DocMan - how does it work?
Generally DocMan works by synchronising from CRM to SharePoint, and this applies for Changes to Records in CRM, Content Types and Security.

This is great when CRM is the Master, as this controls SharePoint for Document Management – however is less effective for any changes to SharePoint, as these are not then synchronised from SharePoint to CRM. This is based on the CRM Event Model being much easier to work with rather than Events in SharePoint, and our primary expertise in being Dynamics Experts first and then SharePoint experts second.

Can I Rename a Document Library or Site?
In CRM, yes. DocMan will pick up on the change of name in CRM and rename the corresponding area in SharePoint as a result.

However changes directly in SharePoint are not mapped back across to CRM.

Can I remove a Document Library or Site?
This can be done by deleting the record in CRM, this will then delete the corresponding area in SharePoint.

This delete in CRM and automatically delete in SharePoint is build to accommodate Right-to-be-Forgotten Requests and so remove both CRM and Document Personal Data on demand.

This setting can be deactivated if there is any fear of an accidental deletion in CRM passing through to SharePoint.

Can the site be amended in any way to reflect the way it should have been setup?
Changes to the site can be made in SharePoint after it has been created – DocMan will re-synchronise Security but does not currently have any concept of 'tombstoning' to remove any areas in SharePoint based from the rules in CRM.

It does however have the concept to re-create areas in SharePoint based on the rules from CRM to fix errors in SharePoint and ensure consistently.

Can DocMan create new Sites using a specific Template?
Yes, the logic can be configured to create a new Site per Record (of a certain entity) in CRM.

This Site Creation is then configurable to use a Site Template – either a standard template in SharePoint, or a custom template.

If a Template for Site Creation is updated in CRM, is it possible to apply these updates to existing SharePoint sites that are using that Template?
Unfortunately Sites based from a SharePoint Template cannot be re-applied from an updated Template – this is because the Template defines the Content in a new SharePoint Site, and so an updated Template may need to remove Content as much as add new Content into Sites using an updated Template.

If not, how could/would the changes be rolled out to the existing sites?
For new Document Libraries or Sub-Sites, DocMan will create these automatically based on new records added to CRM and the Config Rules we put in place.

For Content Types, these can be published into SharePoint Site and Sub-Site at a time – the way this works is that DocMan loops through an Advanced Find Query in CRM, reviews the record in CRM, and then publishes the following into SharePoint:

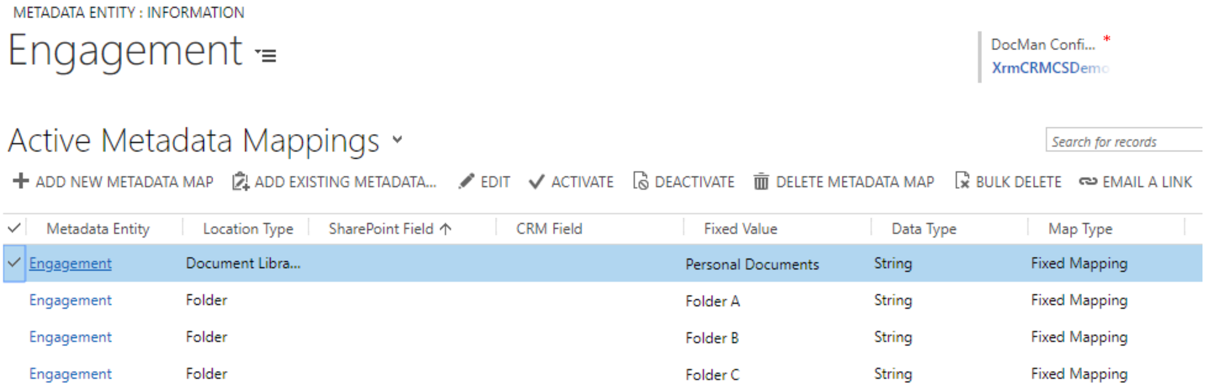
- **Location -** Creates the Location in SharePoint if this does not exist already.
- **Content –** Add any fixed areas to the Location in SharePoint. (typically fixed libraries or folders)
    - Adds new fixed Document Libraries or Folders that have been configured in the DocMan Rules Configuration.
        - So taking the scenario where DocMan is configured to create a new Document Library per Opportunity, with 3 fixed folders for ‘Proposal’, ‘PID’ and ‘Project Plans’
        - If the Document Library already exists for the Opportunity, DocMan will continue to use this existing Library
        - But if the 3 Fixed Folders do not exist, DocMan will automatically create each of the 3 Folders into the Library so this is setup in the way we expect.
    - In this way, the desired ‘template’ for a Site or Document Library can be defined by the DocMan Config Rules and then applied to all new or existing SharePoint Locations.
    - This then creates the Folders and Libraries required for existing Sites, and so provides a ‘re-publish’ option to distribute content to previously created sites.
- **Content Types –** Associates the existing Location with the correct Content Types as per the Config Rules.
- **Security -** Publishes the Security from CRM into the existing Location so the initial Security is set correctly.

