SavvyDox Publishing Augmenting SharePoint and Office 365 Document Content Management Systems

Executive Summary

This white paper examines the challenges of obtaining timely review feedback and managing fast version updates when publishing and distributing documents managed using SharePoint or Office 365 Document Content Management (DCM) systems. This document describes an integration prototype that augments the SavvyDox system to alleviate these issues while allowing SharePoint document authors to publish and control source updates with no changes to their existing practices and procedures. The integrated prototype distributes updates by detecting source document changes in a SharePoint Library and automatically publishing them to designated SavvyDox recipients. The paper shows how this document “synching” solution can be easily deployed into an existing SharePoint installation, with minimal customization of the SharePoint system.

Overview

This white paper is intended for a technical audience that is already familiar with SharePoint document publishing and review workflow, and knowledgeable of capabilities in the SavvyDox publishing and distribution system. Ideally, the reader will have some operational experience with both these systems.

Obtaining timely feedback and performing fast revision turnaround of documents in today’s business environment is critical to success in an enterprise organization. For the purpose of this paper, the term ‘responsive review’ will be used to denote the fast, accurate cycle between author changes to a document and the parallel review feedback obtained from document recipients.

First, the challenges of responsive review within a DCM system will be contrasted against the lighter weight ad-hoc practices used for controlling source documents using an environment like SavvyDox and Microsoft Office. Next, a solution that integrates the responsive review into existing DCM workflows will be described. Finally, the integration points will be highlighted and potential future enhancements outlined.

Publishing Practices and Options in an Office 365/SharePoint DCM Environment

The examination will proceed using SharePoint for context. The same considerations also apply for Office 365 which is Microsoft's cloud based solution for DCM, which embeds SharePoint within it.

SharePoint provides a controlled, known location for source documentation as well as control over who gets to make changes to the document source content. A document is first checked out of a SharePoint Library, modified and then checked back in. The access control and location of the source document is set and there is no risk of misplacement as there might be when documents are managed in an ad-hoc fashion on different users' desktops. Any document ownership (authorship) is handled via either explicit or implicit policy – explicit via access controls or implicit via agreed upon team norms for access and update interaction.

Challenges with using SharePoint for Review Collaboration

These center around ease of document discoverability, ease of changed content uptake and ease of document collaboration.

Document discoverability

Documents within the SharePoint environment are organized into Libraries, with the structure defined by a central project coordinator who owns the Library. As with any shared organization, it will likely compromise the preferred structure of a task oriented user, in favor of one that is broadly applicable to all users of the Library. Typically, the structure is a top down oriented folder decomposition. A user will discover the likely location for a document within the structure via document title search. The search result often appears in a different logical location than a user with a particular interest might have surmised.

Also, notification of changes to documents in the Library to recipients, if set up at all, is likely done via email, which can result in the update notification being 'lost' in the large number of emails that flood a typical business user's inbox.

Creation and navigation to changed content

Changes in SharePoint documents are typically managed using the Microsoft Word Track Changes feature. First, the source document editor needs to ensure that her setting is enabled before starting any edits targeting subsequent versions of a document. Document recipients control their view of either reading the latest content or consuming changes since the prior version via their control of the Track Changes display setting. The challenge of attempting to identify and understand changed content is substantial when there has been many edits to the document over the course of several versions being published. Understanding is challenging because the complete history of the change tracking is displayed – it's difficult to extract only the last set of changes that were relevant compared to the document version that the recipient last opened. Change tracking is further compounded for users with mobile tablets that do not have the source document editing software (Word Office or equivalent) resident on their device.

Document Collaboration

One side effect of the check-in, check-out safeguard is that it constrains document reviews to occur serially. After one users checks out the document, anyone else attempting to perform modifications must wait for the original user to check it back in, or access a read-only copy. Having all reviewers work against the published version in parallel without overriding each other's work would save time.

Document collaboration can take place via different mechanisms. In one scenario, recipients suggest changes by directly changing the content (hopefully, with Track Changes enabled). The document is then checked back in to the Library from which they retrieved it and other users now have another set of Track Changes style mark up to read and understand. Another possibility is to make use of Microsoft Word's comment feature – this avoids having document recipients providing feedback from changing the document content directly. This situation works better when there is a designated author/editor and all other users of that document are considered as reviewers that should not change the document content.

