CopyMove for SharePoint 2016 Administrators Guide

Applies to product version 4.0

Printed 31-05-2016

© 2008 - 2016 SharePoint Products

CopyMove for SharePoint 2016 Administrators Guide

© 2008 - 2016 SharePoint Products

All rights reserved. No parts of this work may be reproduced in any form or by any means - graphic, electronic, or mechanical, including photocopying, recording, taping, or information storage and retrieval systems - without the written permission of the publisher.

Products that are referred to in this document may be either trademarks and/or registered trademarks of the respective owners. The publisher and the author make no claim to these trademarks.

While every precaution has been taken in the preparation of this document, the publisher and the author assume no responsibility for errors or omissions, or for damages resulting from the use of information contained in this document or from the use of programs and source code that may accompany it. In no event shall the publisher and the author be liable for any loss of profit or any other commercial damage caused or alleged to have been caused directly or indirectly by this document.

Table of Contents

Part I Introduction 6

1 Audience ... 6
2 Product history ... 7
3 Product Features ... 8
4 Supported List Types ... 12

Part II Installation & Setup 14

1 Product Files ... 14
2 System Requirements ... 15
3 Making CopyMove Available ... 16
4 Scripted Installation ... 17
5 Activating the CopyMove Features ... 18
Activating CopyMove at the Farm level ... 18
Activating CopyMove for selected Site Collections ... 21
6 Upgrading CopyMove ... 23
7 Upgrading from CopyMove 2013 ... 23
8 Uninstalling CopyMove ... 24
9 Manual Installation ... 25

Part III Administration 30

1 License Management ... 33
2 Global Settings ... 37
3 Site Collection Settings ... 43
4 Permissions Management ... 45
Granting access with a new CopyMove permission level ... 47
Granting access through an existing permission level ... 50
5 Personal Links Location ... 51
6 Auditing of Copy and Move events ... 52
7 Timer Jobs ... 55
8 Diagnostic Logging ... 56

Part IV PowerShell Cmdlets 59

1 Get-SPPLicense ... 60
2 Set-SPPLicense ... 60
3 Get-CopyMoveSettings ... 61
4 Set-CopyMoveSettings ... 61
5 Remove-CopyMoveSettings ... 63

© 2008 - 2016 SharePoint Products

6 Set-CopyMovePermissionLevel ... 64
7 Copy-SPPItem ... 65
8 Move-SPPItem ... 67
9 Export-SPPItem ... 69
10 Export-SPPList ... 71
11 Import-SPPItem ... 72

Part V Extending CopyMove 75

1 API ... 75
CopyMove .NET API ... 82
CopyMove SOAP Web Service ... 86
CopyMove REST Web Service ... 89
CopyMove JavaScript API ... 92
2 Using CopyMove in SharePoint 2010 Workflows ... 94
3 Using CopyMove in SharePoint 2013 Workflows ... 99
4 Registering CopyMove with Custom Lists ... 107
5 Detecting CopyMove in SharePoint Event Receivers ... 111

Part VI Troubleshooting 114

1 Installation Error - Cannot access the local farm ... 114
2 Unexpected error in the CopyMove dialog ... 116
3 How to report a problem to the CopyMove support team ... 117

© 2008 - 2016 SharePoint Products

Part I

Introduction

1 Introduction

CopyMove for SharePoint 2016 is a commercial add-on for Microsoft SharePoint Server 2016. It enables end-users to easily and safely copy and move files, list items and folders with full fidelity. That is, CopyMove will preserve timestamps, author information, meta-data, version history and item level permissions. There is also an export and import function that enables export of content from one SharePoint farm and import to another farm.

The product is primarily designed for end-users who need to migrate smaller amounts of content at a time. It can of course also be used by administrators - but should not be confused with a full blown migration tool to migrate large volumes of content. The maximum size of one CopyMove transaction is 2GB for normal users and 10GB for site collection administrators.

Technically, CopyMove performs all the work through the official SharePoint server side .NET API. The product is currently only available for SharePoint 2016 on-premises. But we are also looking into an offering for Office 365.

All copy and move transactions follow the procedure outlined below:

1. User selects the items to copy or move.
2. User selects the destination location where the selected items should be copied or moved to.
3. CopyMove validates the requested transaction and reports any errors and warnings before the transaction is initiated. Errors will abort the transaction while warnings must be confirmed by the user to initiate the transaction.
4. CopyMove exports the selected items to disk on the SharePoint Web front-end server handling the request.
5. CopyMove imports the items again from disk to the selected destination location.
6. If it is a move transaction and the import operation was successful, CopyMove then deletes the selected items in the source location.
7. CopyMove logs the transaction to the SharePoint audit log if enabled.
8. CopyMove deletes the exported items on disk.

1.1 Audience

This product manual is intended for SharePoint server administrators who need to install and maintain CopyMove for SharePoint 2016. Secondly, it also targets SharePoint developers who need to extend CopyMove programmatically. The chapters **Installation & Setup** and **Administration** are written for the SharePoint administrator while the **Extending CopyMove** chapter is designated to the SharePoint developer.

© 2008 - 2016 SharePoint Products

1.2 Product history

The first version of CopyMove for SharePoint 2007 was developed early 2008 for one specific client who had long sought a good and user friendly tool for copying and moving documents in SharePoint. It was deployed to their production environment in the autumn of 2008 and is still pleasing the end-users there. The product then received some more work to take it from a single-client product to a shrink-wrap product, which is quite a different world in terms of product completeness and stability. The first shrink-wrap version was released to Web on www.sharepointproducts.com in March 2009.

In the following years CopyMove was downloaded and evaluated by thousands of registered users and was in turn also purchased by many. The product now has a strong customer base and we have listened to a lot of feedback over the years and improved the product accordingly. The constant improvements has turned it into a very usable and stable product that will work in almost any SharePoint environment. CopyMove has been deployed by customers to single-server SharePoint farms as well as multi-server SharePoint farms consisting of 10+ servers.

In May 2010 a new major version of SharePoint from Microsoft hit the market, and it quickly became evident that CopyMove was still needed to fill the gap of a safe, fast and user friendly way to copy and move content. SharePoint 2010 did not bring any improvements over SharePoint 2007 in this respect. Consequently, we decided to invest our efforts in developing a major upgrade of CopyMove to provide support for SP2010 as well. Much work was required to support the new SharePoint user interface and new functionality like document sets and managed Metadata. Finally, late 2010 we released the first version of CopyMove for SharePoint 2010.

SharePoint 2013 was also not shipping with a solid Copy&Move function out-of-the-box. Hence, we again offered a major upgrade of CopyMove to support the new version of SharePoint. CopyMove 2013 was released in June 2013 and is basically a 1:1 upgrade from CopyMove 2010.

Same story with SharePoint 2016 - still no good and Copy&Move function! As a result, CopyMove 2016 was prepared and released in June 2016.

One should think that developing a tool for copying and moving stuff in SharePoint is a simple endeavor - but it's not! We first tried to rely on the SharePoint migration framework (the SPExport and SPImport classes) - but that quickly turned out to be a dead end as this framework is seriously flawed for use in end-user Web requests. Instead, we had to invest in the development of our own migration framework on top of the SharePoint object model. It was a big investment but worth it today. It is very flexible, extensible and stable and serves as a good engine for CopyMove. Let's for a moment go back to why SharePoint does not make it easy to copy and move content. First of all it is difficult because documents and list items can be associated with a huge array of different artifacts that can take a lot of work to recreate programmatically in a new location. The list of possible artifacts include timestamps, user information, version history, content types, Web parts, properties in a property bag, 10+ types of different Metadata columns, content approval, ratings, work flows, item permissions, unique document ids and barcodes. Next, there are several types of documents that all work through different parts of the SharePoint API and in very different ways. Types of documents include regular document files, Web part pages, Wiki pages, Publishing pages and document sets. Next, it takes a lot of time to perfect a migration framework like the one we built because of the complexity involved in supporting all the different formats and artifacts in SharePoint. Also, integrating so wide and deeply with the SharePoint API inevitably gets you into corners where SharePoint has unexpected and undocumented behavior. Finally, CopyMove has also grown into a complex product under the hood because it uses a lot of effort to validate each transaction to guide and protect users from loosing data. In other words, creating a fool-proof tool is also a complex task. See the section Product Features for more details on transaction

© 2008 - 2016 SharePoint Products

validation.

1.3 Product Features

CopyMove was designed with the following mindset:

- It must be easy to install, upgrade and uninstall the product.
- It must look nice and be easy to use.
- It must provide seamless integration with the SharePoint Web user interface.
- It must work exclusively through the official SharePoint object model (.NET API).
- It must work fast and perform copy and move tasks without making the user wait longer than necessary.
- It must never alter any configuration settings in document libraries and lists like adding a new content type because it is missing in the destination.
- It must be extensible.

The following table outlines the features included in the product.

Feature	Description
Copy and Move documents	Enables users to copy and move selected documents across folders, lists, sites and site collections within the same farm.
Copy and Move list items	Enables users to copy and move selected list items across folders, lists, sites and site collections within the same farm.
Copy and Move folders	Enables users to copy and move complete folder structures including all content. However, users are prevented from performing a move in the event that the folder contains one or more items and/or sub-folders with unique permissions preventing the current user from deleting them.
Copy and Move Web part pages	Enables users to copy and move Web part pages including any Web parts.
Copy and Move Wiki pages	Enables users to copy and move Wiki pages including contents and Web parts.
Copy and Move publishing pages	Enables users to copy and move Publishing pages including contents and Web parts.
Copy and Move document sets	Enables users to copy and move complete document sets.
Copy and Move across site collections	Enables users to copy and move content across any accessible site collections within the same farm.
Export and Import documents	Enables users to export documents to a ZIP archive and import them again to a new location in any SharePoint farm with CopyMove installed. The document version history, Meta data and item permissions are all recorded to a file named manifest.xml on export. The import function relies on this manifest.xml file to import the documents again with full fidelity. However, the import function also recognizes plain ZIP archives that was not created by the CopyMove export

© 2008 - 2016 SharePoint Products

Feature	Description
	function. This is handy to quickly import multiple new documents and folders to SharePoint from a local hard drive or a shared folder.
Export and Import list items	Enables users to export list items to a ZIP archive and import them again to a new location in any SharePoint farm with CopyMove installed.
Export and Import folders	Enables users to export and import complete folder structures.
Export and Import document sets	Enables users to export and import complete document sets.
Version history preservation	CopyMove can preserve the full version history for files, folders and list items unless version history has not been enabled for the destination location.
Meta data preservation	CopyMove preserves all Meta data for files, folders and list items including the created date, last modified date, created by user, last modified by user, content type, other system properties as well as all custom properties. Managed Meta data is also supported.
Item permissions preservation	Unique permissions on files, folders and list items can also be preserved .
Unique Document ID preservation	Move operations will preserve any assigned document id's while copy operations will assign new id's to the document copies.
Limit access by permissions	Site collection administrators and site owners can limit access to the CopyMove functions using the standard permissions model in SharePoint.
Enforce required fields	CopyMove will enforce required fields when copying or moving items to another library or list with additional required meta data columns. The user is in case asked to supply values for the missing columns on the source items.
Warning on file overwrite	Warns users if they are about to overwrite any files in the destination location.
Warning on loss of version history	Copying, moving and importing files and list items from lists with version history enabled to another list without versioning, implies that the version history is lost. In this situation, CopyMove issues a warning to the user before the transaction is started.
Warning on loss of Content Type information	Copying, moving and importing files and list items associated with Content Types that do not exist in the destination list, implies that this information is lost. In this situation CopyMove issues a warning to the user before the transaction is started.
Warning on loss of Metadata	Copying, moving and importing files and list items associated with custom Meta data in list columns that do not exist in the destination list, implies that this information is lost. In this situation CopyMove issues a warning to the user before the transaction is started.
Cross browser support	CopyMove has been tested with Internet Explorer 7, 8 and 9 plus FireFox and Google Chrome.
Seamless integration	Users can access the Copy, Move, Export and Import functions directly from the Ribbon in list views.
Administrators can configure transaction limits	Moving and copying large volumes of data in one transaction can be very time consuming and resource intensive leading users to believe that the application has entered a deadlock. Consequently, CopyMove enforces a limit on the number of files and the total MB that each user is allowed to select for one

© 2008 - 2016 SharePoint Products

Feature	Description
	transaction. The limit can be configured by SharePoint administrators from SharePoint Central Administration - but only up a ceiling of 2000 files / 2 GB for users and 10000 files / 10 GB for site collection administrators.
Administrators can configure farm wide settings for CopyMove	CopyMove has several configuration options that SharePoint administrators can configure for CopyMove throughout a SharePoint farm.
Site collection administrators can override farm settings	By default CopyMove operates with the farm wide settings. But site collection administrators can choose to override most of these settings for anyone or their site collections. However, the transaction limits cannot be overridden by site collections - the limits specified by the SharePoint administrator always applies to CopyMove throughout the farm.
User profile integration	CopyMove includes special support for My Sites and Followed Sites in the destination selection tree.
Workflow integration	CopyMove 2016 also installs a custom workflow action to enable workflow designers to use the Copy function in SharePoint Designer workflows.
.NET API	For server-side programmatic access to the Copy and Move functions.
JavaScript Plug-in API	Client side API for interacting with the CopyMove UI and CopyMove transactions.
SOAP Web Service	For remote programmatic access to the Copy and Move functions.
REST Web Service	For remote programmatic access to the Copy and Move functions.
Logs transactions to the Audit log	CopyMove automatically logs all transactions to the SharePoint audit log if it has been enabled for the site collection in question.
Support for custom list templates	The CopyMove item actions are also available in lists created from custom list templates that leverages the Ribbon control. That is, the CopyMove actions are registered for specific Ribbon locations regardless of list type.