As an example..

If we set up our Engagement Entity in CRM to have:

- New Site per Engagement
- Default Document Library for ‘Engagement Documents’
    - 3 x Folders in this Document Library as the initial template
- Additional Document Library for ‘Personal Documents’

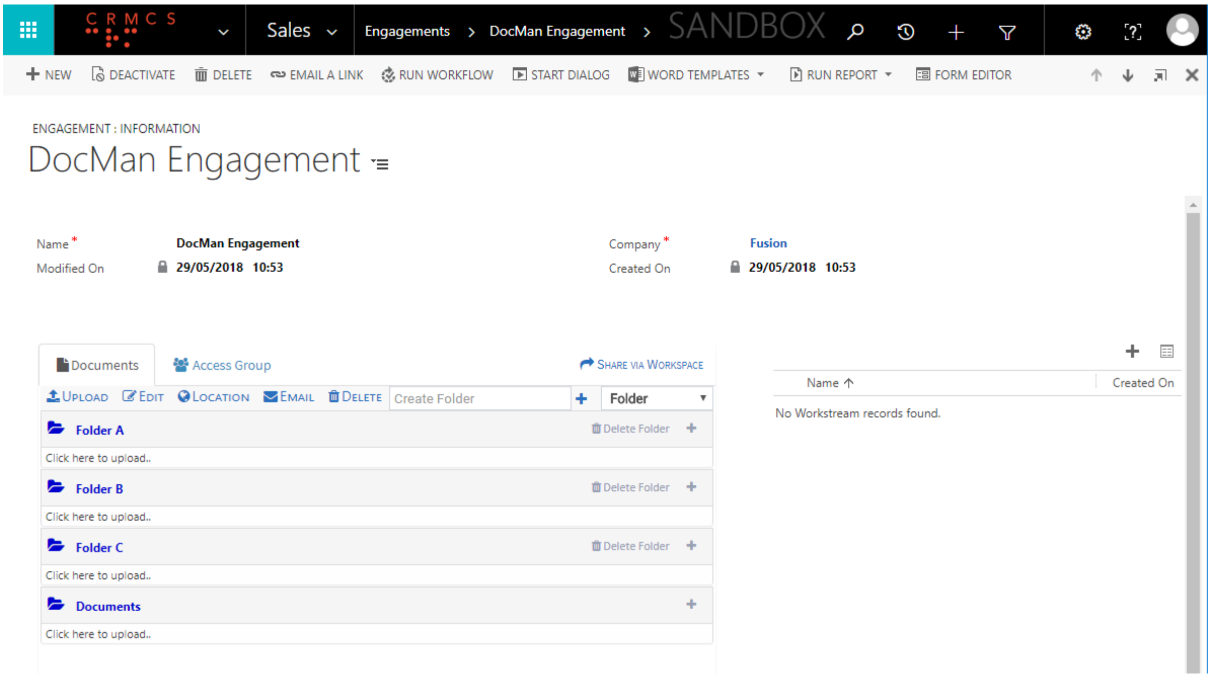
This would be configured in the DocMan Settings as:



Metadata Entity	Location Type	SharePoint Field ↑	CRM Field	Fixed Value	Data Type	Map Type
Engagement	Document Libra...			Personal Documents	String	Fixed Mapping
Engagement	Folder			Folder A	String	Fixed Mapping
Engagement	Folder			Folder B	String	Fixed Mapping
Engagement	Folder			Folder C	String	Fixed Mapping

Figure 1 - Defining the List of Document Libraries and Folders that should be created automatically as the Template when a new Site is created for an Engagement Record in CRM

This would be created for each new Engagement as:



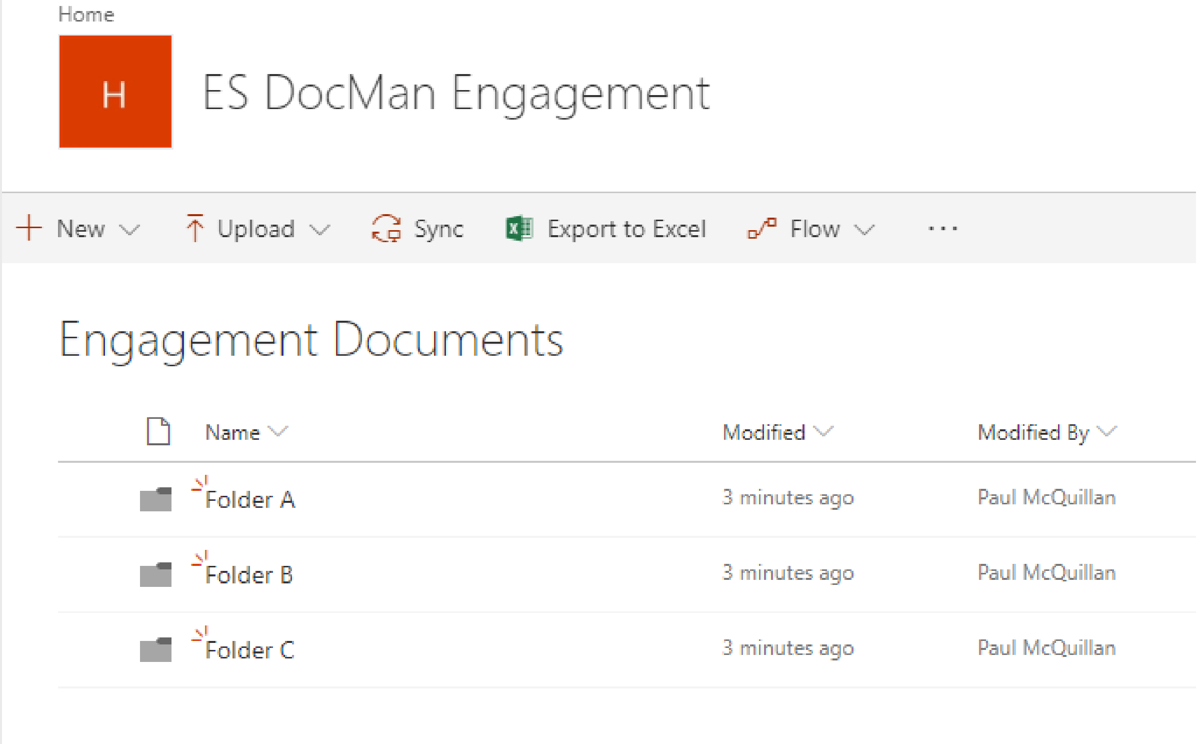
Name * DocMan Engagement
Modified On 29/05/2018 10:53
Company * Fusion
Created On 29/05/2018 10:53

Folder A
Click here to upload..
Folder B
Click here to upload..
Folder C
Click here to upload..
Documents
Click here to upload..

No Workstream records found.

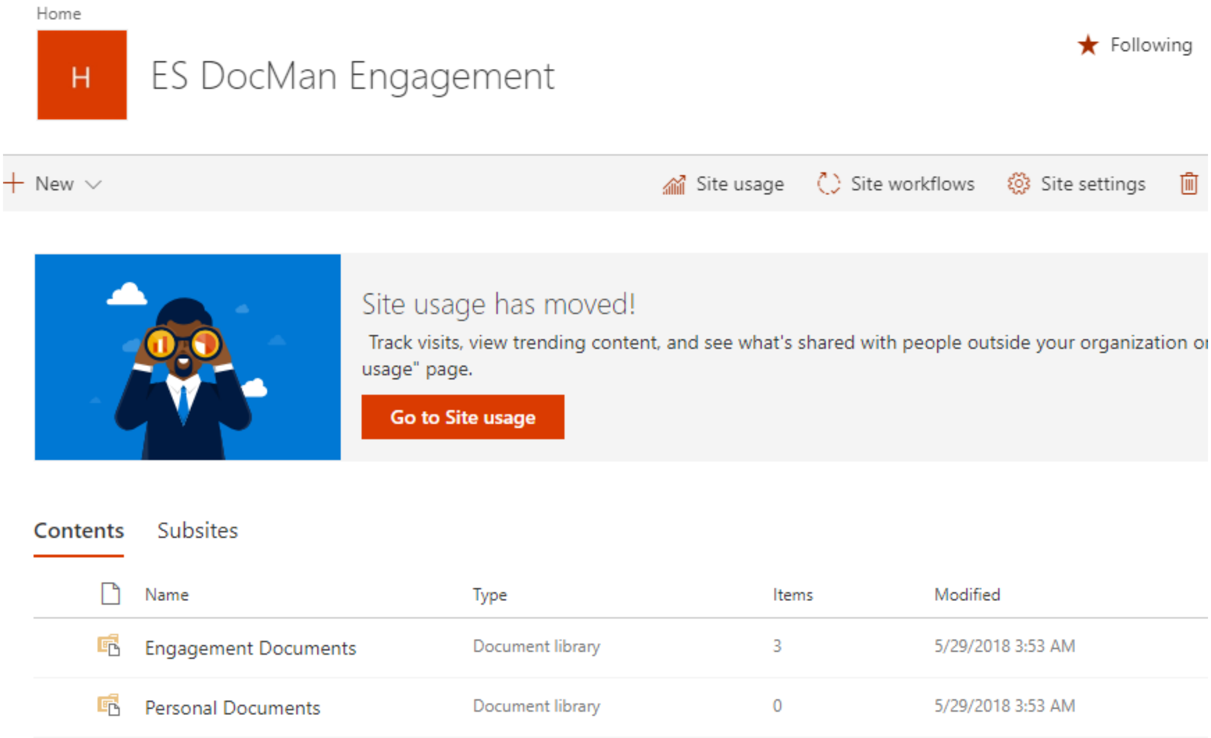
Figure 2 - The resulting DocMan format for a new Engagement Record in CRM