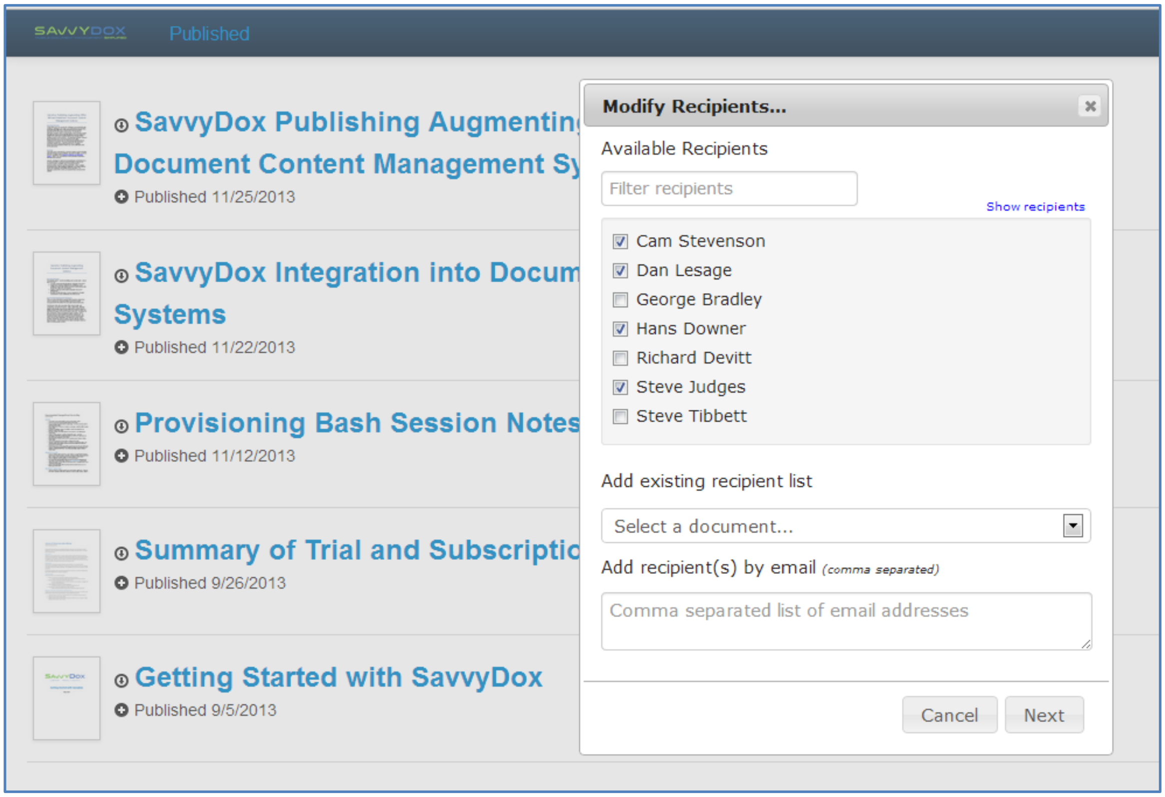
SharePoint provides a tightly controlled system that is optimized for internal enterprise use with heavily structured access control. Access to documents is typically restricted to users within an organization and more likely only to users within a certain defined group or department. The access control structure often subsets user and group definitions in Active Directory, Exchange or a corporate LDAP system. Sharing and collaboration with users outside the organization is restricted within the environment and is typically handled out of band (via email attachments as an example) which can result in external participant losing track of versions or timely updates to the document. Forwarding via email can also potentially place intellectual property at risk if it is not adequately controlled or monitored.

Publishing Practices and Options in an Ad-Hoc Document Management Environment

In contrast, a system like SavvyDox does not dictate how source documents are created prior to publishing, nor where they reside. This provides great flexibility and freedom, but it also means that ownership and management of source documents can be problematic for multiple authors or if the documents are not well organized and backed up from the author's desktop. Documents make their way into SavvyDox via one of the following ways:

Web Portal

Convert the document to PDF and then upload to the Web portal. Documents of the same name being published by the same author can be interpreted as either a new version or as a new document. The source of the document in this case is whatever produced the PDF being uploaded. The Web portal is also used to specify the recipient list and any author supplied information such as description of the document or changes associated with the version.



SavvyDox Publishing Augmentin
Document Content Management Sy
Published 11/25/2013

SavvyDox Integration into Docum
Systems
Published 11/22/2013

Provisioning Bash Session Notes
Published 11/12/2013

Summary of Trial and Subscriptic
Published 9/26/2013

Getting Started with SavvyDox
Published 9/5/2013

Modify Recipients...

Available Recipients

Filter recipients

Show recipients

☑ Cam Stevenson
☑ Dan Lesage
□ George Bradley
☑ Hans Downer
□ Richard Devitt
☑ Steve Judges
□ Steve Tibbett

Add existing recipient list

Select a document...