Some of the features mention that CopyMove validates Copy and Move transactions before executing them and this is indeed true. The pre-validation protects users from inadvertent loss of data and it also minimizes the risk that the transaction will fail half-way. CopyMove specifically issues a warning when it finds one or more of the following conditions to be true:

- The user copies, moves or imports a file that already exists in the selected destination folder. The file will on approval of the warning, be overwritten.
- The user copies, moves or imports items from a list with version history enabled to another list where version history is not enabled. The version history will on approval of the warning, be lost in the transaction.
- The user copies, moves or imports items associated with a content type that does not exist in the destination list. The items will on approval of the warning be associated with the default content type in the destination list.
- The user copies, moves or imports items with custom Metadata columns that are not present in the destination list. The items will on approval of the warning loose all Metadata values for the missing columns.

© 2008 - 2016 SharePoint Products

Then there are also the conditions where CopyMove simply aborts the transaction with a validation error:

- The user copies, moves or imports a file that already exists in the selected destination folder and is checked out.
- The user copies, moves or imports a file that already exists in the selected destination folder and the file has special permissions that does not allow the user to overwrite it.
- The user moves a folder that contains one or more files or sub folders that the user is not allowed to delete.
- The user copies or moves items to a location where the resulting URL would exceed 256 characters. SharePoint does not support longer URLs than that.
- The user is trying to initiate a transaction larger than the administrative limit, which is by default 250 items or 250 MB. SharePoint administrators can increase the limit to a maximum of 2000 items and 2000 MB.

© 2008 - 2016 SharePoint Products

1.4 Supported List Types

The product does out-of-the-box integrate with all SharePoint document libraries and lists that leverage the Ribbon control in SharePoint 2016. That is, a new Ribbon group named CopyMove will appear in the ribbon for all libraries and lists in site collections where CopyMove is enabled. However, for the item drop-down menu in libraries and lists, CopyMove will out-of-the-box only integrate with the libraries and lists created from the SharePoint list templates shown in the table below.

List Template ID	List Template Name
100	Custom list
101	Document library
103	Links list
201	Agenda list
104	Announcement list
105	Contacts list
106	Calendar list
107	Task list
108	Discussion board
109	Picture library
119	Web page library
119	Wiki page library
150	Project tasks list
151	Help library
171	Tasks with Timeline and Hierarchy
202	Meeting Attendee list
204	Decision list
207	Meeting Objective list
211	Things to Bring list
301	Blog Posts list
700	My Site Document Library
850	Publishing page library
1100	Issue Tracking list

Lists created from other templates including custom ones will not include the Copy and Move actions in the item drop-down menu. However, it is possible to add support for such libraries and lists by developing and deploying a custom SharePoint feature that registers the CopyMove menu actions with the custom list template. For more information, see the **Custom Lists** section in the chapter named **Extending CopyMove**.

© 2008 - 2016 SharePoint Products

Part II

Installation & Setup

2 Installation & Setup

CopyMove is implemented according to Microsoft and SharePoint Community guidelines and best practices. It is packaged as a standard SharePoint farm solution, which means that it is easy to install and deploy in any SharePoint Server 2016 farm using standard SharePoint methods and tools. But CopyMove also ships with a PowerShell script that automates the entire installation and deployment process. The installation script must run locally on the SharePoint server. If you have a multi-server SharePoint farm then CopyMove will automatically be deployed to the other servers as needed via the standard SharePoint solution mechanism. Hereafter, you can activate and configure CopyMove as described in this manual.

In the event of a problem installing or using CopyMove, consult the **Troubleshooting** section, which provides answers to known problems. If that does not prove to be the case then contact support@sharepointproducts.com for assistance.

2.1 Product Files

CopyMove for SharePoint 2016 is available for download as a ZIP file from www.sharepointproducts.com. The ZIP file contains the following files:

Path	Description
EULA.rtf	The CopyMove End-User-License-Agreement.
Get-CopyMoveLogFile.cmd	Launches the PowerShell script PS\Get-CopyMoveLogFile.ps1
Install-CopyMove.cmd	Launches the PowerShell script PS\Install-CopyMove.ps1
ReadMe.htm	HTML document with brief product information and version history
Reset-LocalServer.cmd	Resets the Internet Information Server (IIS) and the SharePoint Timer Service on the local server
Reset-AllServers.cmd	Resets the Internet Information Server (IIS) and the SharePoint Timer Service on all servers in the SharePoint farm
Uninstall-CopyMove.cmd	Launches the PowerShell script PS\Uninstall-CopyMove.ps1
Upgrade-CopyMove.cmd	Launches the PowerShell script PS\Upgrade-CopyMove.ps1
CopyMove 2016 - Administrators Guide.c	Administrators guide for CopyMove 2016 in Windows Help format.
CopyMove 2016 - Administrators Guide.p	Administrators guide for CopyMove 2016 in PDF format.
PS\Functions.ps1	Shared PowerShell methods
PS\Get-CopyMoveLogFile.ps1	Retrieves all recent CopyMove events from the SharePoint diagnostic logs on all servers in the farm and writes them to a file named spp.log
PS\Install-CopyMove.ps1	PowerShell script that installs and deploys the WSP solution packages to the SharePoint farm
PS\Launch.cmd	Windows batch file for launching a specified PowerShell

© 2008 - 2016 SharePoint Products

Path	Description
	script
PS\Uninstall-CopyMove.cmd	PowerShell script that retracts and removes the WSP solution packages from the SharePoint farm
PS\Upgrade-CopyMove.ps1	PowerShell script that upgrades the WSP solution packages in the SharePoint farm
Samples\API Sample	Folder containing a Visual Studio 2012 C# solution demonstrating how to work with the CopyMove 2016 .NET API
Samples\Custom List Sample	Folder containing a Visual Studio 2012 C# solution demonstrating how to integrate CopyMove 2016 with document libraries and lists created from a custom list template.
Samples\PowerShell	Folder containing sample PowerShell scripts leveraging the CopyMove PowerShell cmdlets.
Samples\WCF Sample	Folder containing a Visual Studio 2012 C# solution demonstrating how to work with the CopyMove 2016 WCF Service.
WSP\sharepointproducts_copymove201	SharePoint 2016 solution package containing CopyMove 2016 specific files and resources
WSP\sharepointproducts_platform2016.v	SharePoint 2016 solution package containing cross product files and resources

2.2 System Requirements

CopyMove requires the following to install and operate as expected:

Server Requirements

- Microsoft Windows Server 2012 R2 or later
- Microsoft .NET Framework 4.5
- Microsoft PowerShell 3.0 or later
- SharePoint Server 2016
- 3 GB free disk space in the temporary files location. Default location is LOGS\SharePointProducts relative to the SharePoint root folder.

Client Requirements

- Microsoft Internet Explorer 9 or later.
- Microsoft Edge
- Mozilla FireFox
- Google Chrome

© 2008 - 2016 SharePoint Products

2.3 Making CopyMove Available

Complete the following steps to make CopyMove available to end-users:

1. **Read and accept the End-User License Agreement**
   The CopyMove EULA file is included in the product download as EULA.rtf

2. **Install and deploy the WSP solutions**
   Add the solutions *sharepointproducts_platform2016.wsp and sharepointproducts_copymove2016.wsp* to the SharePoint solution store and deploy them to all Web applications including SharePoint Central Administration. See the section **Scripted Installation** for detailed instructions.

3. **Deploy CopyMove to all relevant Web applications**
   This is automatically done by the scripted installation, but if you create new Web applications or choose not to deploy automatically, then you may need to do so manually. See the section **Manual Installation** for instructions.

4. **Activate the CopyMove farm feature**
   See the section **Activating CopyMove at the Farm level** for instructions.

5. **Activate CopyMove for all relevant site collections**
   The CopyMove menu actions only appear in document libraries and lists in site collections where the CopyMove site collection feature is activated. See the section **Activating CopyMove for selected Site Collections** for instructions.

6. **Purchase and install a CopyMove license**
   CopyMove is not a free product but you may evaluate it for free before buying. You are automatically granted a 30 day evaluation period counting from the day it was first installed. The product blocks all functionality once the evaluation period has expired. If you wish to continue using CopyMove in your SharePoint farm then you will need to purchase a license from www.sharepointproducts.com. See the section **License Management** for instructions how to install a license file received from the SharePoint Products sales team.

© 2008 - 2016 SharePoint Products

2.4 Scripted Installation

Follow the steps below to install CopyMove using the PowerShell installation script included in the product download:

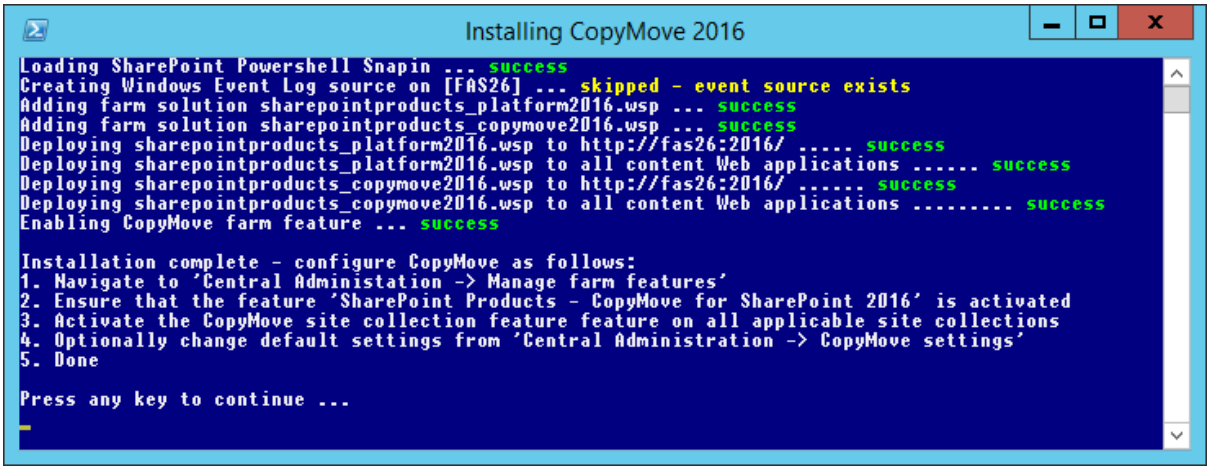
1. Login to the SharePoint server with a local administrator account that have also been granted the `db_owner` role on the SharePoint configuration database and the SharePoint administration content database. If your SharePoint installation is a multi-server farm then you can install on just one server after which the CopyMove files are automatically propagated to the other servers in the farm.

   TIP: Login with the Windows domain account that was used for installing and configuring SharePoint Server 2016.

2. Download the latest version of CopyMove 2016 from www.sharepointproducts.com. It is distributed as a ZIP archive.

3. Extract the CopyMove ZIP archive to a local folder on the SharePoint server.

4. Run the `Install-CopyMove.cmd` script from the local folder that the CopyMove files were extracted to, and wait for it to complete. The screen shot below illustrates how a successful installation should look like.



5. CopyMove is now installed and deployed to all Web applications. But it is not yet available to any users. For that you need to activate the CopyMove features as described in the next section.

© 2008 - 2016 SharePoint Products

2.5 Activating the CopyMove Features

The CopyMove product is enabled through a regular SharePoint "Feature", which means it can easily be activated or deactivated by SharePoint administrators and site collection owners. The product ships with the following two features:

- **Farm Feature**. This feature adds a group of CopyMove related administration links to SharePoint Central Administration. The feature must also be enabled prior to activating the Site Collection feature described below. When deactivated the administration links are again removed from Central Administration. It will also prevent further activations of the Site Collection feature. However, CopyMove will continue to be available in site collections for which one of these two features are already activated. The feature XML files are deployed to the folder `TEMPLATE\FEATURES\SPPCopyMoveFarm` relative to the SharePoint 16 root folder.