Clicking Location would then view this structure within SharePoint itself:



Engagement Documents

Name	Modified	Modified By
Folder A	3 minutes ago	Paul McQuillan
Folder B	3 minutes ago	Paul McQuillan
Folder C	3 minutes ago	Paul McQuillan



Site usage has moved!

Track visits, view trending content, and see what's shared with people outside your organization o
usage" page.

Name	Type	Items	Modified
Engagement Documents	Document library	3	5/29/2018 3:53 AM
Personal Documents	Document library	0	5/29/2018 3:53 AM

Is it possible to Check In/Out Documents, etc. using DocMan in CRM?
No, DocMan does not currently allow Check In and Check Out of Documents – instead favouring a single click to open the Document in SharePoint and start making edits. (dependent on Internet Explorer as this integrates directly into SharePoint, this is possible in Chrome using a Plugin)

However this could be built into the DocMan Panel / Inline Editor relatively easily as an extension to the App if required.

Does DocMan work with volume?
Yes, SharePoint is scalable platform that can expand up to as many Sites, Libraries and Folders as we need and DocMan works the way SharePoint does in this regard.

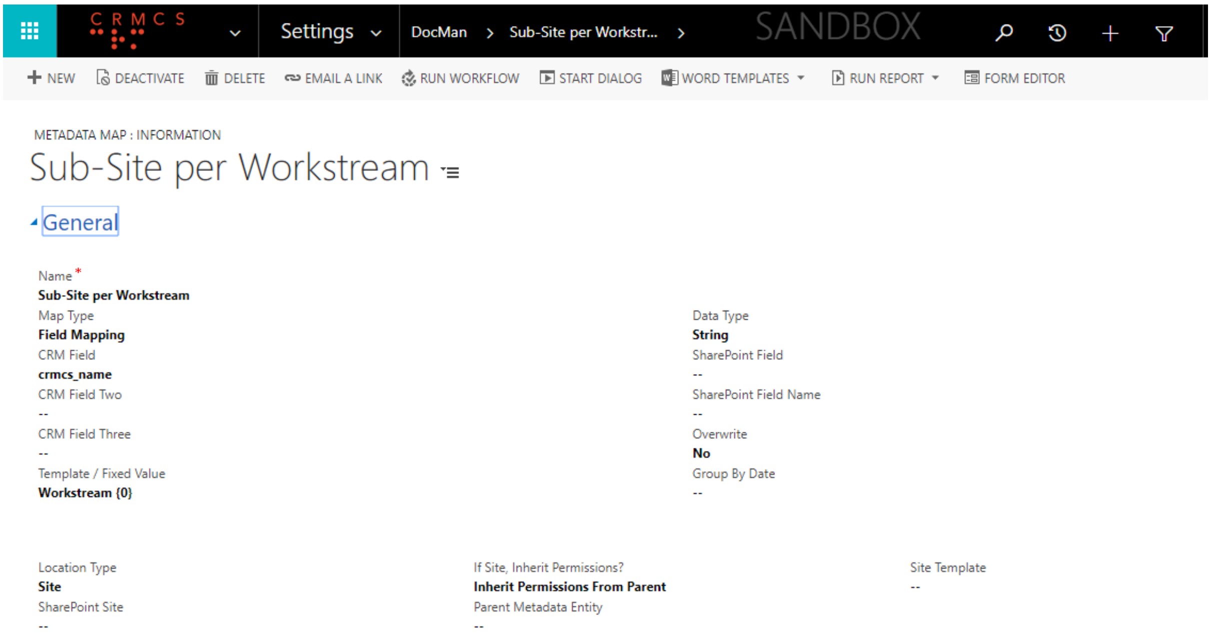
As an example, our tentpole DocMan Client uses a 0.6 Terabyte Documents Database with 26,300 Document Libraries across 200+ SharePoint Sites.