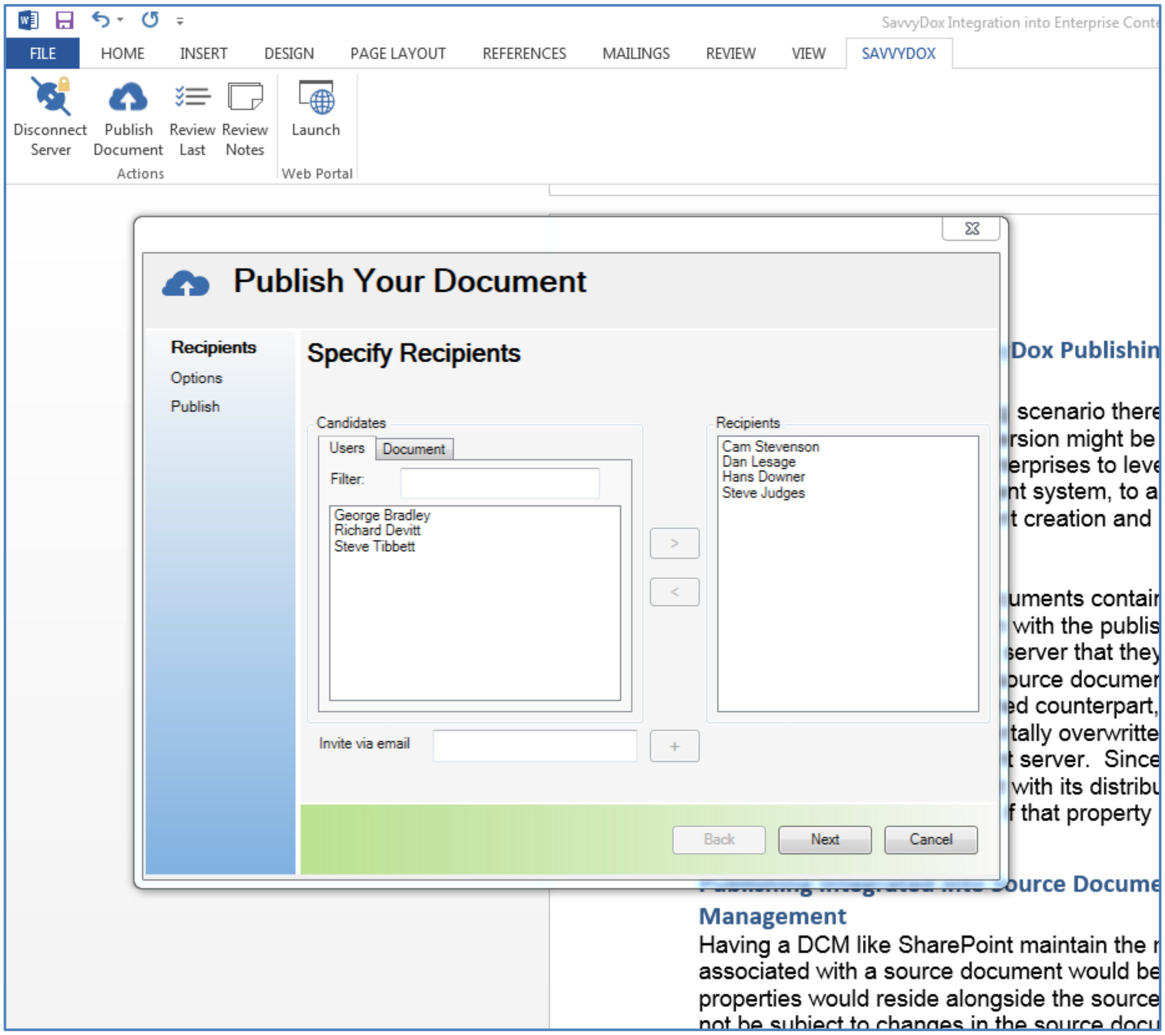
Add recipient(s) by email *(comma separated)*

Comma separated list of email addresses

Figure 1 Modifying metadata (recipients) for an uploaded PDF document in the Web Portal

Microsoft Office Integration

A client extension to Microsoft Office (known as an Add-In) augments Word and PowerPoint applications with extra controls to allow publishing directly from within these applications. How and where the Office documents are organized is totally up to the document author.