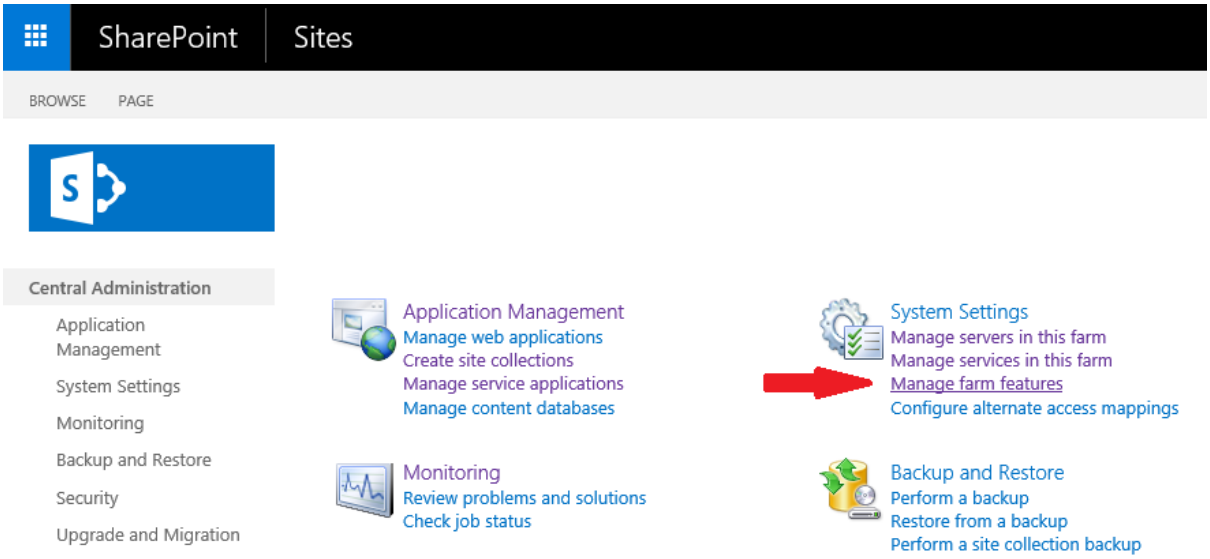
- **Site Collection Feature**. Activating this feature registers the CopyMove menu actions with all supported document libraries and lists in the site collection. It also adds a link to the CopyMove site settings page on the SharePoint site settings page. Deactivating the feature removes the menu actions and the site settings link again. The feature XML files are deployed to the folder `TEMPLATE\FEATURES\SPPCopyMoveSite` relative to the SharePoint 16 root folder.

2.5.1 Activating CopyMove at the Farm level

To activate or deactivate the CopyMove farm feature, follow the steps listed below:

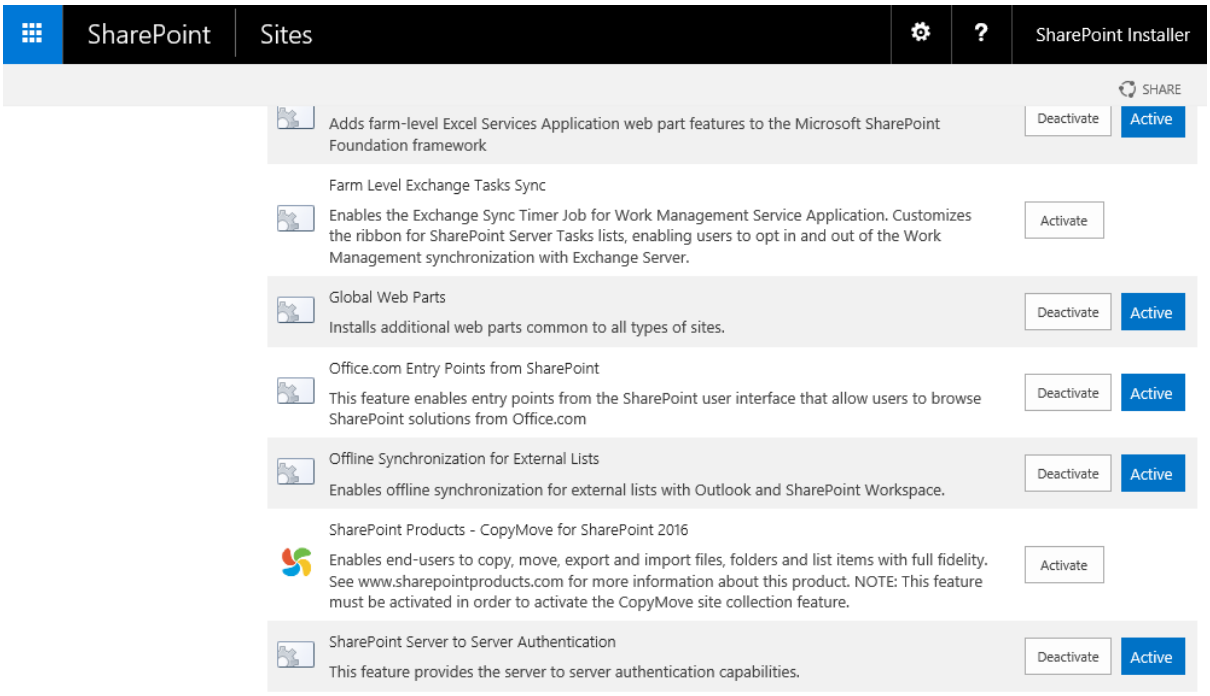
1. Open the SharePoint 2016 Central Administration site.

2. Click the *Manage farm features* link as shown in the following screen shot.



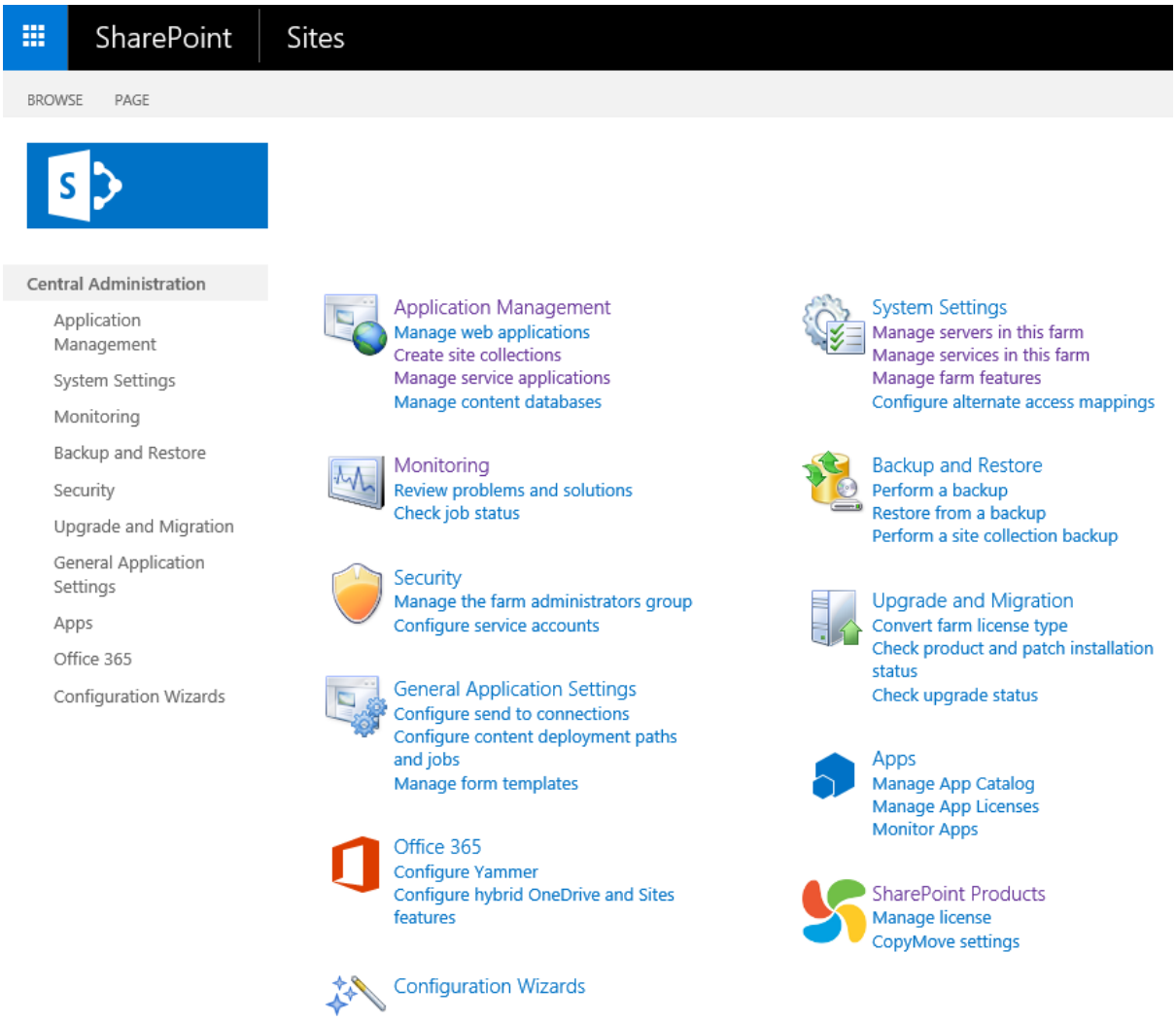
3. On the *Manage Farm Features* page, locate the feature entitled *SharePoint Products - CopyMove for SharePoint 2016* and click the *Activate* button. If the feature is already activated then you will get the option to deactivate the feature again.

© 2008 - 2016 SharePoint Products



4. Activating the feature will reveal a new administration group named *SharePoint Products* as shown below. See the **Administration** section for more information about the administration links in this group.

© 2008 - 2016 SharePoint Products



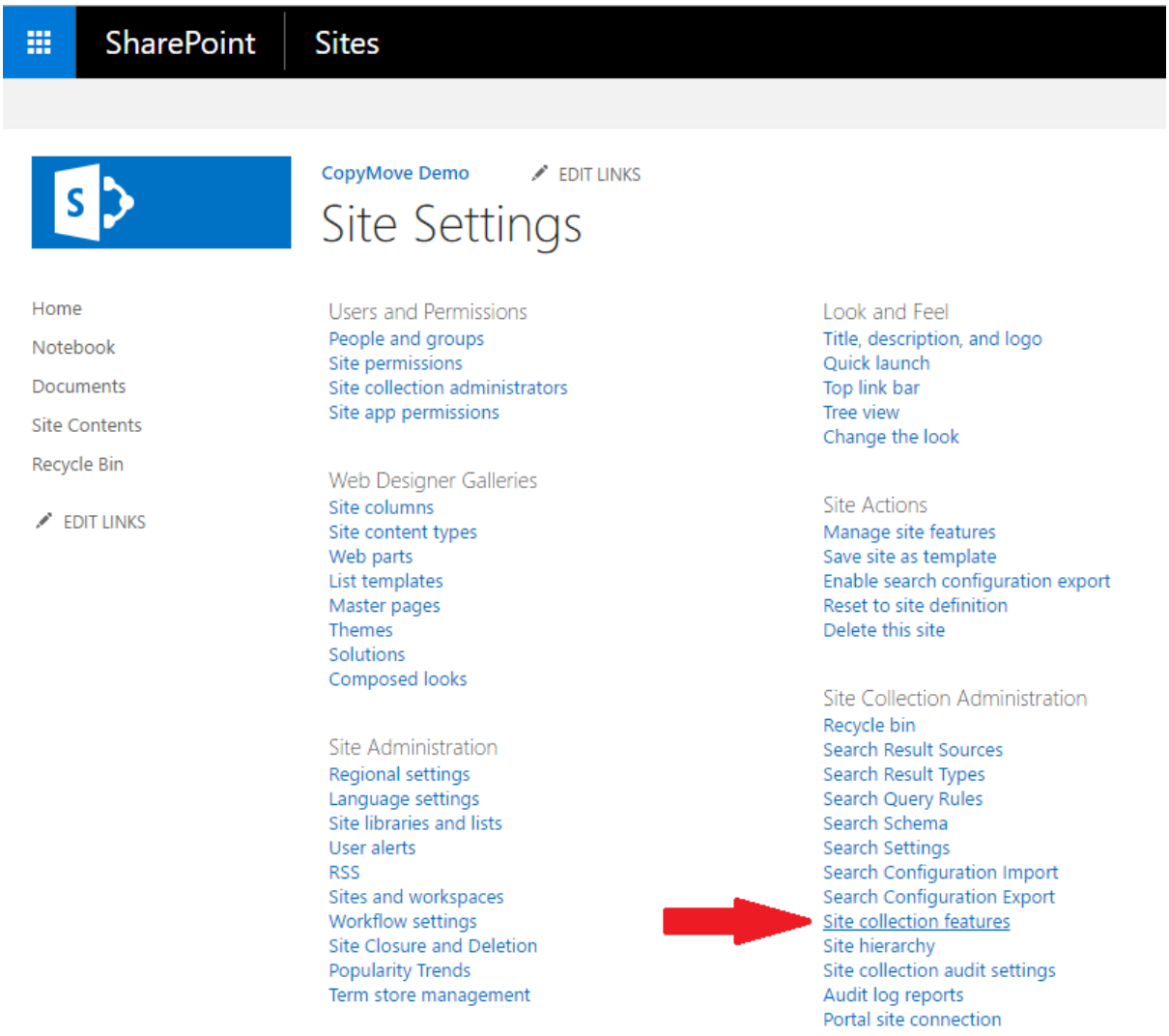
© 2008 - 2016 SharePoint Products

2.5.2 Activating CopyMove for selected Site Collections

To activate or deactivate the CopyMove feature at the level of the site collection, follow these steps:

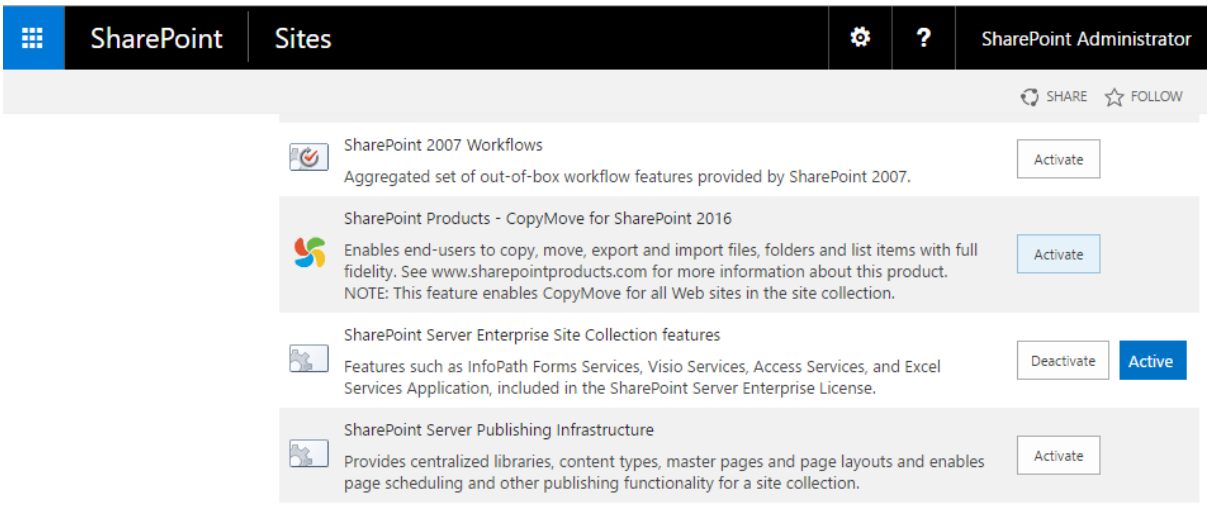
1. Open the root site of the site collection for which to activate or deactivate CopyMove.

