

eFax Developer™

Universal User Guide for Inbound Processing

Revision Date: 03/27/2009

Contents

WELCOME	4
CONTENTS AT A GLANCE	4
PRODUCT INTRODUCTION	5
eFAX DEVELOPER™ INBOUND OVERVIEW	5
COMPONENTS	5
XML (<i>Extensible Markup Language</i>)	5
Base64 Encoding	5
Barcode Recognition with DataMatrix Barcode Symbology (<i>optional</i>)	5
Client Defined Field Handling (<i>optional</i>)	5
Page Splitting (<i>optional</i>)	6
XSD (<i>XML Schema Definition</i>)	6
eFax Developer™ Online Interface	6
UserName and Password validation	6
PRODUCT INTEGRATION.....	7
INTEGRATION REQUIREMENTS	7
INTEGRATION PLANNING	7
PROGRAMMING.....	7
CLIENT-SIDE PROGRAMMING OVERVIEW	7
XML ELEMENT MAPPING	8
Inbound POST Request XML	8

Notice

In all communications concerning this documentation, please refer to the revision date displayed on the front cover of this user's guide.

Copyright

The use, disclosure, reproduction, modification, transfer, or transmittal of this work for any purpose in any form or by any means without the written permission of j2 Global Communications, Inc. is strictly prohibited.

© 2006 j2 Global Communications, Inc. All rights reserved. j2® is a registered trademark of j2 Global Communications, Incorporated.

Welcome

Welcome to *eFax Developer's Universal User Guide for Inbound Processing*. This guide was designed to assist you in integrating *eFax Developer™* into your current inbound processing.

This documentation was designed to assist clients who are required to build their own client-side API.

Before continuing, please consider using an *eFax Developer™* SDK if applicable. Our Software Developer's Kit is designed to ease a client's overall development timeline while giving client applications full functionality as described within this document.

Users who wish to "cut to the chase" can skip directly to the Programming section as a quick-start.

Contents at a glance

- **Product Introduction** will provide an introduction to *eFax Developer™ Inbound*.
- **Product Integration** will provide information for integrating with *eFax Developer™*.
- **Programming** will provide essential information for the client-side programmer.

Product Introduction

eFax Developer™ Inbound Overview

eFax Developer™ Inbound is a delivery option that allows clients the ability to have inbound faxes forwarded via an HTTP POST request to a desired URL address. The POST will contain an XML-formatted argument representing data specific to the inbound fax received.

Clients will establish a procedure that will accept and process the request then return a response indicating that the POST request was successfully received.

Components

XML (Extensible Markup Language)

The XML-formatted data argument will contain client access and data fields specific to the inbound fax. This argument will also contain a Base64 encoded representation of the fax document received.

eFax Developer™ supports the XML 1.0 specification as recommended by the World Wide Web consortium (<http://www.w3.org/TR/REC-xml>). All XML messages must comply with this specification.

Base64 Encoding

Base64 encoding will convert the document's binary data into an ASCII text string.

An in depth RFC on *Base64 Content-Transfer-Encoding* can be found at the following addresses

- <http://www.freesoft.org/CIE/RFC/1521/7.htm>
- http://www.w3.org/Protocols/rfc1341/5_Content-Transfer-Encoding.html

Barcode Recognition with DataMatrix Barcode Symbology (optional)

Based on client account settings, fax images can be analyzed for barcodes. If this option is selected, the XML-formatted data will contain additional elements that hold the results of this analysis. Due to its reliability and robustness, the 2-dimentional DataMatrix symbology was chosen as our default.

Capable of encoding all 128 ASCII characters along with a number of different character sets, DataMatrix can accommodate up to 500 MB per square inch with a data capacity of 3,116 digits or up to 2,335 ASCII characters. DataMatrix barcodes have a high degree of redundancy and resists printing defects which make them the perfect match for fax image processing.

Client Defined Field Handling (optional)

Based on a client's requirements, *eFax Developer Inbound* may be setup to detect one-to-many pre-established client defined fields. If this option is selected, the XML-formatted data will contain additional elements that hold each client defined field name and its dynamic value. Clients that desire further information regarding this special feature should contact eFax Developer's Customer Support Group.