Publish Your Document

Specify Recipients

Candidates

Filter:

George Bradley
Richard Devitt
Steve Tibbett

Recipients

Cam Stevenson
Dan Lesage
Hans Downer
Steve Judges

Invite via email

Dox Publishin

scenario there
rsion might be
erprises to leve
nt system, to a
t creation and

uments contair
with the publis
server that they
ource documer
ed counterpart,
tally overwritte
t server. Since
with its distribu
f that property

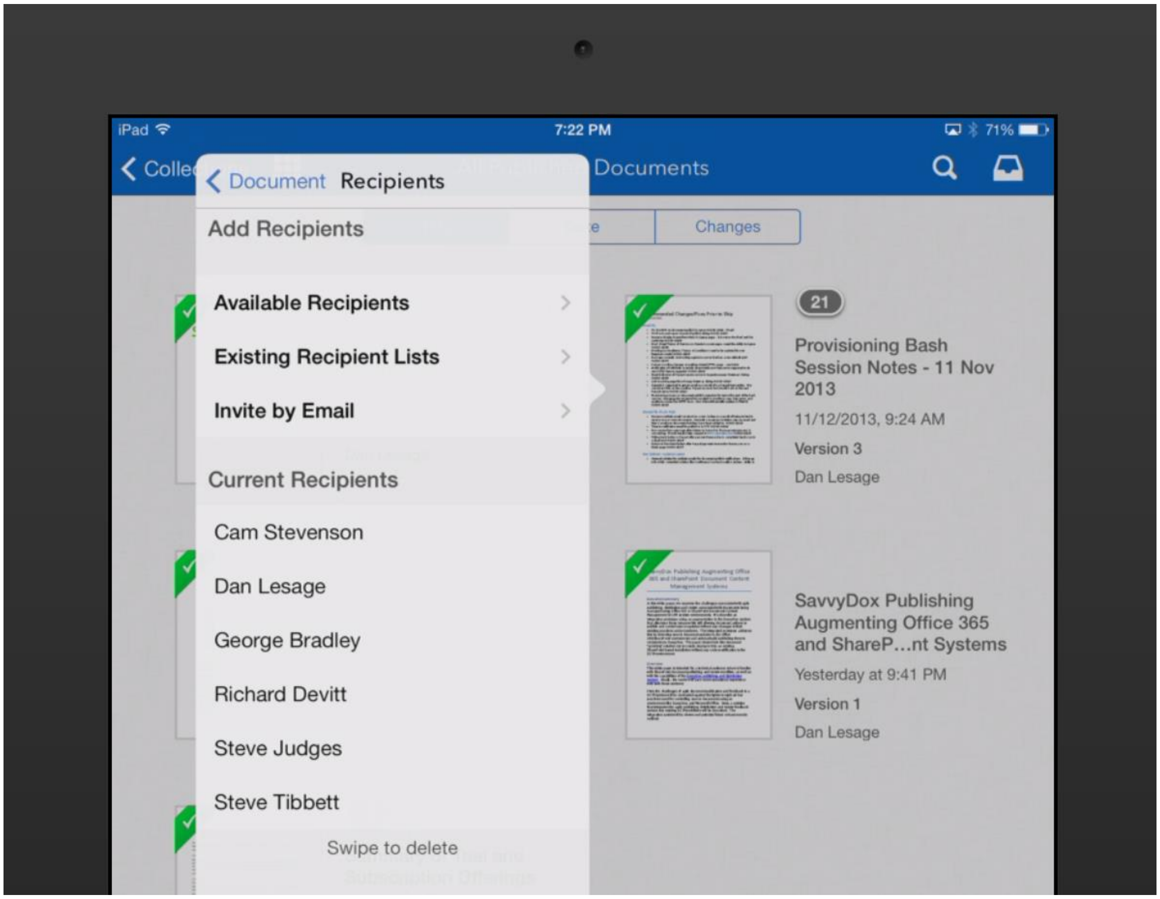
ource Docume
Management

Having a DCM like SharePoint maintain the r
associated with a source document would be
properties would reside alongside the source
not be subject to changes in the source docu

Figure 2 Metadata (recipients) being added via Office Add-In wizard

iPad Application and Reader

Many iPad document creation apps provide a hook to other applications for export via the “Share” button, which will bring up a list of applications that can consume the sharing app’s output. For iPad apps that can produce PDF output, the SavvyDox iPad Reader icon will appear in the applicable list. Selecting the Reader will cause a publish operation to the server. Selection of recipients and other metadata is done as a separate step on the iPad. The implied location of the source document is presumably derived from iCloud or another iPad friendly file storage provider.