2. Open the *Site Settings* page shown below.



3. Click the *Site collection features* link in the *Site Collection Administration* group. This brings up the site collection features page where the activation status of each installed feature is shown.

© 2008 - 2016 SharePoint Products



4. To toggle the activation status of CopyMove, click the *Activate/Deactivate* button for the feature labeled *SharePoint Products - CopyMove for SharePoint 2016*.

© 2008 - 2016 SharePoint Products

2.6 Upgrading CopyMove

We at SharePoint Products are continuously improving CopyMove based on customer feedback and own ideas. CopyMove receives a major upgrade with every new version of SharePoint and minor upgrades when we are adding new features. Finally we also do maintenance releases with minor improvements and bug fixes. We typically make 1-2 minor releases a year and 6-10 maintenance releases a year. The complete change history can be studied in the ReadMe.htm file included in the distribution.

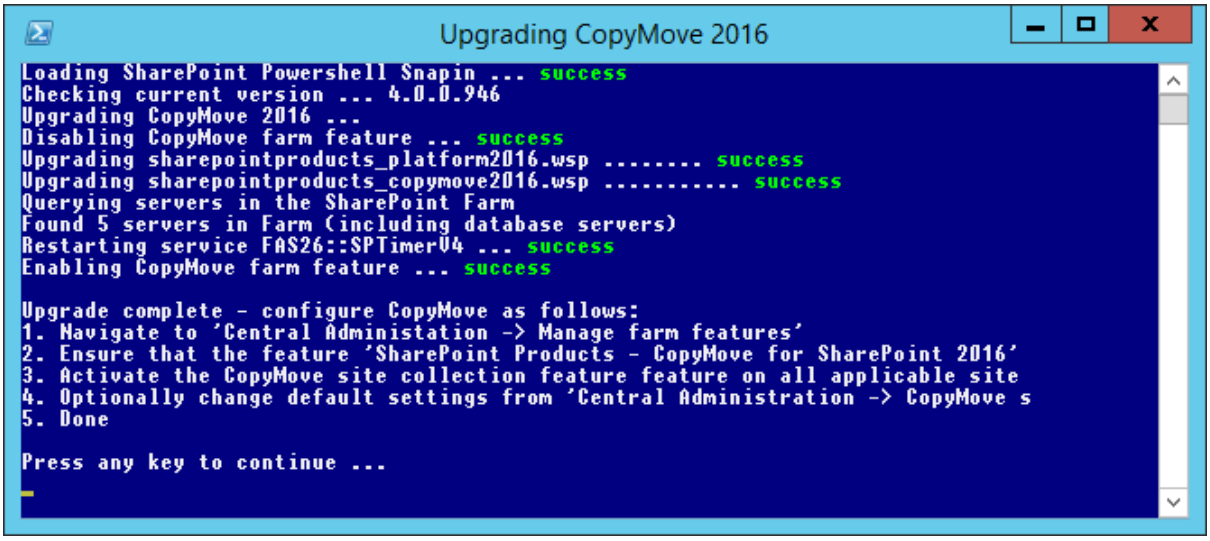
Consequently, you should periodically watch out for new releases of CopyMove and upgrade whenever the changes can add more value to your SharePoint installation. Follow the steps below to upgrade from earlier versions of CopyMove:

1. Login to the SharePoint server with a local administrator account that have also been granted the `db_owner` role on the SharePoint configuration database and the SharePoint administration content database. If your SharePoint installation is a multi-server farm then you can upgrade on just one server after which the CopyMove files are automatically upgraded on the other servers in the farm.

2. Download the latest version of CopyMove 2016 from www.sharepointproducts.com. It is distributed as a ZIP archive.

3. Extract the CopyMove ZIP archive to a local folder on the SharePoint server.

4. Run the `Upgrade-CopyMove.cmd` script from the local folder that the CopyMove files were extracted to, and wait for it to complete. The screen shot below illustrates how a successful upgrade should look like.



5. CopyMove is now upgraded for the entire SharePoint farm.

2.7 Upgrading from CopyMove 2013

CopyMove 2013 can only be installed to SharePoint 2013 servers. It cannot be installed to SharePoint 2016 servers! Hence, upgrading from SharePoint 2013 to SharePoint 2016 also requires an upgrade of

© 2008 - 2016 SharePoint Products

CopyMove 2013 to CopyMove 2016. Fortunately, upgrading from CopyMove 2013 to CopyMove 2016 is straight forward:

1. Leave CopyMove 2013 installed in the SharePoint 2013 farm.

2. Install CopyMove 2016 to the SharePoint 2016 farm before migrating any of the SharePoint 2013 content databases. You will otherwise get warnings of missing feature templates in the site collections where CopyMove 2013 was activated.

3. Backup, restore and attach the SharePoint 2013 content database(s) to the SharePoint 2016 farm. SharePoint will then upgrade the database(s).

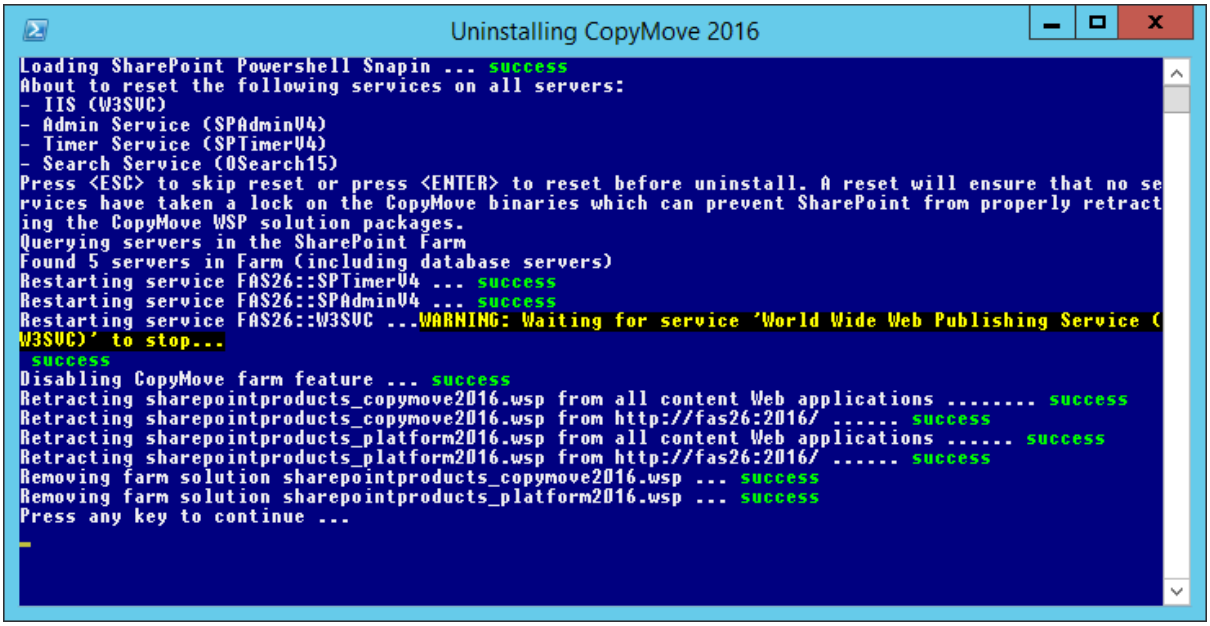
Note: The license key for CopyMove 2013 also works with CopyMove 2016 provided that the support and upgrade subscription has not expired.

2.8 Uninstalling CopyMove

Removing CopyMove from a farm is of course also possible and it is just as simple as installing it. Simply follow the steps below to uninstall CopyMove:

1. Login to the SharePoint server with a local administrator account that have also been granted the `db_owner` role on the SharePoint configuration database and the SharePoint administration content database. If your SharePoint installation is a multi-server farm then you can uninstall on just one server after which the CopyMove files are automatically removed from the other servers in the farm.

2. Run the `Uninstall-CopyMove.cmd` script from the local folder that the CopyMove files were extracted to, and wait for it to complete. The screen shot below illustrates how a successful uninstall should look like.



© 2008 - 2016 SharePoint Products

3. CopyMove is now retracted from all Web applications and the WSP solutions have been removed from the SharePoint solution store.

2.9 Manual Installation

CopyMove can also be installed and deployed manually to a SharePoint farm. You can use this approach if you are experiencing any problems with the supplied PowerShell scripts for installing, upgrading and uninstalling CopyMove. To manually install and deploy CopyMove, follow the steps below:

1. Login to the SharePoint server with a local administrator account that have also been granted the `db_owner` role on the SharePoint configuration database and the SharePoint administration content database. If your SharePoint installation is a multi-server farm then you can install on just one server after which the CopyMove files are automatically propagated to the other servers in the farm.

   TIP: Login with the Windows domain account that was used for installing and configuring SharePoint Server 2016.

2. Download the latest version of CopyMove 2016 from www.sharepointproducts.com. It is distributed as a ZIP archive.

3. Extract the CopyMove ZIP archive to a local folder on the SharePoint server.

4. From the Windows Server start menu, launch the *SharePoint 2016 Management Shell*.

5. Add the solution packages *sharepointproducts_platform2016.wsp* and *sharepointproducts_copymove2016.wsp* to the SharePoint farm using the *Add-SPSolution* PowerShell command:

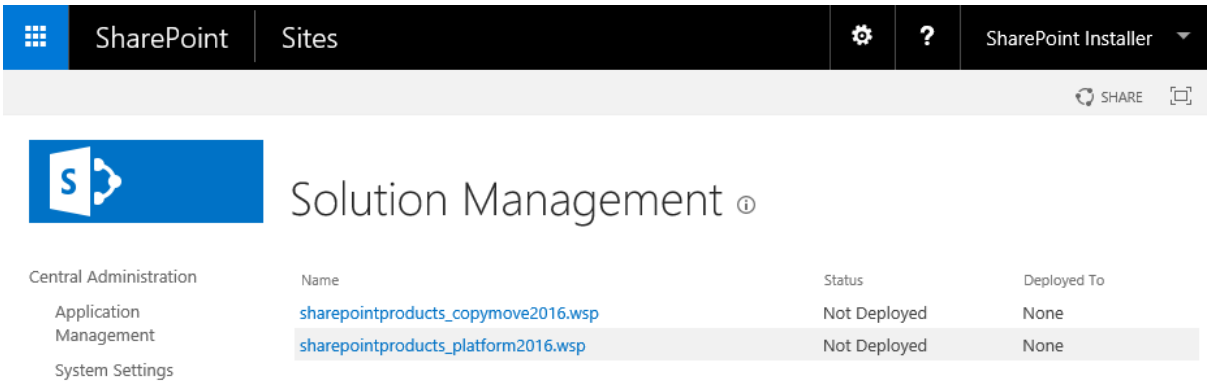
   ```
   Add-SPSolution <Full path to the sharepointproducts_platform2016.wsp
   file>
   Add-SPSolution <Full path to the sharepointproducts_copymove2016.wsp
   file>
   ```

   For example:
   ```
   Add-SPSolution C:\CopyMove2016\WSP\sharepointproducts_platform2016.wsp
   Add-SPSolution C:\CopyMove2016\WSP\sharepointproducts_copymove2016.wsp
   ```

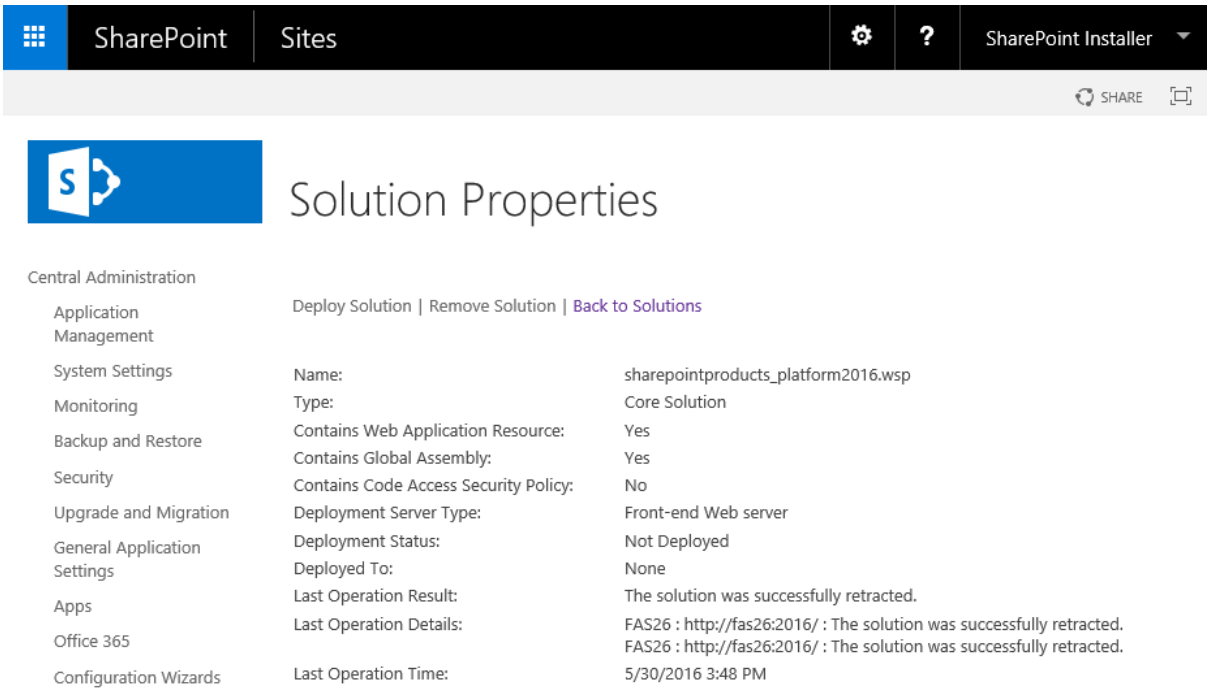
6. Open the SharePoint 2016 Central Administration site.

