# DocuSign Publisher Service

# HTTPS Post Guide

Mike Borozdin, Tom Gonser  
Updated - 11/13/2008

Version 1.6

This overview provides a developer or business analyst with high level information about how the DocuSign Publish service works and discusses the data elements that are available.

Table of Contents

[DocuSign Publisher Service 1](#_Toc209863928)

[HTTPS Post Guide 1](#_Toc209863929)

[DocuSign Publisher Service 3](#_Toc209863930)

[Introduction 3](#_Toc209863931)

[Setup 3](#_Toc209863932)

[Setting up the account 4](#_Toc209863933)

[Solution Architecture 6](#_Toc209863934)

[XML Information Structure 7](#_Toc209863935)

[Key Transaction Elements 7](#_Toc209863936)

[Form Field data 8](#_Toc209863937)

[Technical Details 9](#_Toc209863938)

[Running the Service 9](#_Toc209863939)

[Best Practices 9](#_Toc209863940)

# DocuSign Publisher Service

## Introduction

The DocuSign Publisher service enables instant data updates to be sent to external applications generated by user transactions as they progress through to completion. The Publisher Service provides update information about the status of these transactions, and is also capable of providing updates that include the actual content of documents containing form fields. Publisher is useful to organizations that would like a real-time view into all the transactions across their user base in a centralized location, and which can be customized to drive reporting or workflow specific to that business’s needs.

In order to take advantage of this feature, the external application must have established an HTTPS ‘listener’ which can be accessed from the DocuSign service. This ‘listener’ is an application which will accept XML transactions sent from the DocuSign service as the events happen. This interface is not a SOAP API such as the other interfaces in the DocuSign system.

## Setup

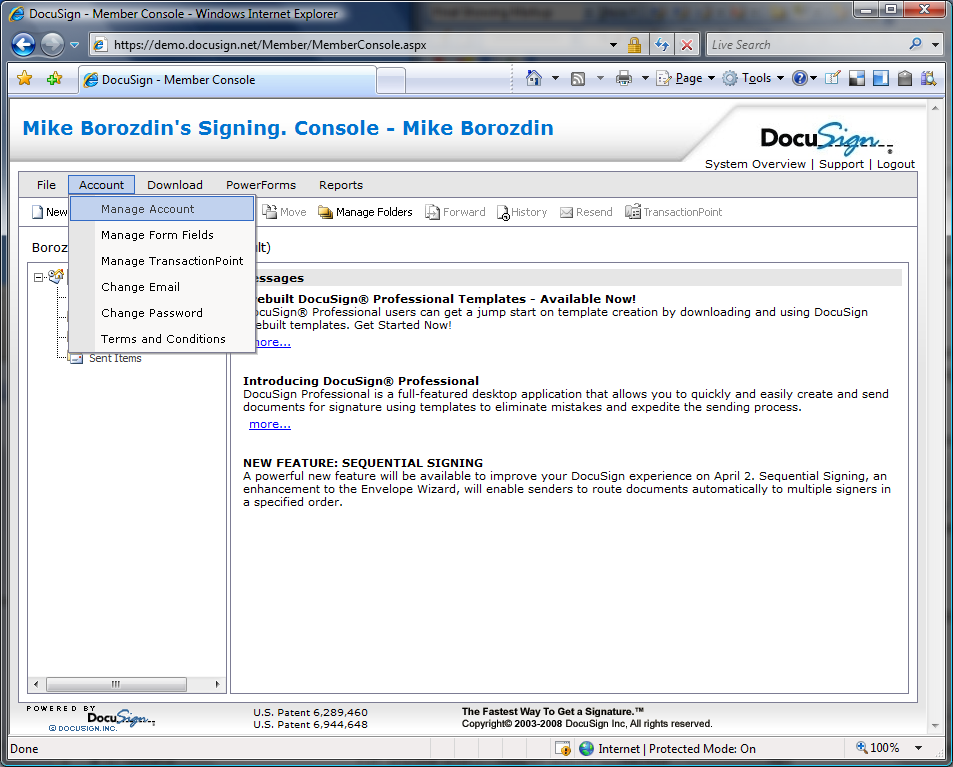
In order to use the DocuSign Publisher solution, the DocuSign account must be enabled. It is not enabled by default. Once enabled, the Publisher Interface can be accessed via the DocuSign Admin section of the Console.