Amount of Documents per Record
SharePoint can store a large volume of Documents within a single Document Library, and DocMan will divide these by the CRM Record involved. (so viewing Documents for a Contact will only show documents for that Contact, whereas viewing Documents for the Account will show all documents in the hierarchy or just for that Account)

This can be a huge number of documents for a particular location, and so we would typically use Metadata or Folders to help keep a large number of Documents for a single CRM Record manageable.

How does DocMan control what Security in CRM maps to SharePoint?
Each record in CRM is linked to a SharePoint Site and/or Document Library.

This Site can inherit permissions from the Parent Site, and so Parent Record in CRM, or have unique permissions. This division between record types that inherit permissions vs unique permissions is defined in the configuration rules for DocMan.



METADATA MAP : INFORMATION

Sub-Site per Workstream

General

Name *
Sub-Site per Workstream

Map Type
Field Mapping

CRM Field
crmcs_name

CRM Field Two
--

CRM Field Three
--

Template / Fixed Value
Workstream {0}

Data Type
String

SharePoint Field
--

SharePoint Field Name
--

Overwrite
No

Group By Date
--

Location Type
Site

SharePoint Site
--

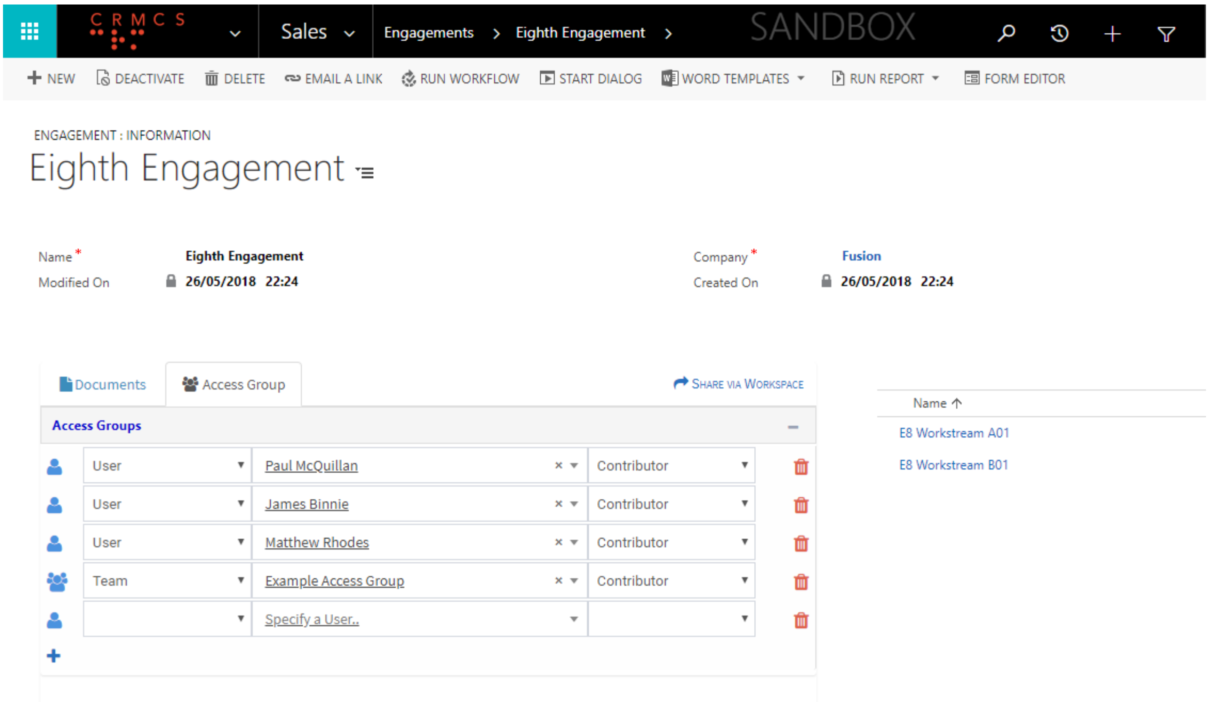
If Site, Inherit Permissions?
Inherit Permissions From Parent

Parent Metadata Entity
--

Site Template
--

Figure 3 - Defining that each Workstream Record in CRM will create a Sub-Site in SharePoint, and the permissions for this Site will inherit from the Engagement Parent Site

Each Site and Library with Unique Permissions has then resulted from its corresponding record in CRM – in CRM we can define the individual Users or Access Groups that have access.