7. Navigate to *System Settings* and click the *Manage farm solutions* link to open the *Solution Management* page shown in the following screen shot:

© 2008 - 2016 SharePoint Products

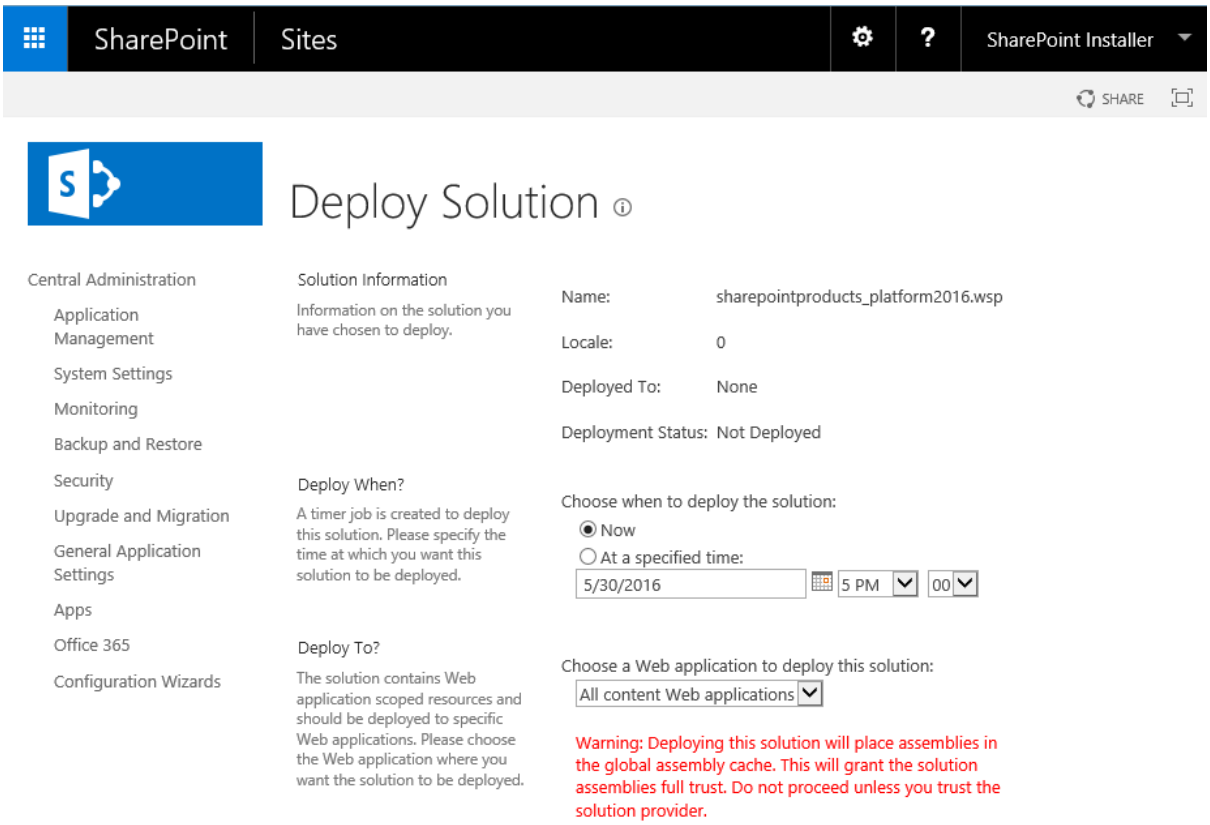


8. Click the *sharepointproducts_platform2016.wsp* link to open the *Solution Properties* page for the clicked solution.



9. Click the *Deploy Solution* link to open the following page from where a solution deployment job can be created.

© 2008 - 2016 SharePoint Products

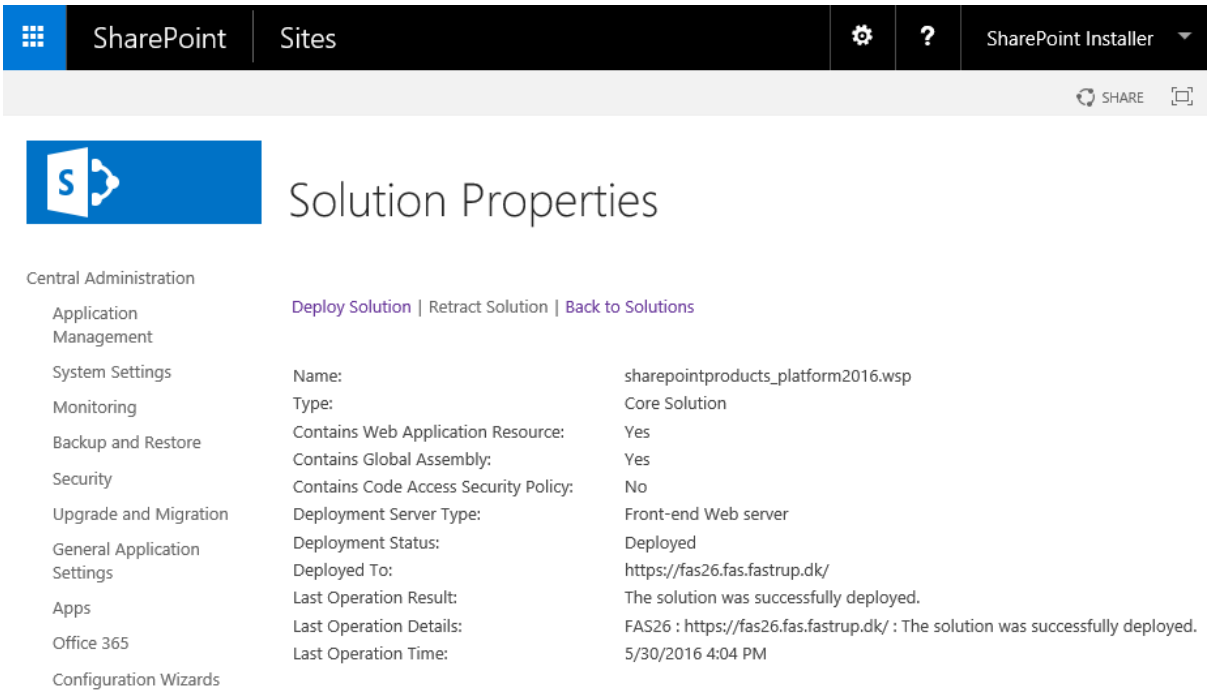


10. Now, create a deployment job that deploys the solution to all content Web applications in the farm. To do this, select *All content Web applications* in the *Deploy To* drop-down and click the *OK* button.

11. Next, create a deployment job that deploys the solution to the Web application hosting SharePoint Central Administration. To do this, select the URL of SharePoint Central Administration in the *Deploy To* drop-down and click the *OK* button.

12. Use the *Solution Properties* page to very that the solution was deployed without errors. The *Deployment Status* row must say Deployed, the *Deployed To* row should include the URLs of all Web applications in the farm and the *Last Operation Result* row should say *The solution was successfully deployed*.

© 2008 - 2016 SharePoint Products



13. Repeat steps 7 to 12 for the *sharepointproducts_copymove2016.wsp* solution package.

14. Finally, activate the necessary features as previously discussed in the **Activating the CopyMove Features** section.

© 2008 - 2016 SharePoint Products

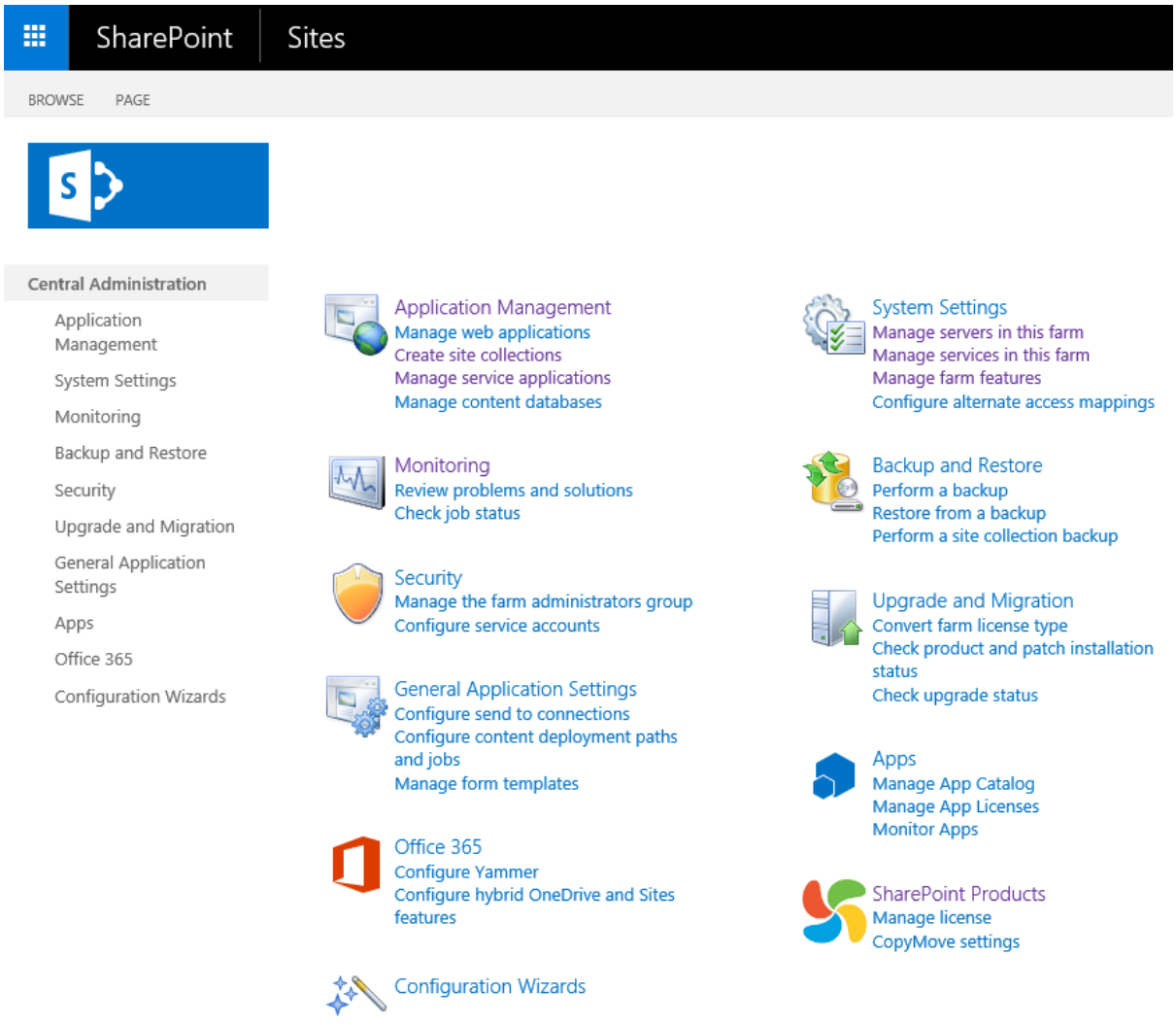
Part III

Administration

3 Administration

Like CopyMove is easy to install it is also easy to administrate through the SharePoint Web administration user interface that CopyMove integrates with. Two roles in SharePoint has the required permissions to activate CopyMove and to change the CopyMove configuration settings; the SharePoint farm administrator and the site collection administrator. The options for both roles are outlined below.

SharePoint Farm Administrator
This role can activate and deactivate the CopyMove features at all levels including the farm level and the site collection level. See the **Activating the CopyMove Features** section for more information. Next, the role can also configure the global CopyMove settings available from the SharePoint Central Administration site as shown in the screen shot below. The available settings are outlined and described in detail in the **Global Settings** section.