At a high level, the following steps must be taken to use the Publisher Service:

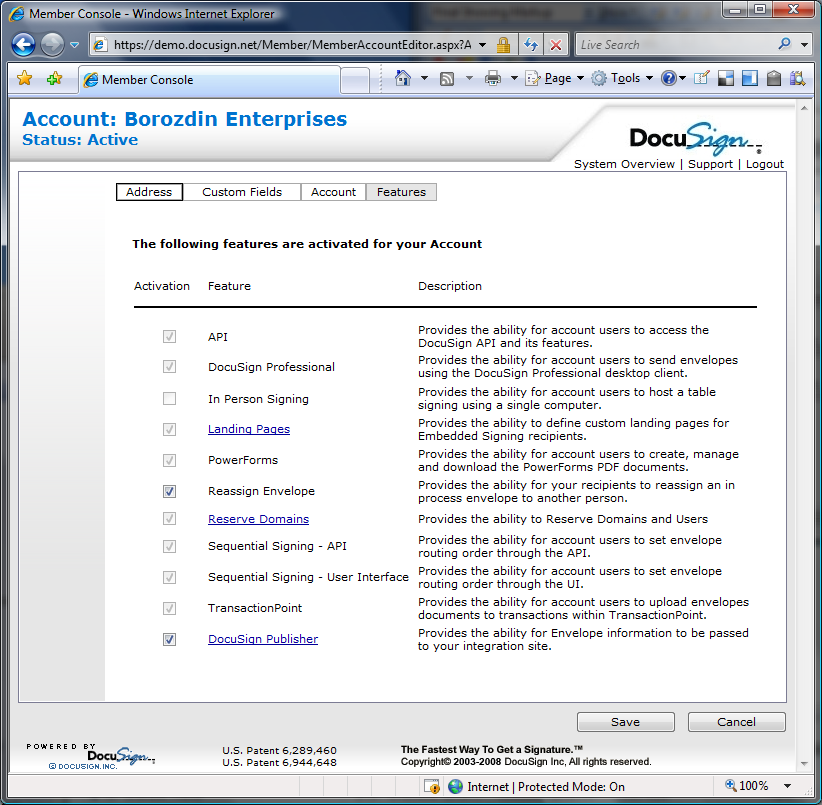
1. Request your account be configured to publish transaction updates. Your DocuSign Account Manager can help you with this step.
2. Define what events you wish to use as triggers for updating the information. Events can include several items such as document sent, viewed, signed, completed, etc.
3. Develop an understanding of the XML data that will be sent from the DocuSign Publisher solution to your application.
4. Develop an application that can accept data at your HTTPS location that can parse the inbound XML data and utilize it. This is a web application written specifically for your business.

### Setting up the account

In order to access the Publisher setup screen, log into your DocuSign account as an Account Administrator and go to Account/Manager Account menu:

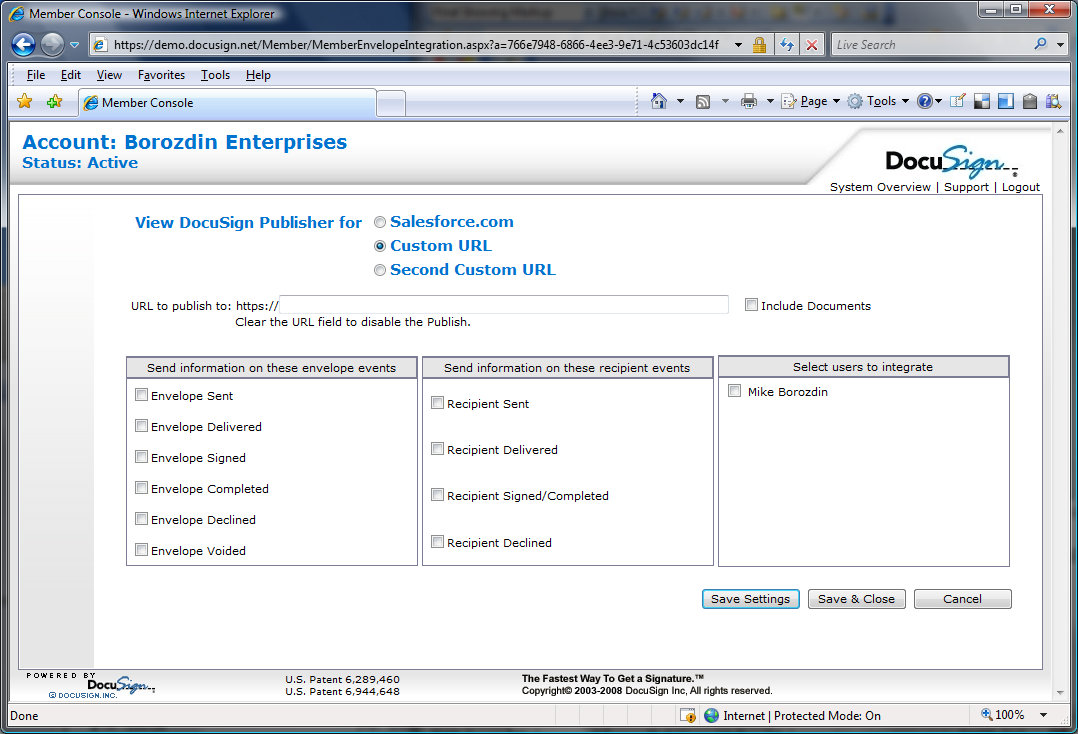


On the features Tab you will find DocuSign Publisher checkbox and a link:



As shown above, this account is enabled for Publishing, but the administrator can turn this feature on or off by selecting or de-selecting this checkbox.

Next you will need to click on the DocuSign Publisher hyperlink which will display the main Publisher screen.



On this screen you will be able to select the events and the landing URL for the DocuSign Publishers. There are 3 Publishers that can be configured, one specific to Salesforce.com, and two others for custom application connections. (For information specific to the Salesforce.com Publisher, please see the setup for that Publisher.)

The elements on this setup screen are as follows:

* Landing URL – this is the web address of your listener. Enter only the URL WITHOUT the “HTTPS:// “
* Trigger Events – Events from your DocuSign service can be triggered upon several events such as ‘envelope events’ – events which happen when envelopes are reviewed, and recipient events, events triggered by recipients. The primary difference between envelope events and recipient events is that recipient events trigger with each recipient and envelope events only trigger when envelope status changes.
* Select User Accounts – a list of all users associated with your account also appears. You may select to track events from any or all of them.

Notice also that you have the option of sending information to a SalesForce account, and two URL accounts. These may ALL be used or just one or another. To change views select the radio button.

The ‘Include Documents’ box instructs the service to send the PDF document along with the update XML. If you do not wish to receive the document with the updates, de-select this box.

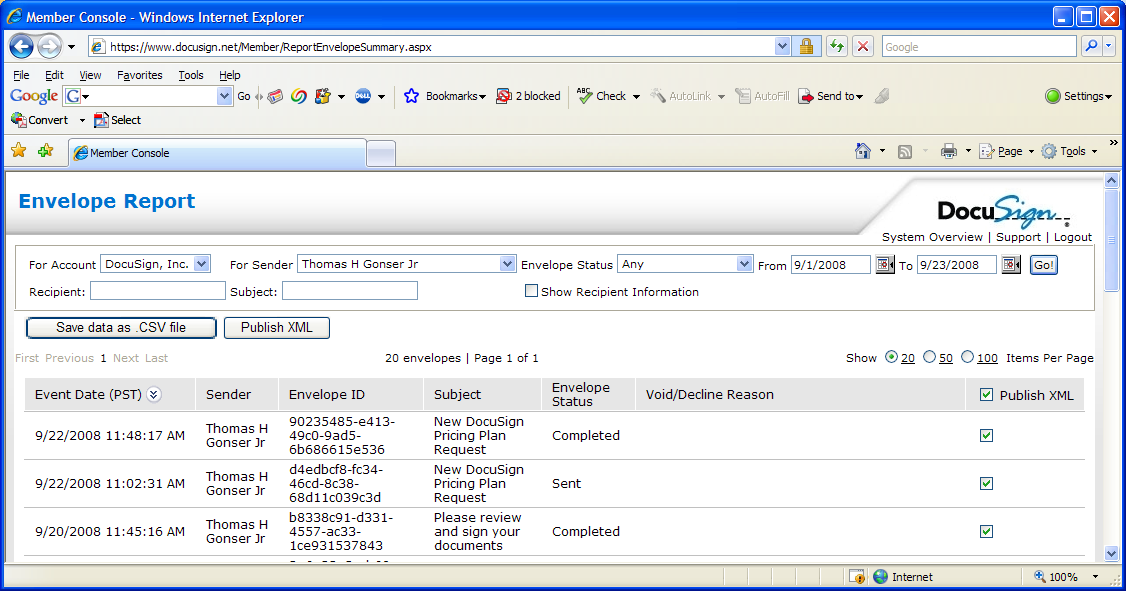
Click SAVE to save your configuration on any of the 3 settings screens.