ENGAGEMENT : INFORMATION

Eighth Engagement

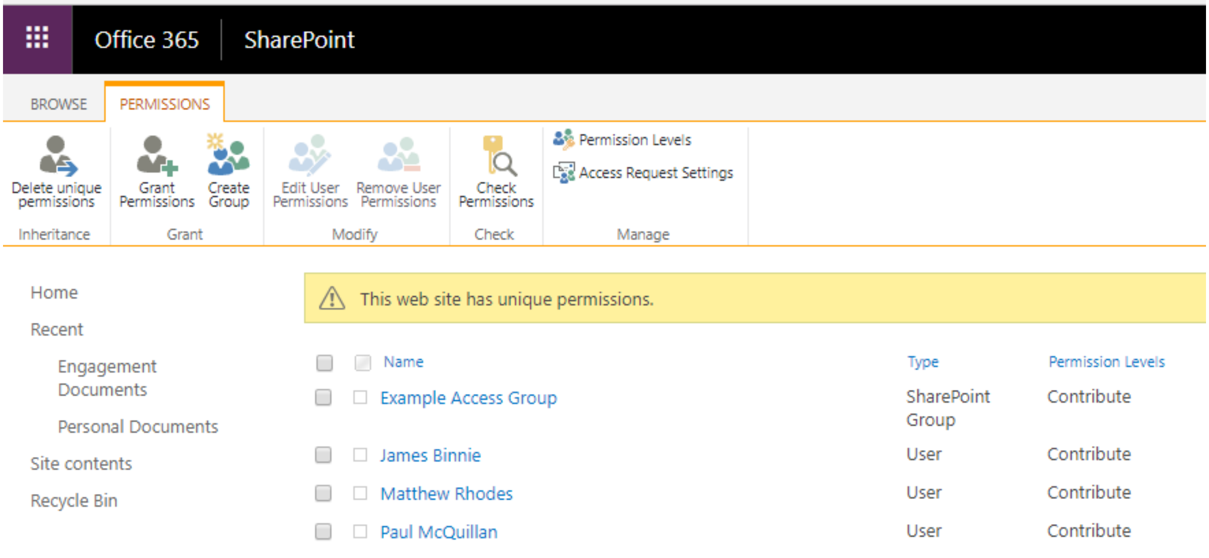
Name *	Eighth Engagement	Company *	Fusion
Modified On	26/05/2018 22:24	Created On	26/05/2018 22:24

Access Groups

User	Paul McQuillan	Contributor
User	James Binnie	Contributor
User	Matthew Rhodes	Contributor
Team	Example Access Group	Contributor
	Specify a User..	

Name ↑
E8 Workstream A01
E8 Workstream B01

Figure 4 - Defining the Access Group for an Engagement Record in CRM; this then maps to the corresponding Site for the Engagement in SharePoint