Cam Stevenson
Dan Lesage
George Bradley
Richard Devitt
Steve Judges
Steve Tibbett

Provisioning Bash Session Notes - 11 Nov 2013
11/12/2013, 9:24 AM
Version 3
Dan Lesage

SavvyDox Publishing Augmenting Office 365 and ShareP...nt Systems
Yesterday at 9:41 PM
Version 1
Dan Lesage

Figure 3 Modifying metadata (recipients) using iPad Reader Application

Challenges with Using SavvyDox Publishing in DCM Environments

In all these ‘ad-hoc’ publishing scenario there is always the risk that the latest source document version might be deleted, misplaced or that an older version is mistaken as the current version. It therefore makes sense for enterprises to leverage their existing document content management system, to allow similar controls to be applied to source document creation and update, as for document publication to recipients.

Ideally, links between a source document and its distributed counterpart are maintained somewhere. The link is used to identify the source document to the publishing system, so that document metadata (e.g.: list of recipients, document description) can be retrieved and used in subsequent publishing operations. For ad-hoc document publishing, those links are stored within the source document itself. Since source documents are potentially subject to mentioned misplacement issues, it would be better if the link relationship maintained between the source document and the published metadata was also maintained in a central location.

Integrating Publishing into Source Document Content Management

Having a DCM like SharePoint maintain the metadata properties associated with a source document would be an improved solution – the properties would reside alongside the source document, but would not be subject to changes in the source document file format itself as Word or PowerPoint file properties might be. The source document resides in a location within SharePoint as designed by the project director, while the recipients are allowed to organize their on-device equivalents in a way that makes sense to them using a SavvyDox document collection.

Another benefit of publishing based on DCM source is that the (source of truth) author is always the DCM system itself – temporary authorship is easily handled by users via check-out check-in capabilities that they are already familiar with.

To build a 'connector' that would operate between the DCM document source environment and the publishing and distribution environment is feasible as an add on to SavvyDox with a very lightweight integration into SharePoint. To better understand how to accomplish this requires a bit of background on the SavvyDox system architecture.

Architectural Support for Integration

The SavvyDox architecture has been built specifically to provide a flexible way to expand the system to operate within a wide variety of environments. The key element of the architecture is the client/server support that allows for operation within public cloud, private cloud and on-premise environments.

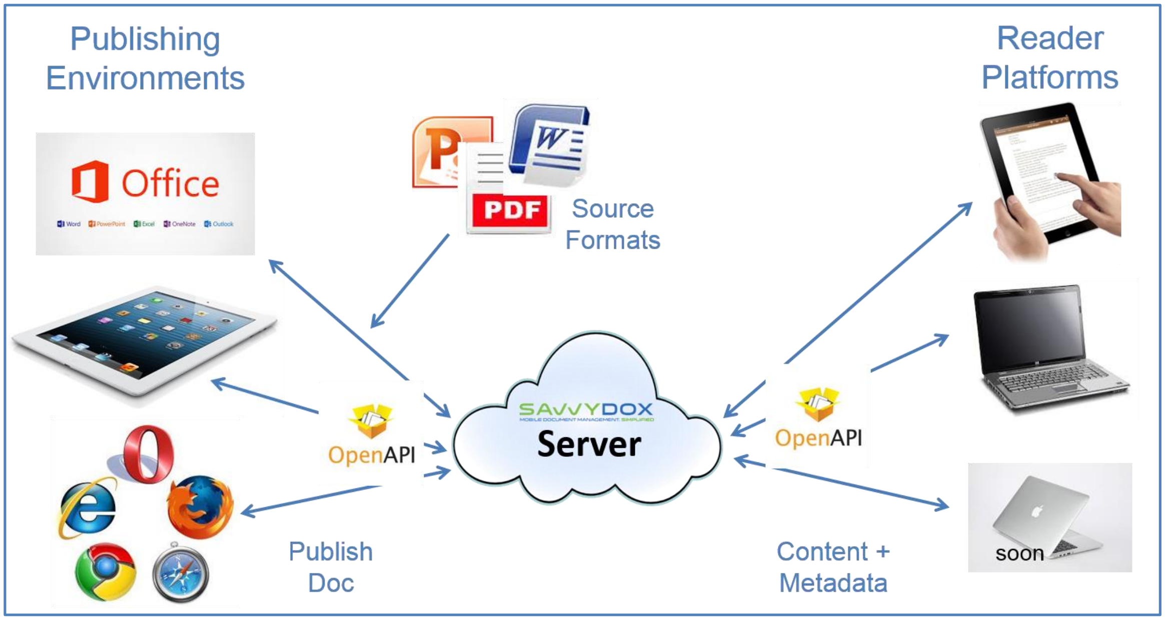


Figure 4 SavvyDox Client/Server Architecture

Communication with a SavvyDox server is all done from client platforms via a Web based Application Programming Interface (API). The API is stateless and employs the practices of [REpresentational State Transfer (REST)] – this means that clients carry and manage state information between requests – the server handling of a client session is stateless and only responds to queries or actively broadcasts event information via push and other notification mechanisms. Data posted or retrieved from the server is stored in a server-associated database. The format for posting and retrieve data in the Web API is JavaScript Object Notation (JSON).