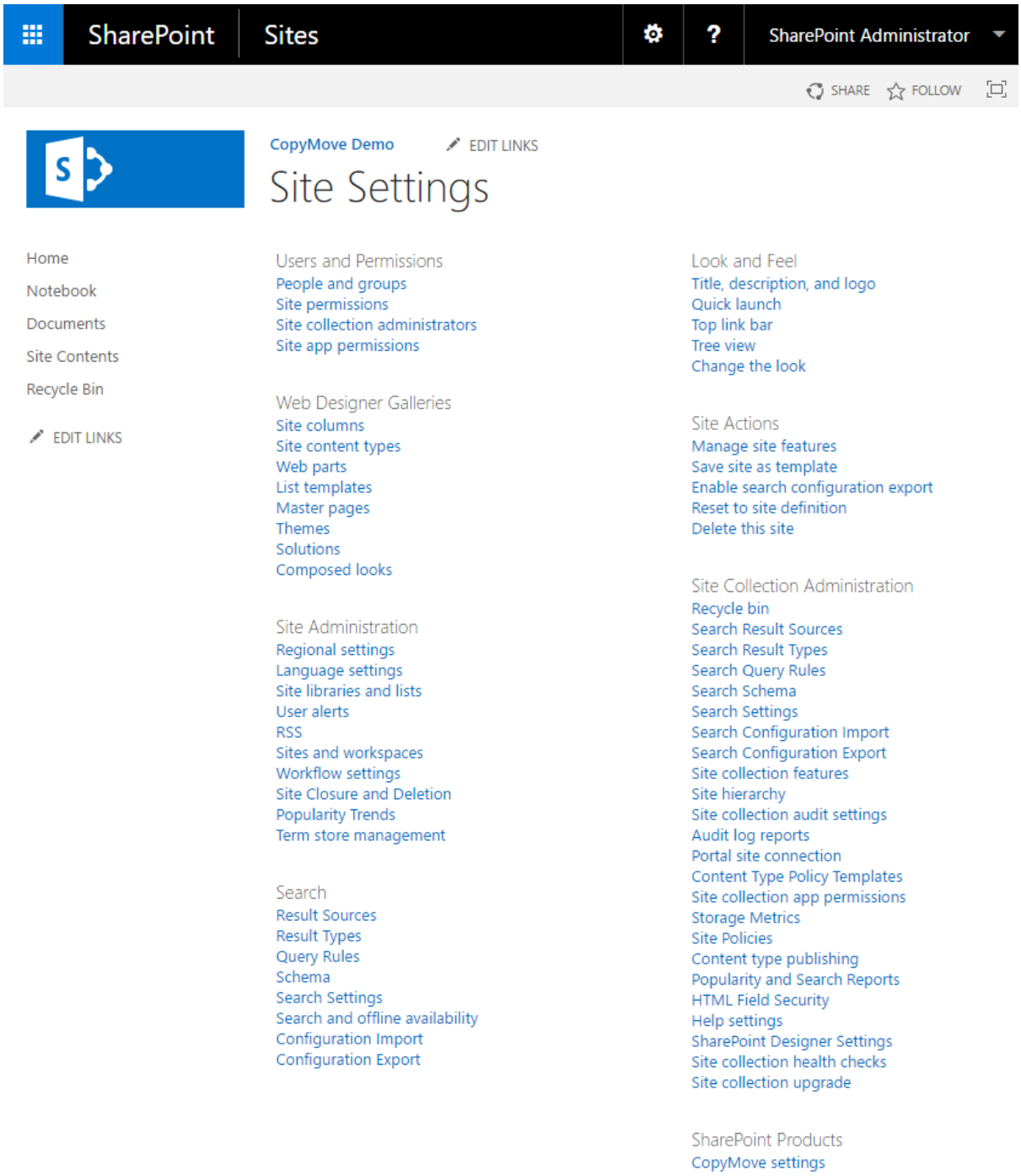
Finally, the role can also apply CopyMove license keys by clicking the *Manage license* link. See the **License Management** section for more information on applying a license key received from SharePoint

© 2008 - 2016 SharePoint Products

Products.

SharePoint Site Collection Administrator
This role can activate and deactivate CopyMove for the site collections they own - see the section **Activating CopyMove for selected Site Collections** for details. Next, the role can also configure special settings for site collections by overriding the global settings managed by the farm administrator. The following screen shot shows the SharePoint site settings page with a link to the CopyMove site collection settings page. The available settings are outlined and described in detail in the **Site Collection Settings** section.

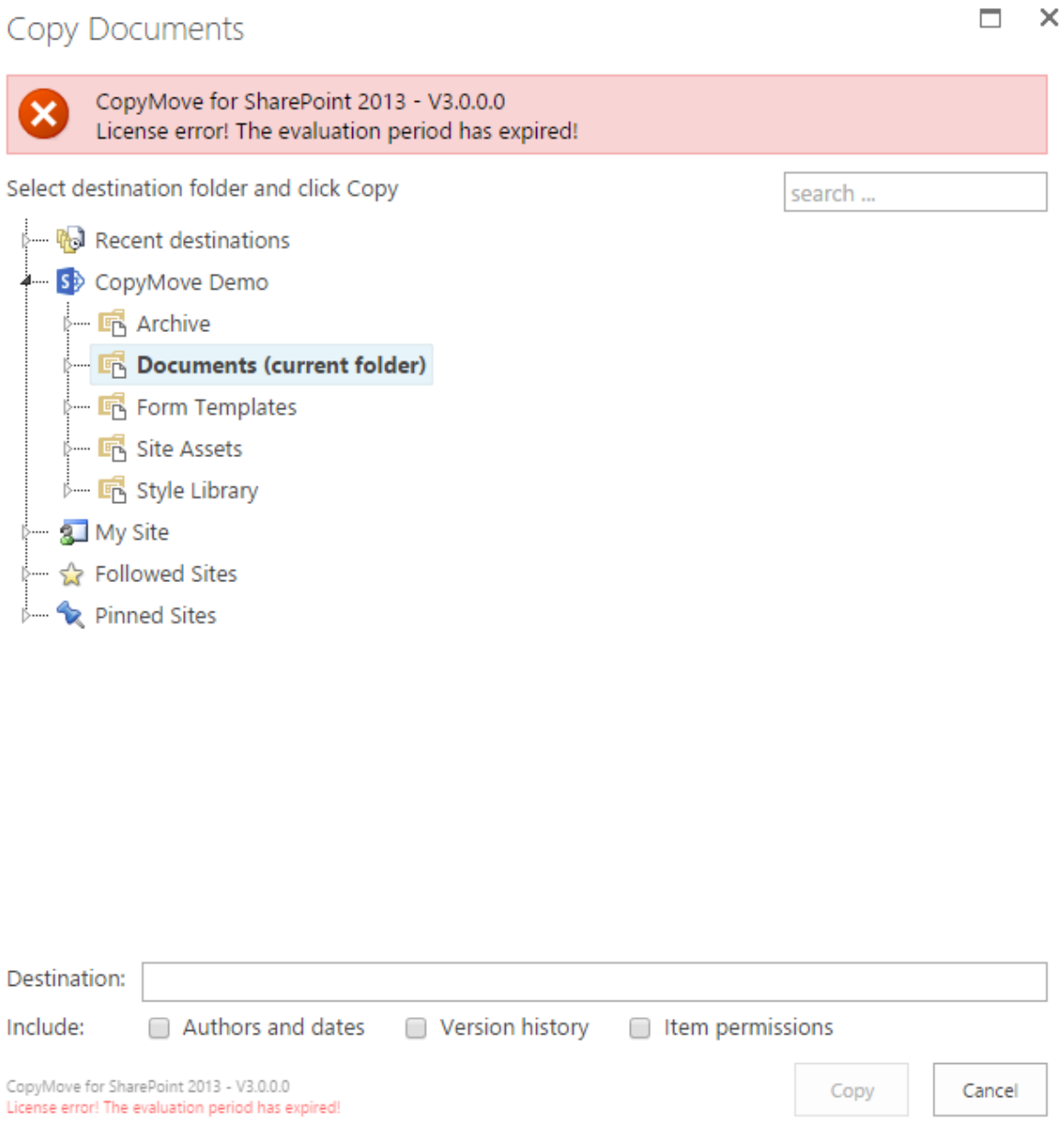
© 2008 - 2016 SharePoint Products



© 2008 - 2016 SharePoint Products

3.1 License Management

CopyMove requires customers to purchase and install a license key from SharePoint Products to continue using the product beyond the 30 day trial period. Users will otherwise receive the message; *License Error! The evaluation period has expired!* as shown in the screen shot below. Selecting a destination folder and pressing the *Copy* button will yield the same error.



The license for unlocking CopyMove is just a simple text file in the following format:

```
SharePoint Products License
Generated on 2013-06-10 14:04:07Z
```

© 2008 - 2016 SharePoint Products

```
Id: 2d34f79f-a3f0-4d4f-9f31-70e2342c7a9e
Customer: Demo Customer
Contact: John Doe, jdoe@democustomer.com
Issuer: Lars Fastrup, lars@sharepointproducts.com
Products: CopyMove
Edition: Standard
Expiration Date: Never
SU Expiration Date: 2012-05-02
Servers: 2
Clients: Unlimited
Evaluation: No
Comment: Production license.
------------------------------------------------------------
ClNQUExpY2Vuc2WAAAAAYCl+BMQfMvH3boR8prD/mztTHyeSqa0RHwo6Q0RS
U+8SWKmjxGBJ8cGp+2fSQ/FFEEA2Aa2FhT8sCuFZWnIPljc/SRZcdAVZ2Bo8
J8rjOWXwoALGMTySf9gb/0S3rI62+fxYQOq2mMEUdw+vpB09llwMVvcqzheq
SOIQHdwHbGUMAAJJZAaf9zQt9KNPTZ8xcOI0LHqeCEN1c3RvbWVyCw1EZW1v
IEN1c3RvbWVyB0NvbnRhY3QLH0pvaG5gRG9lLCBqZG9lQGRlbW9jdXN0b21l
ci5jb20GSXNzdWVyCylMYXJzIEZhc1RydXAsIGxhcnNAc2hhcmVwb2ludHBy
b2R1Y3RzLmNvbQhQcm9kdWN0cwELAQAAAAhDb3B5TW92ZQdFZGl0aW9uCwhT
dGFuZGFyZA9FeHBpcmF0aW9uIERhdGUF/z839HUoyisSU1UgRXhwaXJhdGlv
biBEYXRlBQAA2KYN9s4IB1NlcnZ2cnMIAgAAAAdDbGllbnRzCAAAAAAKRXZh
bHVhdGlvbgIAB0NvbW1lbnQLE1Byb2R1Y3Rpb24gbGljZW5zZS4=
```

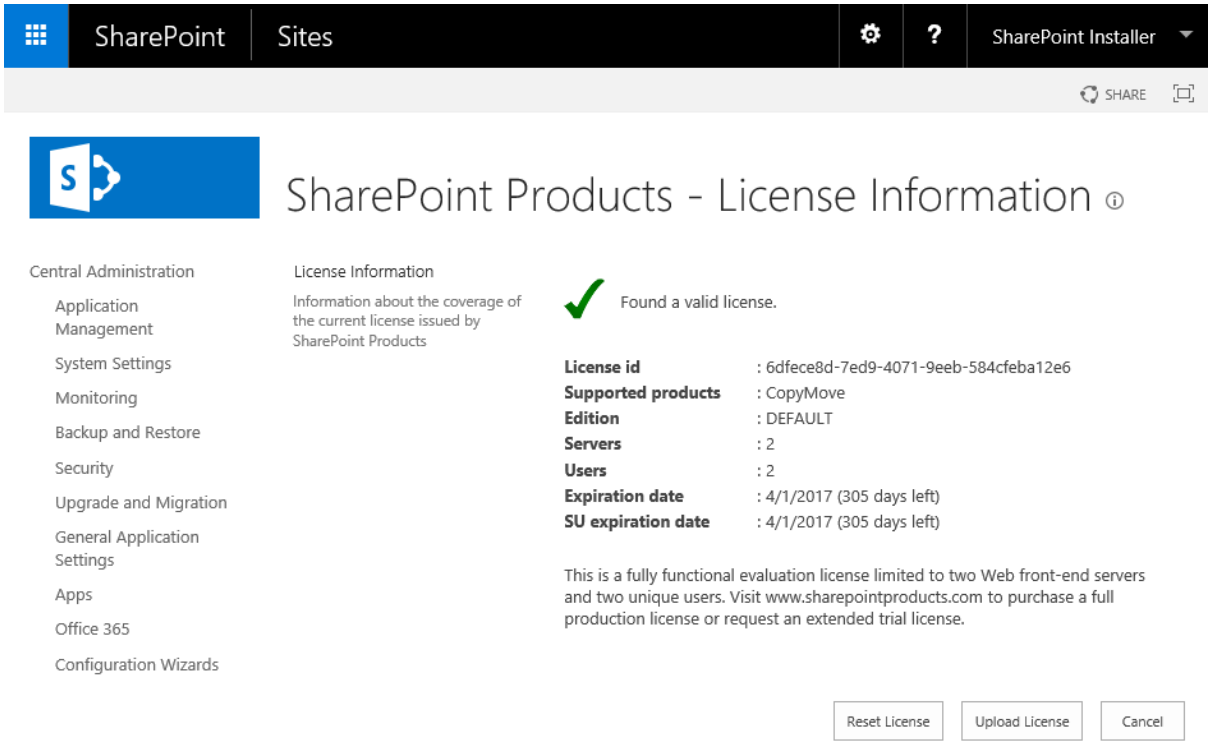
The license includes a number of properties in clear text for informational purposes. The same properties are also embedded in the encrypted section below the dashed line. CopyMove reads and verifies the properties from the encrypted section. Modifying the properties in clear text has no effect. Modifying the encrypted section will invalidate the license. The following table describes the purpose of the different license properties.