Page Splitting (optional)

Based on client account settings, a multi-page fax image can be split out and returned as multiple single-page fax images. Clients who choose this option will receive XML-formatted data containing elements representing each page as a separate Base64 encoded container. Clients that desire further information regarding this special feature should contact eFax Developer's Customer Support Group.

XSD (XML Schema Definition)

To insure data integrity, clients may wish to validate inbound requests against an XSD schema. A sample schema is provided as part of this SDK for clients who wish to perform XML validation.

***eFax Developer*TM Online Interface**

Clients can change various inbound settings through *eFax Developer*TM online by clicking the "Settings" icon located on any page of the interface. Go to https://secure.efaxdeveloper.com/EFax_Login.serv to log into the *eFax Developer*TM online interface.

UserName and Password validation

*eFax Developer*TM will always send a username and password as part of the XML POST. As an additional security measure, clients may change their XSD appropriately to validate the username and password. Clients that perform username and password validation will be responsible for keeping their XSD in sync with any username and password changes made online via the *eFax Developer*TM online interface.

Product Integration

Integration Requirements

To integrate *eFax Developer™ Inbound* functionality with your application, you must develop custom code that will accept a standard HTTP POST request of XML-formatted data.

- You will need to know how to decode XML documents. There are a number of tools, tutorials and editors freely available to assist you as needed.
- You will need to know how to decode a Base64 encoded string. There are a number of tools freely available to assist you with Base64 encoding.

Integration Planning

To successfully integrate *eFax Developer™ Inbound* functionality, it is important to have the appropriate tools as well as a project plan prior to attempting integration. The following is provided to assist you in the integration planning process.

- Carefully review this documentation.
- Determine any requirements for barcode recognition and/or client defined field handling.
- Contact *eFax Developer™* Support to set up your account. Optional barcode recognition and/or client defined field handling should be discussed if desired.
- Obtain software for XML decoding.
- Obtain software for Base64 decoding.
- Develop your application.
- Log into the *eFax Developer™* online interface to set your inbound settings as desired.
- Test your application's integration with *eFax Developer™*.
- Make your application live.

Programming

Client-Side Programming Overview

The following is provided as a high-level overview of expected client-side processing.

- Create an application that accepts a URL-encoded HTTP(S) POST containing a single name/value pair where the parameter name equals "xml" and the parameter value contains the XML-formatted data. Process the data as desired then return an HTML response of "Post Successful" back to *eFax Developer™*.

XML Element Mapping

Inbound POST Request XML

Element	Required/Cardinality	Type	Length	Description/Values
InboundPostRequest/ Access Control	Yes/One	Container	N/A	Access Control Container
InboundPostRequest/ AccessControl/ UserName	No/One	Alphanumeric	20	User Name
InboundPostRequest/ AccessControl/ Password	No/One	Alphanumeric	20	User Password
InboundPostRequest/ RequestControl	Yes/One	Container	N/A	Request Control Container
InboundPostRequest/ RequestControl/ RequestDate	Yes/One	Alphanumeric	19	The XML generation date in “MM/dd/yyyy HH:mm:ss” (24 hour) format. Pacific Time Zone
InboundPostRequest/ RequestControl / RequestType	Yes/One	Alphanumeric	15	“New Inbound” “Manual Repost” The type of request that generated the XML POST.
InboundPostRequest / FaxControl	Yes/One	Container	N/A	Fax Control Container
InboundPostRequest/ FaxControl/ AccountID	Yes/One	Alphanumeric	10	<i>eFax Developer</i> TM account identifier.
InboundPostRequest/ FaxControl/ NumberDialed	No/One	Alphanumeric	10	The fax number that was dialed.
InboundPostRequest / FaxControl/ DateReceived	Yes/One	Alphanumeric	19	The original fax received date in “MM/dd/yyyy HH:mm:ss” (24 hour) format. Pacific Time Zone
InboundPostRequest / FaxControl / FaxName	Yes/One	Alphanumeric	50	The actual fax name given to this fax by <i>eFax Developer</i> TM as changed by the client.