The API separates out the publishing “meta data” associated with a document from the document content itself. This allows the document content to remain under control of the author, as SavvyDox will not augment the source document contents with additional data. Metadata associated with publishing operations, recipient lists, change differences, recipient reading state, devices, users, notes and events are all stored independently in the database, yet associated with the document where applicable. This also allows the system to easily derive analytic data from a wide range of query sources – not just those contained to an individual document’s data. A typical request from client to server to retrieve the documents that a user has access to would look like:

```
GET https://cloud.savvydox.com/2/documents?since= 2011-12-18T10:55:26-0500
```

and the (abbreviated) resulting retrieved JSON data:

```
"documents": [
    {
      "author": "0959252A-A1EC-4676-B1E3-216984BEC9A5",
      "collections": ["DD50881F-AF79-41CE-B386-78DCF65DDE53"],
      "id": "33089376-FA34-4B47-81FA-630AEFFB3521",
      "lastClosed": "2011-12-18T12:37:02-0500",
      "lastOpened": "2011-12-18T10:55:26-0500",
      "lastPageClosed": 3, "lastPageOpened": 3,
      "lastVersion": 2,
      "message": "Could you please confirm this meets your expectations?",
      "noteDefaultVisibility":"Personal", "noteVisibilityLimit":"Personal",
      "notes": 0,
      "pages": 1,
      "tags":["Pending"],
      "title": "Board Meeting Agenda",
      "uploadDate": "2011-12-17T09:15:35-0500",
      "version": 2
    },...
```

Authors publish using one of the 3 supported publishing client environments described above. Recipients read and provide feedback via one or more document reader client environments installed on one or more of their devices.

Synchronized Publishing from DCM to SavvyDox

SavvyDox's API allows for automated publishing based on an external web call. As long as the same information that an author would enter (e.g.: recipient list, version comments) is available, SavvyDox does not care where the REST API call originates from.

One way to accomplish an automated publish to SavvyDox as the result of an SharePoint check in is to have a small service application check for updates in SharePoint, determine if there are new published document updates and then publish those updates to SavvyDox based on preconfigured publishing information (e.g.: recipients, etc.).

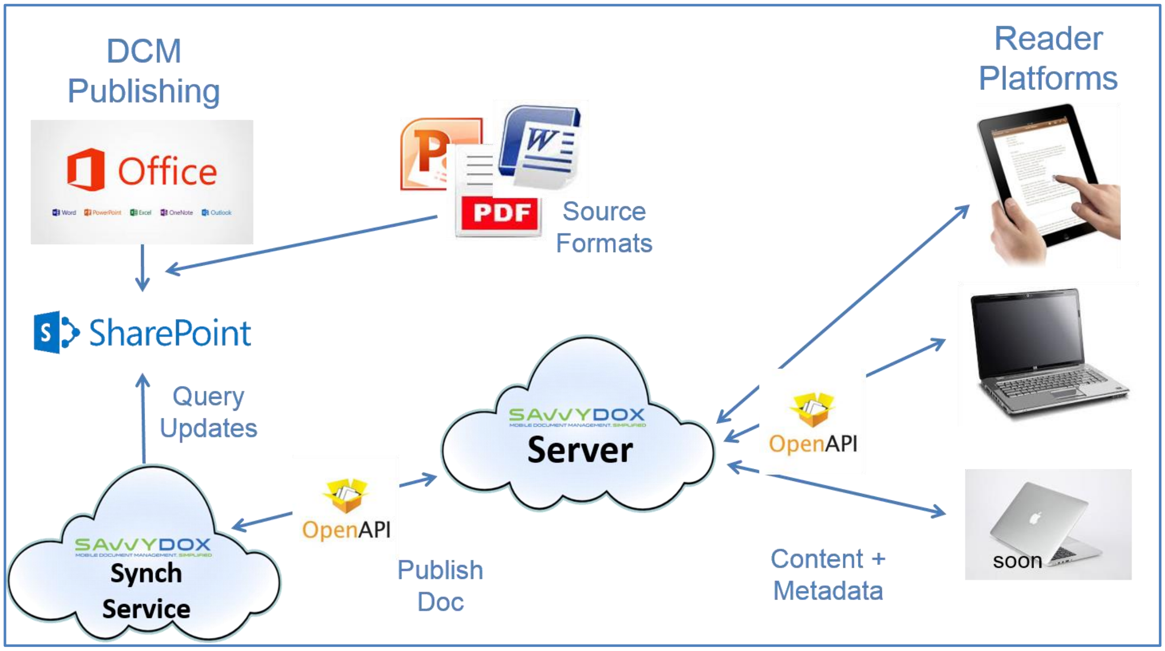


Figure 5 SavvyDox architecture incorporating SharePoint DCM and Synch Service

The information required for publishing the document is split into several sets:

1. Information about or for the SavvyDox cloud or on premises server and its environment
2. Information about or for the SharePoint (or Office 365) DCM environment

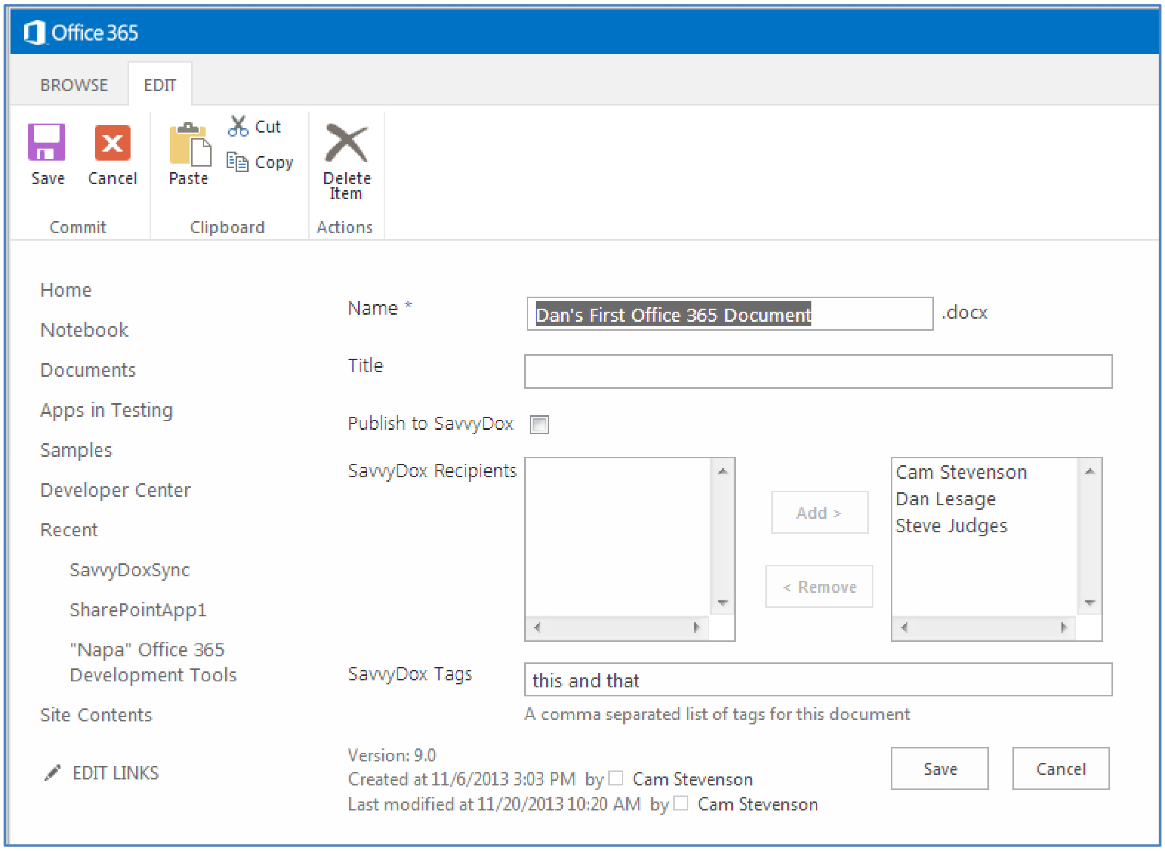
3. Metadata specific to the document being published