License Property	Description
Id	Globally Unique ID of the license.
Customer	Company name of the customer.
Contact	Name and email of the person who purchased the license.
Issuer	Name and email of the person who issued the license.
Products	List of products supported by the license.
Edition	Product edition supported by the license. Currently ignored by CopyMove.
Expiration Date	The date after which the license is no longer valid. Used for trial licenses and temporary production licenses. Permanent license will not have an expiration date.
SU Expiration Date	The date when the support and upgrade contract expires. New versions of CopyMove released after this date will not work with the license. But versions released within the support and upgrade contract will continue to work as long as the Expiration Date property above is not violated.
Servers	The number of Web Front-End servers supported by the license. CopyMove will fail with a license error if this number is exceeded.
Clients	The number of named users supported by the license. If this number is not

© 2008 - 2016 SharePoint Products

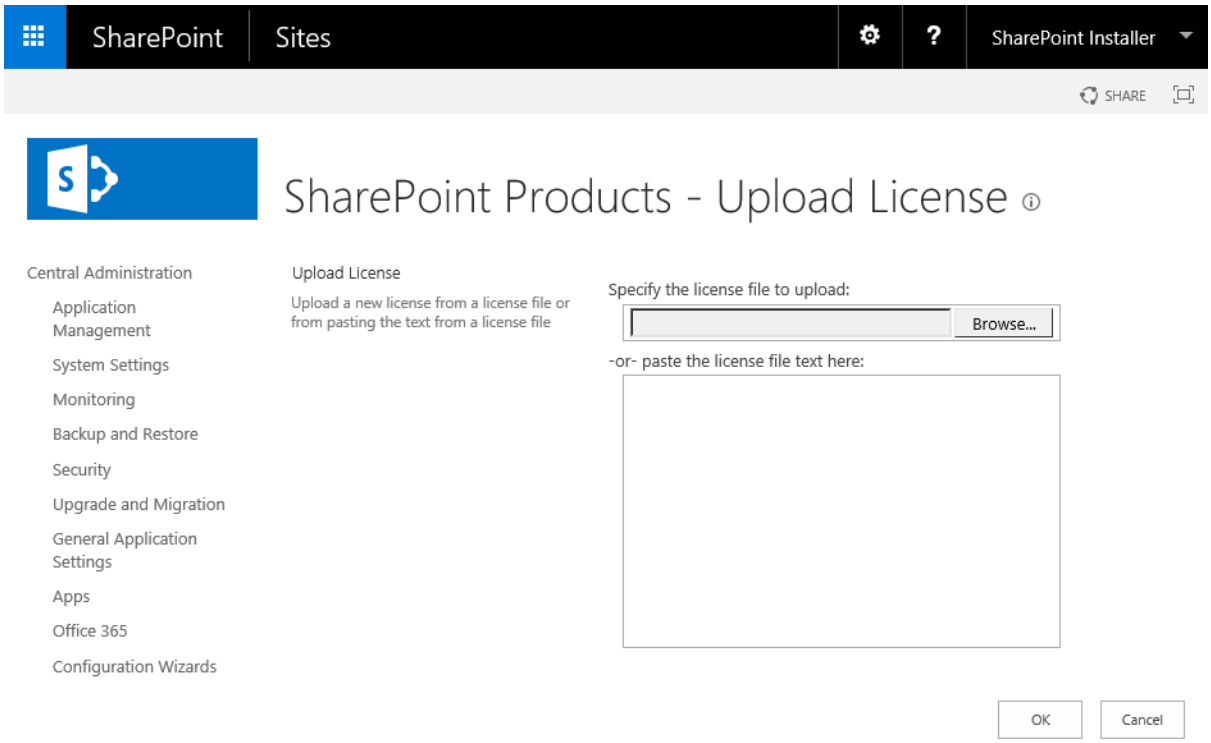
License Property	Description
	unlimited then CopyMove tracks the login name of each unique user that accesses CopyMove. Users that exceed the limit will receive a license error.
Evaluation	Specifies if the license is for evaluation purposes or not.
Comment	Additional license information shown on the *License Information* page shown in the screen shot below.

The status of the current license installed can be viewed by clicking the *Manage license* link in the *SharePoint Products* group on the SharePoint Central Administration site. Clicking this link opens the *License Information* page shown below.

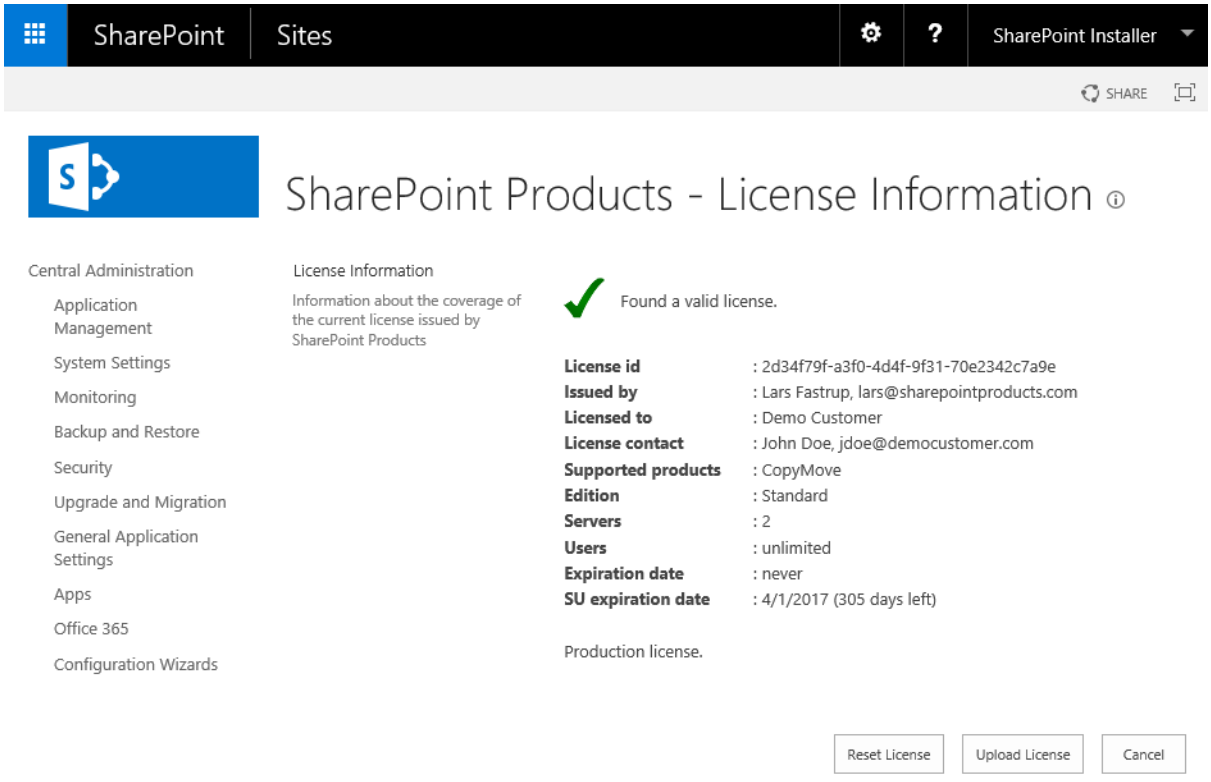


Use the *Reset License* button to revert to using the built-in license in CopyMove. Click the *Upload License* button to install a new license from the *Upload License* page shown below.

© 2008 - 2016 SharePoint Products

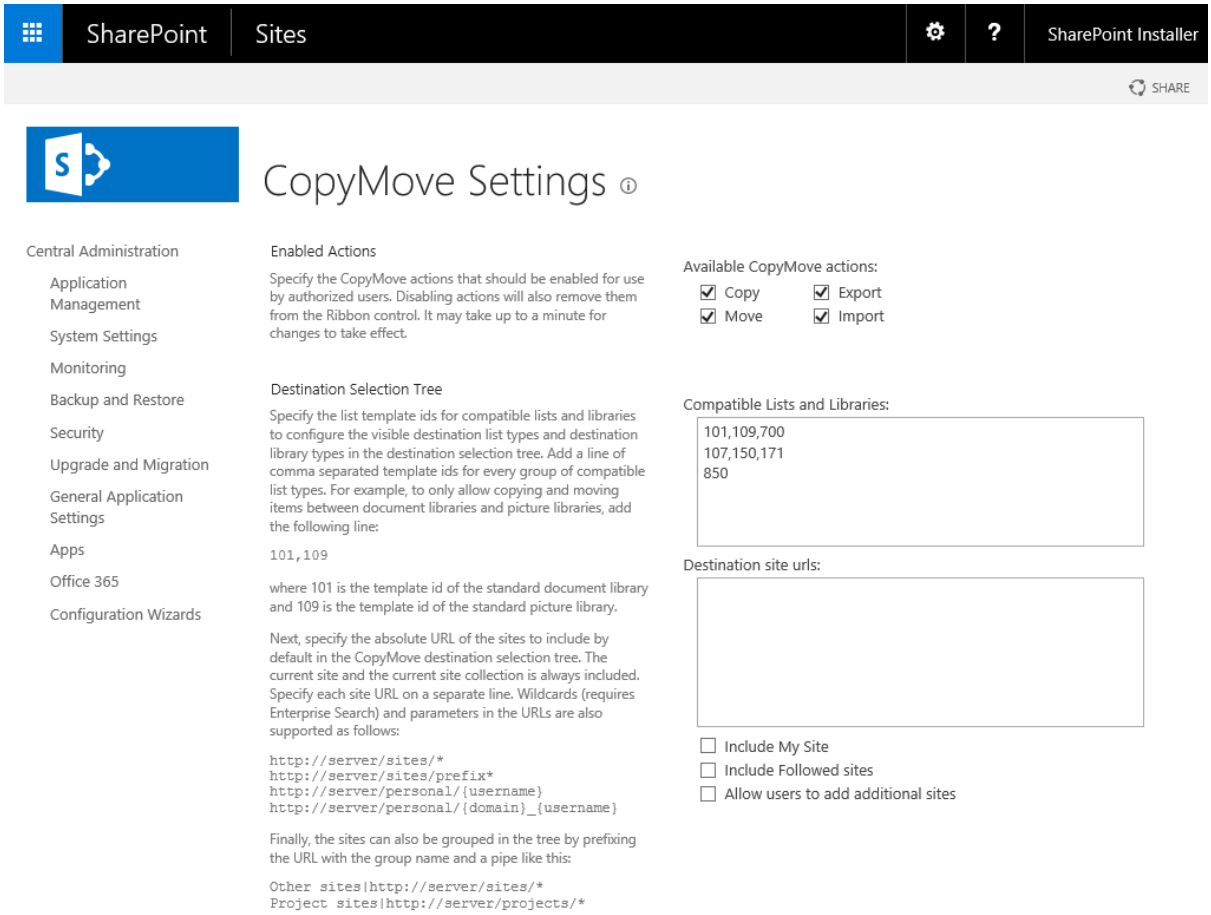


Specify the path to the license file to upload to the SharePoint server or simply copy and paste the contents of the license file into the large text box. Click the *OK* button to validate and install the license. Note: The sample license file shown earlier has been invalidated and will not upload without errors. A valid production license will look as follows on the *License Information* page.