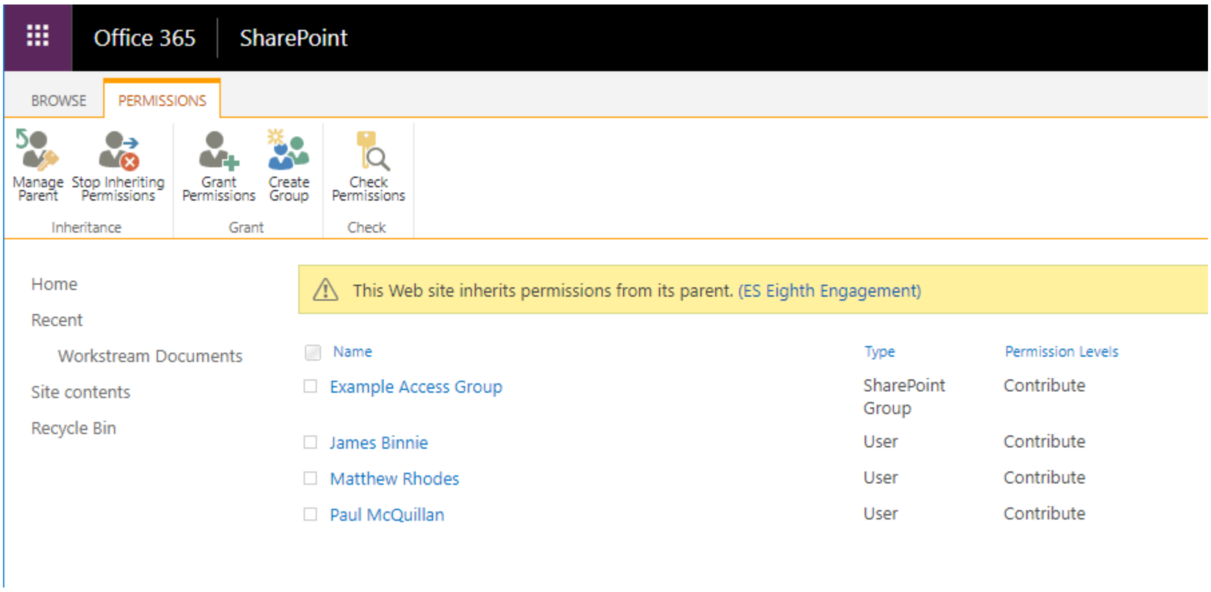
This web site has unique permissions.

Name	Type	Permission Levels
Example Access Group	SharePoint Group	Contribute
James Binnie	User	Contribute
Matthew Rhodes	User	Contribute
Paul McQuillan	User	Contribute

Figure 5 - The Permissions in SharePoint for the corresponding Engagement

This list of permissions is then defined uniquely for the Engagement, so each Engagement can have a different set of access levels easily set in CRM.

Any Workstreams for that Engagement then inherit these permissions from the Engagement and so have a fixed set of permissions controllable from the Engagement in CRM.



This Web site inherits permissions from its parent. (ES Eighth Engagement)

□ Name	Type	Permission Levels
□ Example Access Group	SharePoint Group	Contribute
□ James Binnie	User	Contribute
□ Matthew Rhodes	User	Contribute
□ Paul McQuillan	User	Contribute

Figure 6 - The Inherited Permissions for a Workspace linked to the Parent Engagement

Is it possible to map DocMan to existing sites?
Yes.

If present, DocMan this will use the existing Document Location records in CRM – and so use the SharePoint Location that each record in CRM is currently associated with.

If for any reason the Document Locations are not present in CRM DB Table – then it will use the existing SharePoint Sites based on the Naming Convention for how the Record in CRM maps to SharePoint. (creating the SharePoint Site, Library or Folder if not present, using the existing SharePoint Site, Library or Folder if already present)

If neither the Document Location or Naming Convention is in place, then it is possible to configure different SharePoint Sites in the configuration matrix within DocMan. (i.e. if Account A meets Condition B then use Site C, whereas if not then use Site D)

However this last option is typically used to configure a small number of Top-Level Sites and not a Site per Record.

So a typical usage would be:

- Site for Investment Clients
- Site for Fund Clients
- Site for Legal Clients

Where the decision for the Account Sub-Site or Document Library would be based on a Condition in CRM defining which Top-Level Site to use to create for that type of Account.

So all Investment Client Sub-Sites and Document Libraries would use the Investment Site, and all Fund Clients would similarly be created from the Top-Level Fund Site, and so on.

This is typically where the existing set of Top-Level Sites has already been created and we want our new CRM to SharePoint Areas to 'snap-in' to this existing structure.

Can we customise the naming convention used for Sites? (i.e. if we wanted to use reference numbers rather than Account Names for the URL?)
Yes – we configure DocMan with Metadata Maps that specify how a Record in CRM will be reflected in SharePoint.

So each Site, Document Library or Folder created in SharePoint for a Record in CRM can be named from any field in CRM or a combination of fields in CRM.

For example we could define our structure as:

Field 1: fullname
Field 2: parentcustomerid
Field 3: new_contactnumber

Then define our map as:

{2} {1} - Contact - {0}

This would then create each Contact as:

0045 CRMCS – Contact - Paul McQuillan

Is DocMan added to a CRM form as a Web Resource?
Yes – this is added to any entity as a Web Resource and then has a number of configurable parameters. (group by X, sort by Y, and so on)

In terms of the demo, we'd be grateful if you could also demonstrate how DocMan would function on a site with multiple document libraries with hundreds of documents as this would be more reflective of our environment. We'd also appreciate it if you could demo the Site Creation process if possible.
I have this configured for the demo on Wednesday – using a script to generate several hundred example records and therefore example SharePoint Sites.