## Solution Architecture

The Publisher Service acts on behalf of user accounts when transactions reach specified triggers. At that point the DocuSign XML is sent to the set HTTPS URL.



Note that the delivery of the status is not guaranteed and DocuSign system doesn’t retry delivery in the case that the Web Server is unavailable. You may, however, manually publish a transaction from the Envelope Report tool in the DocuSign system to your listener system. This is accomplished by selecting the envelopes in your report, and clicking on the ‘Publish XML’ button.



The flow of events is outlined below.



## XML Information Structure

Publisher sends the status update in the body of an HTTP post. The receiving web server needs to take the entire structure and parse the XML in order to make use of the various elements available in the XML.

### Key Transaction Elements

Key transaction elements available are listed below. The full XML structure contains more, but these are the most commonly used data elements:

* Status Information
  + Sent Date/Time
  + Envelope Status (in process, completed)
  + Envelope ID
* Envelope Information
  + Document(s)
  + Recipients
  + Tabs
  + Subject
  + Email
  + Custom Fields (3)
* Recipient activity and information
  + Recipient ID (s)
  + Recipient email (s)
  + Recipient username(s)
  + Recipient Note(s)
  + Recipient Type (Signer, CC, CD)
  + Recipient Sent Date/time
  + Recipient Delivered Date/time
  + Recipient Signed Date/time
  + Recipient routing order
  + Recipient status code (created, sent, delivered, signed, declined)
  + Recipient Event (viewed, printed, download copy, reassigned, declined, signed)
* Document Information
  + Document name (s)
  + Document ID(s)
  + Document Password(s)
* Document Content
  + Custom Tab Name
  + Custom Tab Value
  + Custom Tab Label
  + Custom Tab Required/not
  + Custom Tab Type (text, checkbox, radio, list)
  + TabTypeCode (Signature, initial, name, company, title, date)
  + Document PDF Bytes (Base 64)

To make use of this information your application must parse the inbound XML looking for the data associated with each node you are evaluating. Then extract the data and place it into the external application.

### Form Field data

The Publisher Service is able to publish not only the status of the transaction, but also the values contained in any form fields or envelope fields in the transaction. This is useful to help interpret what transaction data can be updated into the external system. DocuSign supports many different field types including checkbox, radio button, form field, and drop down. These all have a common name structure, and the value from the signer can be extracted from the XML structure.

## Technical Details

The XML post from DocuSign contains the EnvelopeStatus object along with DocumentPDF objects if the configuration has the checkbox to include the push of the documents.

The attached WSDL file contains definitions for both structures.



## Running the Service

Once your Publisher is activated, DocuSign will send the DocuSign XML object to the HTTPS URL in the configuration screen for every event selected from every user selected. If your application is not configured to accept these post messages, the DocuSign system will not return an error.

It is important that you do not turn on the Publisher if you do not have a configured listener at the HTTPS address you enter in the setup screen. Once you have the listener setup, you may test the publisher by sending transactions and observing the behavior of your application.

Test code for an HTTPS listener based on Microsoft .NET is available from DocuSign by contacting your Account Manager.

## Best Practices

In order to take advantage of the Publisher Service, a clear understanding of the use of the information needs to be understood. Questions such as:

1. What data do you want to capture?
2. Who will be accessing this information?
3. What decisions or reporting will be generated?
4. Should the document be pushed?

These questions must be thought out and agreed in order to deploy a solution that will meet business needs. Additionally, developing the HTTPS Listener application to have some flexibility such as the DocuSign Salesforce.com service may enable modifications to the data that is collected without requiring coding for minor adjustments. This ‘field mapping’ approach enables future modifications and changes that can be made by analysts.