x00030004 (also Acrobat 6.0 beta software),
- equal to 0x00030006 (Acrobat 7.0 beta software)

then the message "Error. The version of this FDF Data Exchange File is no longer supported." is displayed.

If the file format version (**V**) is greater than `PUBSEC_FDF_NEWEST_VERSION` (0x0004FFFF), then the message "Error. The version of this FDF Data Exchange File requires a later version of this application." is displayed.

The version of PubSec code in Acrobat 7.0 was 0x0002016. If the required PubSec code revision (**R**) is greater than 0x00020016 (`PUBSEC_FDF_A7_REQUIRED_VERSION`), then the message "Error. The version of this FDF Data Exchange File requires a later version of this application." is displayed.

8.1.5 Acrobat 8.0 Authoring

By default, FDF is created with an Acrobat 5.0 compatibility set. The FDF will be set for Acrobat 8.0 compatibility if any of the following occur:

- ASSP Server settings are exported

FDF will be set for Acrobat 7.0 compatibility if any of the following occur:

- timestamp settings are exported
- Policy Server settings are exported

FDF will be set for Acrobat 6.0 compatibility if any of the following occur:

- an exported certificate is not compatible with Acrobat 5.0 (determined by using the `isCertAcrobat5Compatible()` method in the FDF Toolkit).
- exporting directory settings

If the FDF file can be opened with Acrobat 5.0, then

- **R** is set to 0x00020004 (`PUBSEC_FDF_A5_REQUIRED_VERSION`)
- **V** is set to 0x00020009 (`PUBSEC_FDF_CURRENT_VERSION2`)

If the FDF file will require Acrobat 6.0, then

- **R** is set to 0x00020015 (`PUBSEC_FDF_A6_REQUIRED_VERSION`),
- **V** is set to 0x00030009 (`PUBSEC_FDF_CURRENT_VERSION3`)[1]

If the FDF file will require Acrobat 7.0, then

- **R** is set to 0x00020016 (`PUBSEC_FDF_A7_REQUIRED_VERSION`),
- **V** is set to 0x00040009 (`PUBSEC_FDF_CURRENT_VERSION4`)

If the FDF file will require Acrobat 8.0, then

- **R** is set to 0x00020017 (`PUBSEC_FDF_A8_REQUIRED_VERSION`),
- **V** is set to 0x00050009 (`PUBSEC_FDF_CURRENT_VERSION5`)

8.1.6 Acrobat 8.0 Consumption

If the file format version (**V**) is any of the following:

- less than `PUBSEC_FDF_OLDEST_VERSION` (0x00020002)
- equal to 0x00020003 (Acrobat 6.0 beta software)
- in the range 0x00030000 to 0x00030004 (also Acrobat 6.0 beta software)
- equal to 0x00030006 (Acrobat 7.0 beta software)

then the message "Error. The version of this FDF Data Exchange File is no longer supported." is displayed.

---

1. Some beta versions of Acrobat 7.0 set this value to 0x00030006. Files with a **V** value of 0x00030006 should be rejected.

If the file format version (**V**) is greater than `PUBSEC_FDF_NEWEST_VERSION` (0x0005FFFF), then the message "Error. The version of this FDF Data Exchange File requires a later version of this application." is displayed.

The version of PubSec code in Acrobat 8.0 was 0x0002017. If the required PubSec code revision (**R**) is greater than 0x00020017 (`PUBSEC_FDF_A8_REQUIRED_VERSION`), then the message "Error. The version of this FDF Data Exchange File requires a later version of this application." is displayed.