Element	Required/Cardinality	Type	Length	Description/Values
InboundPostRequest / FaxControl / FileType	Yes/One	Alphanumeric	3	“pdf” “tif” The document type as stored based on the “Inbound File Format” setting.
InboundPostRequest / FaxControl / PageCount	Yes/One	Numeric	5	The number of pages received.
InboundPostRequest / FaxControl / CSID	Yes/One	Alphanumeric	50	The station identifier, when supplied by the receiving fax machine upon successful transmission.
InboundPostRequest / FaxControl / ANI	Yes/One	Alphanumeric	25	The automatic number identification (caller id) contains the calling party’s fax number.
InboundPostRequest / FaxControl / Status	Yes/One	Numeric	5	Numeric field indicating the fax status. “0” indicates a successful transmission while all other values indicate an error code which can be cross-referenced with an <i>eFax Developer</i> ™ supplied table.
InboundPostRequest / FaxControl / MCFID	Yes/One	Numeric	8	<i>eFax Developer</i> ™ fax ID.
InboundPostRequest/ FaxControl/ UserFieldControl	No/Zero or One	Container	N/A	User Field Control Container <u>This parent container and its siblings may or may not exist based on the client’s account settings.</u>
InboundPostRequest/ FaxControl/ UserFieldControl/ UserFieldsRead	Yes/One	Numeric	5	The total number of fields. This count reflects the number of <i>UserField</i> containers found within the XML POST.
InboundPostRequest/ FaxControl/ UserFieldControl/ UserFields	Yes/One	Container	N/A	User Fields Container <u>This container will hold one-to-many <i>UserField</i> containers depending on the number of client defined fields specified.</u>

Element	Required/Cardinality	Type	Length	Description/Values
InboundPostRequest/ FaxControl/ UserFieldControl/ UserFields/ UserField	No/Zero or Many	Container	N/A	User Field Container <u>This container will hold the name and value for each client defined field.</u>
InboundPostRequest/ FaxControl/ UserFieldControl/ UserFields/ UserField/ FieldName	Yes/One	Alphanumeric	100	Client defined field name
InboundPostRequest/ FaxControl/ UserFieldControl/ UserFields/ UserField/ FieldValue	Yes/One	Alphanumeric	100	An associated value for the client defined field.
InboundPostRequest / FaxControl/ BarcodeControl	No/Zero or One	Container	N/A	Barcode Control Container <u>This parent container and its siblings may or may not exist based on a client's account settings.</u>
InboundPostRequest / FaxControl/ BarcodeControl/ BarcodesRead	Yes/One	Numeric	5	The total number of barcodes. This count reflects the number of <i>Barcode</i> containers found within the XML POST.
InboundPostRequest / FaxControl/ BarcodeControl / Barcodes	Yes/One	Container	N/A	Barcodes Container <u>This container will hold one-to-many <i>Barcode</i> containers depending on the number of barcodes found within the physical document.</u>
InboundPostRequest / FaxControl/ BarcodeControl / Barcodes/ Barcode	No/Zero or Many	Container	N/A	Barcode Container

Element	Required/Cardinality	Type	Length	Description/Values
InboundPostRequest / FaxControl/ BarcodeControl/ Barcodes/Barcode/ Key	Yes/One	Alphanumeric	ANY	The interpreted barcode value
InboundPostRequest / FaxControl/ BarcodeControl/ Barcodes/Barcode/ AdditionalInfo	Yes/One	Container	N/A	Additional Info Container
InboundPostRequest / FaxControl/ BarcodeControl/ Barcodes/Barcode/ AdditionalInfo/ ReadSequence	Yes/One	Numeric	3	A per-page sequence number given to each barcode as it is read on a given page.
InboundPostRequest / FaxControl/ BarcodeControl/ Barcodes/Barcode/ AdditionalInfo/ ReadDirection	Yes/One	Alphanumeric	50	<p>“2-Dimentional” “Left/Right” “Top/Bottom” “Right/Left” “Bottom/Top”</p> <p>The read direction used to interpret the barcode stored within this container.</p>
InboundPostRequest / FaxControl/ BarcodeControl/ Barcodes/Barcode/ AdditionalInfo/ Symbology	Yes/One	Alphanumeric	30	The symbology or protocol used to generate the barcode stored within this container.
InboundPostRequest / FaxControl/ BarcodeControl/ Barcodes/Barcode/ AdditionalInfo/ CodeLocation	Yes/One	Container	N/A	Code Location Container
InboundPostRequest / FaxControl/ BarcodeControl/ Barcodes/Barcode/ AdditionalInfo/ CodeLocation/ PageNumber	Yes/One	Numeric	5	The physical page location of the barcode stored within this container.