From there, we can review how new Records are created as New Sites and how this works with Templates.

Our background with SharePoint and CRM
Paul McQuillan in our team has been working with Dynamics CRM and SharePoint since the initial release of Microsoft CRM v1.2 and SharePoint 2003 – and since the introduction of SharePoint Integration in CRM 2011 has built a number of different integrations for various projects, and written articles and blogs on how SharePoint can be used to bring Document Management best practise into CRM.

https://crmconsultancy.wordpress.com/2011/10/23/crm-2011-integration-with-sharepoint-taking-a-deeper-look/

https://crmconsultancy.wordpress.com/2011/10/27/crm-2011-integration-with-sharepoint-custom-document-management/

DocMan is the culmination of these efforts into a more App-based Product to deliver integration 'out-of-the-box'. (or in this case, out-of-the-app)

http://www.crmcs.co.uk/docman-for-dynamics/

<u>Q/A Bullet Points</u>

- **Rename a Document Library or Site**
  - ***Changes in CRM***
    - The Name of a Document Library, Site or Folder is controlled by CRM
    - So if the record is changed in CRM, this will automatically rename the corresponding SharePoint Area to match
  - ***Changes in SharePoint***
    - However changes in SharePoint are not passed back to CRM
    - So CRM will look for the Document Location according to the Document Locations in CRM, find the Library is no longer present, and then raise an error.
    - Following this error, CRM can then offer the option to recreate the SharePoint Area as it should be according to the rules governing how a CRM Record should exist in SharePoint.
    - Alternatively, we can manually connect the CRM Record to the new location of the SharePoint Site, Document Library or Folder.
- **Remove a Document library or Site**
  - Same as Renaming in going from CRM to SharePoint, but not SharePoint to CRM

  - Deletions in CRM can (as an optional setting) pass through and delete in SharePoint. (move to the SharePoint Recycle Bin)
  - Deletions in SharePoint will prompt CRM to raise an error as the Document Location as been removed.
- **Remove a user from a security group**
  - Access Group in CRM
    - We maintain a number of Access Groups as Teams in CRM
      - Each Access Group being associated to a number of CRM Users via an N:N Relationship
    - Each Access Group is synchronised to SharePoint as a Security Group – so any changes in CRM are passed through to SharePoint
    - Removing a User from the Access Group in CRM, then removes the User from the Security Group in SharePoint
  - Security Group in SharePoint
    - Changing the Security Settings in SharePoint will not change CRM
    - So the changes will be overwritten by the configuration in CRM when the settings are next synchronised across.
- **Removes a site column from a library/content type**
  - If the Site Column is removed in SharePoint then we have the similar point that this is not passed back to CRM
  - This means that when CRM is re-published to SharePoint, the Site Column will be re-added to SharePoint
  - To delete a Site Column completely, we would:
    - Remove the Column from CRM
    - Delete the Column from SharePoint
- **Change permissions on a library**
  - **Inherit Permissions**
    - If inherit permissions is active for a record (say for a Contact linked to parent Account that inherits permissions from the Account)
    - Then is simple as the Library or Site for a record inherits the permissions from the parent record.
  - **Unique Permissions**
    - This can be set for every record or be just for the record at the top of the tree (the Account in the above example)
    - We then have a set of 'stakeholders' in CRM for that record
    - This defines the permissions that different Users or Access Groups have to the Documents for that record.
    - Each User or Access Group added to the list of Stakeholders is then given Read or Contributor Rights in SharePoint for the Document Location corresponding to that Record.
    - Combined with inheritance, this then gives those same rights to the records down the inheritance tree – so in the above example, the permissions to Contact Documents would be controlled by the list of Stakeholders for the Parent Account.
    - Typically Stakeholders would be defined with some automatic rules in CRM:
      - So the Owner of the Account would be automatically added as a Stakeholder
      - Or the Team for the Owner of the Account would automatically be added as a Stakeholder.
  - **Permissions from CRM to SharePoint**

- The permissions are passed from CRM to SharePoint by an Asynchronous Job that triggers whenever the set of Stakeholders is changed in CRM.
- This Asynchronous Job loops through the list of Users and Access Groups added as Stakeholders and updates SharePoint accordingly – creating, removing or leaving in place the list of Users and Security Groups for that SharePoint Site.
- This then works with inheritance where possible as the Child Sites will then inherit the same permissions in SharePoint without further action.