A configuration file residing in conjunction with this SavvyDox to SharePoint “Synch Service” is sufficient to demonstrate the capability for the first two kinds of configuration entries above. This information can be moved to a local, secure database on the system running the Synch Service at a later time. The first two sets of configuration entries connect both ends of the ‘pipe’ from which a source document is retrieved in SharePoint and to which the published document is pushed in SavvyDox. This means that the Synch Service application can reside on any system that can access both SharePoint and the SavvyDox server, with the tradeoffs being latency and affinity to SavvyDox or the DCM store.

```
connectorid=test-connector
connector=sharepoint2013
sduser=cam.stevenson
sdpassword=pwd
sdserverurl=http://test.savvydox.com:8088
sdrecipients=cam.stevenson
sdcomment=Published from sync service
cmsserverurl=savvydoxdev.sharepoint.com
cmsfolder=/Shared Documents
filetypes=pdf,doc,docx
polldelay=10
mapdeletions=true
openofficeapppath=/Applications/LibreOffice.app/Contents/MacOS/soffice
clientid=0dfef2bc-114a-49b3-9479-0c02725a3e28
clientsecret=r822CjhfmBucuPrRkONd40G81Iblg2kV0dbAlsOKHw4=
```

Figure 6 Sample Synch Service configuration entries

The 3rd set of configuration information required for publishing is document specific and therefore needs to be associated with the source document in SharePoint. This is accomplished via extension to SharePoint System Settings by defining custom document properties for common office source file types like .docx (Word). Document properties are an existing capability within SharePoint and are the natural way for SharePoint users to input site specific data like “Document Title” (separate from source file name). To allow for SavvyDox publishing, SharePoint is augmented to include additional document properties that will hold the document specific attributes required by the publishing server.



Name * Dan's First Office 365 Document .docx

Title

Publish to SavvyDox □

SavvyDox Recipients

Cam Stevenson
Dan Lesage
Steve Judges

SavvyDox Tags this and that

A comma separated list of tags for this document

Version: 9.0
Created at 11/6/2013 3:03 PM by □ Cam Stevenson
Last modified at 11/20/2013 10:20 AM by □ Cam Stevenson

Figure 7 SharePoint Document Properties Including SavvyDox Metadata

The SavvyDox Synch server will query this information when it detects that a new document or document version has been added to a configured SharePoint Library, and pass it along with the PDF format of the document to the SavvyDox server for publication.

The Synch Service is modular, with the connector to the DCM system being configurable to work with different DCMs or file synchronization systems.

Augmenting SharePoint for use With SavvyDox

Customizing these environments for use with the SavvyDox Synch Service requires a one time installation of a SharePoint .wsp (Windows SharePoint) solution file, which contains the definitions of SavvyDox publishing metadata properties that would be stored with a document in a Library. The .wsp file import is the automated equivalent of defining and updating SharePoint columns and data types in the System Settings. The solution does not contain any runtime code – all runtime processing is handled via the Synch Service which can reside on the same or a different server than the SharePoint server. This also means that testing of updated SavvyDox Synch Service capabilities can be done independently of the back-office SharePoint installation and vice versa, allowing for more flexible back-office IT validation and certification prior to overall system deployment.

PDF Conversion in SharePoint and Office 365

Since SavvyDox is agnostic as far as source document format is concerned, somewhere along the way, a conversion to PDF needs to take place so that the non-modifiable document content can be published to recipients via their client reader applications. Although Office to PDF conversion is an optional configurable feature within SharePoint, it is not available with Office 365. To allow operation in both environments, the SavvyDox Synch Service prototype contains a third party Office to PDF converter.

For installed SharePoint systems, the Synch Service could also leverage the SharePoint Office automation API that will convert Office source documents to PDF, rather than using the built in 3rd party converter. Mechanism details vary depending on the specific SharePoint version, but ideally an event trigger upon updating a source document in a Library would automatically cause a conversion to its PDF equivalent in the same or a different Library. The SavvyDox Synch Service could then look for changes in the PDF version of the document rather than the source version, avoiding having to perform the conversion using its embedded 3rd party PDF converter. For Office/SharePoint users that require Office as their converter, this solution may be preferable.

Areas of Further Investigation

In addition to examining support for additional DCMs, one area of continued research is deeper integration into corporate user management systems. For Microsoft environments, this would mean Active Directory or Outlook (Exchange). There are many possibilities here, including supporting single sign-on or allowing publishing to any user in a domain. Mirroring group publication between corporate address systems and SavvyDox is another area of investigation.

Summary

This paper has examined some of the key challenges face by document publishers and recipients in both ad-hoc and DCM based document management environments. It has also explored how the SavvyDox document publishing and distribution system can be integrated into SharePoint and Office 365. A new component of SavvyDox – the SavvyDox Synch Service has been described, including how it would interact with the SavvyDox server and SharePoint to automatically publish documents based on source document updates in a SharePoint Library. The different configuration options were shown and the integration points between the systems identified to provide a solution that emphasizes the strengths of both environments while retaining the flexibility of existing publication and review practices.