Element	Required/Cardinality	Type	Length	Description/Values
InboundPostRequest / FaxControl/ BarcodeControl/ Barcodes/Barcode/ AdditionalInfo/ CodeLocation/ PageCoordinates	Yes/One	Container	N/A	Page Coordinates Container
InboundPostRequest / FaxControl/ BarcodeControl/ Barcodes/Barcode/ AdditionalInfo/ CodeLocation/ PageCoordinates/ StartEdge	Yes/One	Container	N/A	Starting Edge Container
InboundPostRequest / FaxControl/ BarcodeControl/ Barcodes/Barcode/ AdditionalInfo/ CodeLocation/ PageCoordinates/ StartEdge/ XStartPointA	Yes/One	Number	8	Coordinate grid x-axis for starting edge point A.
InboundPostRequest / FaxControl/ BarcodeControl/ Barcodes/Barcode/ AdditionalInfo/ CodeLocation/ PageCoordinates/ StartEdge/ YStartPointA	Yes/One	Number	8	Coordinate grid y-axis for starting edge point A.
InboundPostRequest / FaxControl/ BarcodeControl/ Barcodes/Barcode/ AdditionalInfo/ CodeLocation/ PageCoordinates/ StartEdge/ XStartPointB	Yes/One	Number	8	Coordinate grid x-axis for starting edge point B.

Element	Required/Cardinality	Type	Length	Description/Values
InboundPostRequest / FaxControl/ BarcodeControl/ Barcodes/Barcode/ AdditionalInfo/ CodeLocation/ PageCoordinates/ StartEdge/ YStartPointB	Yes/One	Number	8	Coordinate grid y-axis for starting edge point B.
InboundPostRequest / FaxControl/ BarcodeControl/ Barcodes/Barcode/ AdditionalInfo/ CodeLocation/ PageCoordinates/ EndEdge	Yes/One	Container	N/A	Ending Edge Container
InboundPostRequest / FaxControl/ BarcodeControl/ Barcodes/Barcode/ AdditionalInfo/ CodeLocation/ PageCoordinates/ EndEdge/ XEndPointA	Yes/One	Number	8	Coordinate grid x-axis for ending edge point A.
InboundPostRequest / FaxControl/ BarcodeControl/ Barcodes/Barcode/ AdditionalInfo/ CodeLocation/ PageCoordinates/ EndEdge/ YEndPointA	Yes/One	Number	8	Coordinate grid y-axis for ending edge point A.
InboundPostRequest / FaxControl/ BarcodeControl/ Barcodes/Barcode/ AdditionalInfo/ CodeLocation/ PageCoordinates/ EndEdge/ XEndPointB	Yes/One	Number	8	Coordinate grid x-axis for ending edge point B.

Element	Required/Cardinality	Type	Length	Description/Values
InboundPostRequest / FaxControl/ BarcodeControl/ Barcodes/Barcode/ AdditionalInfo/ CodeLocation/ PageCoordinates/ EndEdge/ YEndPointB	Yes/One	Number	8	Coordinate grid y-axis for ending edge point B.
InboundPostRequest / FaxControl/ FileContents	No/One	Base64 encoded text	ANY	Base64 encoded file contents. <u>Clients using the page splitting option will NOT receive this container.</u>
InboundPostRequest / FaxControl/ PageContentControl	No/One	Container	N/A	Page Content Container <u>Clients using the page splitting option WILL receive this container and its embedded siblings.</u>
InboundPostRequest / FaxControl/ PageContentControl/ Pages	Yes/One	Container	N/A	Pages Container <u>This container will hold one-to-many page containers depending on the number of physical pages found in the fax document.</u>
InboundPostRequest / FaxControl/ PageContentControl/ Pages/ Page	No/Zero or Many	Container	N/A	Page Container
InboundPostRequest / FaxControl/ PageContentControl/ Pages/ Page/ PageNumber	Yes/One	Numeric	5	The physical page number within the fax document.
InboundPostRequest / FaxControl/ PageContentControl/ Pages/ Page/ PageContents	Yes/One	Base64 encoded text	ANY	Base64 encoded contents for a physical page.