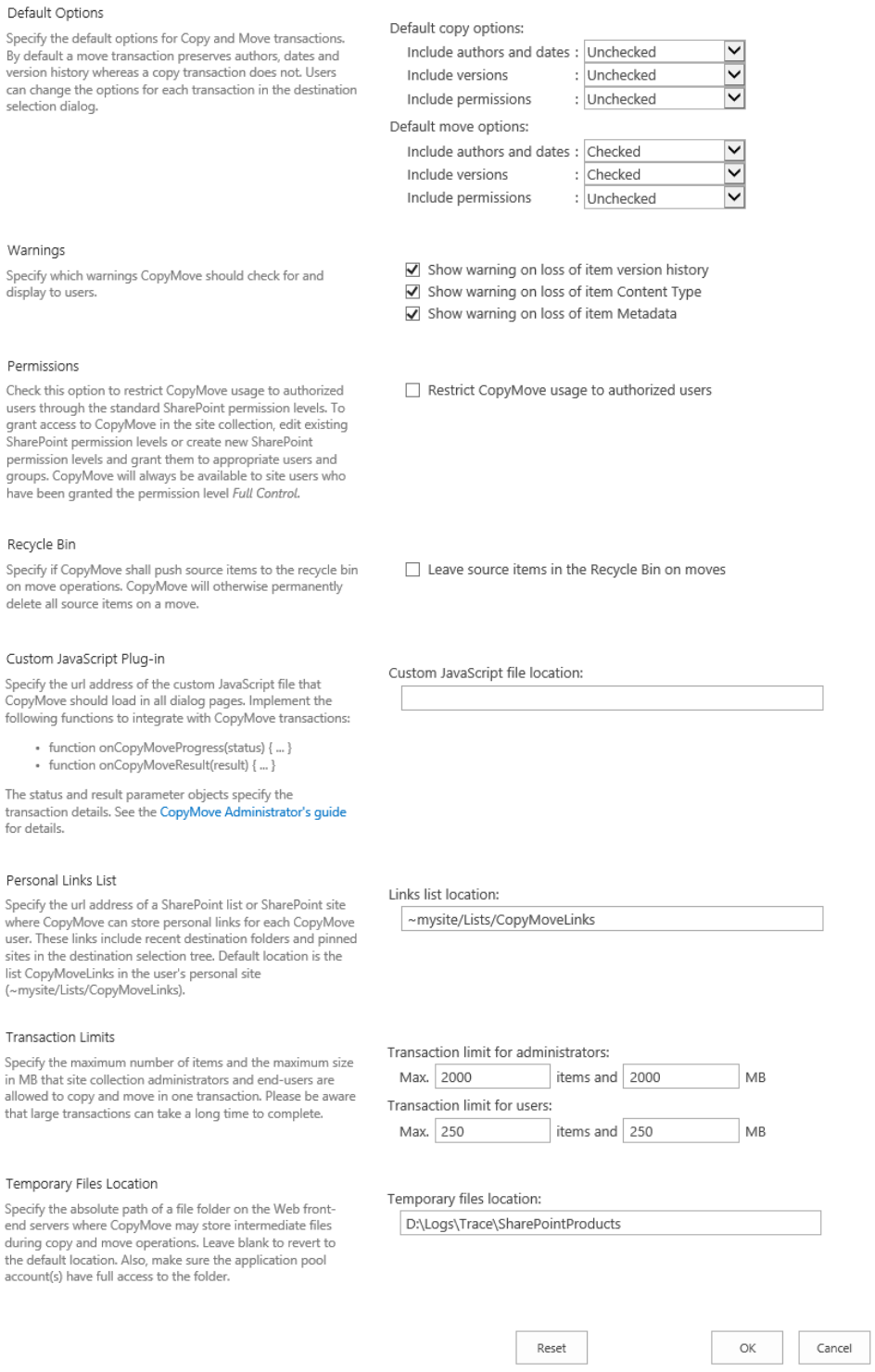


3.2 Global Settings

The CopyMove Settings page shown below is accessible from the SharePoint Central administration site when the CopyMove farm feature is activated. The page surfaces the global configuration settings available in CopyMove. The settings will apply throughout the farm where CopyMove is activated and available to users.



© 2008 - 2016 SharePoint Products



The following sub sections document the effect of the options available in each configuration section on the settings page.

Enabled Actions

Uncheck these check boxes to remove the corresponding buttons from the SharePoint user interface.

© 2008 - 2016 SharePoint Products

For example, uncheck the *Export* and *Import* check boxes to hide the corresponding Export and Import buttons from all users. To show the buttons for selected users only, enable CopyMove permissions in the Permissions section and grant permissions accordingly in each site collection.

Destination Selection Tree - Compatible Lists and Libraries
The destination selection tree in the CopyMove destination selection dialog shown above does by default only show compatible lists and libraries. For example if you select to copy a document in a standard SharePoint document library then the tree will only suggest other standard SharePoint document libraries as a possible destination location. It will not show picture libraries, page libraries or other libraries created from a different list template. The same goes for lists like a task list; if you select to copy a task item the tree will only suggest other task lists as a possible destination location. It does not make much sense to copy a task item to a links list or a document library.

However, there are scenarios where it is required to let CopyMove work across list templates. One scenario could be where you have created one or more custom document libraries from custom list templates. By default CopyMove will not think of these document libraries to be compatible. If they are compatible just specify the list templates ids separated by comma in the text box labeled *Compatible Lists and Libraries*. Use one line for each group of compatible lists and libraries. For example, to allow copying and moving items from picture libraries to document libraries and vice versa simply specify the line *101,109* where *101* represents the document library template id and *109* represents the picture library template id.

Destination Selection Tree - Destination site urls
The text box labeled *Destination site urls* is linked to the destination selection tree in the CopyMove destination selection dialog seen by end-users. The tree will by default only show the context site collection, the users My Site and Followed Sites but not other site collections that may exist in the farm. This is by design to avoid performance problems when loading the tree, as a farm can potentially contain thousands of site collections. CopyMove does instead allow administrators and end-users to specify relevant site collections and sub-sites for the tree.

Administrators can use the *Destination site urls* text box to specify one or more site collections or sub-sites in a site collection that should always show in the destination tree for all users. One example is the site collection(s) that make up the corporate intranet to make it easy for users to copy and move content from other site collections to the corporate intranet site collection(s). End-users can also add sites to the tree by specifying the absolute site URL. They can do so from the destination dialog itself shown in the screen shot below. They simply have to type or copy & paste the site url into the *Destination* text box and click the *add* link. Sites added by end-users are persisted in the personal links list and can be removed from the tree again by selecting the *Remove from tree* action in the right-click context menu available on the tree node. Sites specified by administrators on the CopyMove settings page cannot be removed from the tree by end-users. Administrators can also prevent users from adding sites to tree by unchecking the check box labeled *Allow users to specify additional sites*

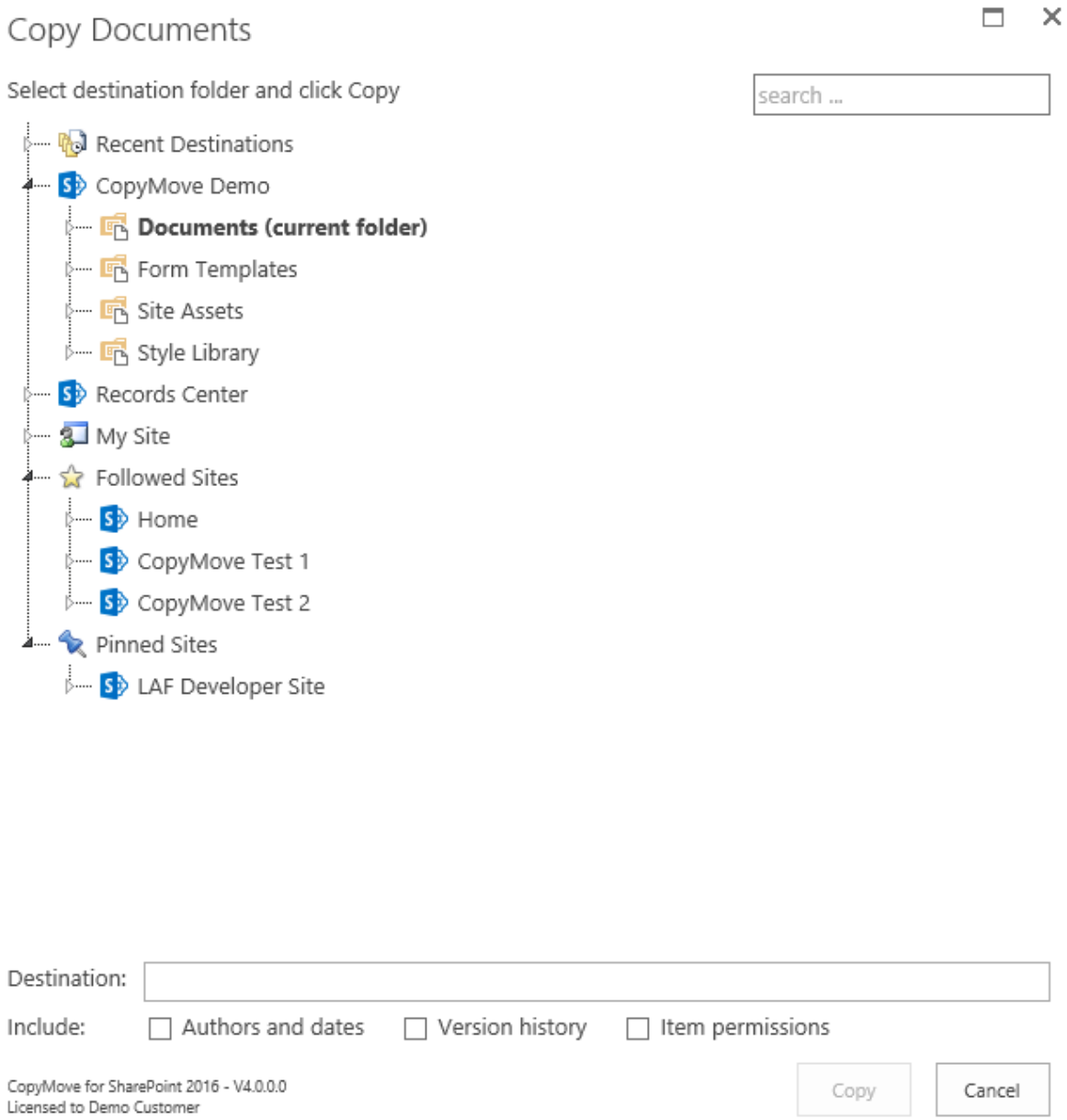
Finally, administrators can also specify if the destination tree should show the personal site (*Include My Site* check box) and followed sites (*Include Followed sites* check box) for the current user. The CopyMove destination selection dialog shown below illustrates the destination selection tree with nine site collections:

1. **CopyMove Demo** - the context site collection.
2. **Records Center** - a specific site collection configured by an administrator on the CopyMove settings

© 2008 - 2016 SharePoint Products

page.

3. **My Site** - the personal site of the current user.
4. **Followed Sites** - site collections that the current user follows.
5. **LAF Developer Site** - a site collection added and pinned by the current user.



Default Options

The CopyMove destination selection page illustrated in the above screen shot, include the three checkable options *Include authors and dates*, *Include versions* and *Include permissions*. The default state (checked/unchecked) of the check boxes can be configured independently for copy operations and move operations by the corresponding drop-downs in the *Default Options* section on the CopyMove settings page. They are by default set to unchecked for copy operations and checked for move

© 2008 - 2016 SharePoint Products

operations.

Warnings
CopyMove is always validating all transactions to detect errors and warnings before executing any content changes. Errors will cancel the transaction while warnings will only pause it and ask the user for approval to continue despite the warnings. Administrators have the option to turn off the following warnings:

- **Warning on loss of version history**. Users receive a warning when trying to copy, move or import items from a list or library with version history enabled to another list or library where version history is not enabled.
- **Warning on loss of content type**. Users receive a warning when trying to copy, move or import items associated with a Content Type that does not exist in the destination list or library.
- **Warning on loss of Metadata**. Users receive a warning when trying to copy, move or import items associated with a list columns that do not exist in the destination list or library.

Permissions
The CopyMove actions copy, move, export and import in the SharePoint list ribbon are by default visible and accessible for all users with read access to the list. To restrict access to selected users or selected groups, check *Restrict CopyMove usage to authorized users* and save the change by clicking the *OK* button. Hereafter, CopyMove is only available to users with Full Control and users who have been granted access. See the section Permissions Management for detailed information on configuring CopyMove permissions for users and groups.

Recycle Bin
The check box in this configuration section controls whether CopyMove should delete or recycle source items in move transactions. The default and recommended value is recycle, i.e. the check box is checked. Recycled source items can be recovered from the Recycle Bin within a window of 30 days whereas deleted items cannot be recovered again. Deleted items can only be recovered from a SharePoint backup. Now, when would you need to change this setting to delete then? The setting is provided to support scenarios where documents are assigned a unique ID. Here it is not desirable to have two documents with the same ID, which would be the case if you move a document and leave a copy in the recycle bin that is later restored by another user.

Custom JavaScript file location
Specifes the url address of a custom JavaScript file that CopyMove should load in all its dialog pages. Implement the following functions to hook into CopyMove transactions:

- function onCopyMoveProgress(status) { ... }
- function onCopyMoveResult(result) { ... }

The status and result parameter objects specify the transaction details. The objects have the properties of the CopyMoveStatus class and CopyMoveResult class respectively. See the API documentation for details.

Personal Links List
Specifies the url address of a SharePoint list or SharePoint site where CopyMove should store personal links for each CopyMove user. Personal links include recent destination folders and pinned sites in the destination selection tree. Default location is the user's personal site at ~mysite/Lists/CopyMoveLinks. The list is created on-demand when the user copies or moves items for the first time. See the Personal Links Location section for more details on changing the location of the list.

© 2008 - 2016 SharePoint Products

Transaction Limits
CopyMove does for performance reasons limit transactions by the number of files and by the total size of files. The default setting is a maximum of 250 items and 250 MB for normal users and 2000 items and 2000 MB for site collection administrators. Farm administrators can increase these limits to 2000 items and 2000 MB for normal users and 10000 items and 10000 MB for site collection administrators. But no higher as these are upper hard limits enforced by CopyMove. The transaction limits prevents users from (accidentally) creating long running and resource intensive transactions, which could otherwise happen by copying or moving a folder containing thousands of large documents. CopyMove should not be confused with a migration tool - it is primarily a tool for end-users who need to copy or move a few documents and folders at a time.

Temporary Files Location
CopyMove needs a temporary location on the Windows server file system for storing documents during transactions. This is because of the way it works internally, which is that it first exports all selected content from the source location and then it imports it again to the selected destination location. The exported content is stored on disk where it is in turn imported from again. The content files are deleted again as soon as the transaction completes or fails. The default location for the content files is the following folder on each Web Front-End server(s) in the farm:

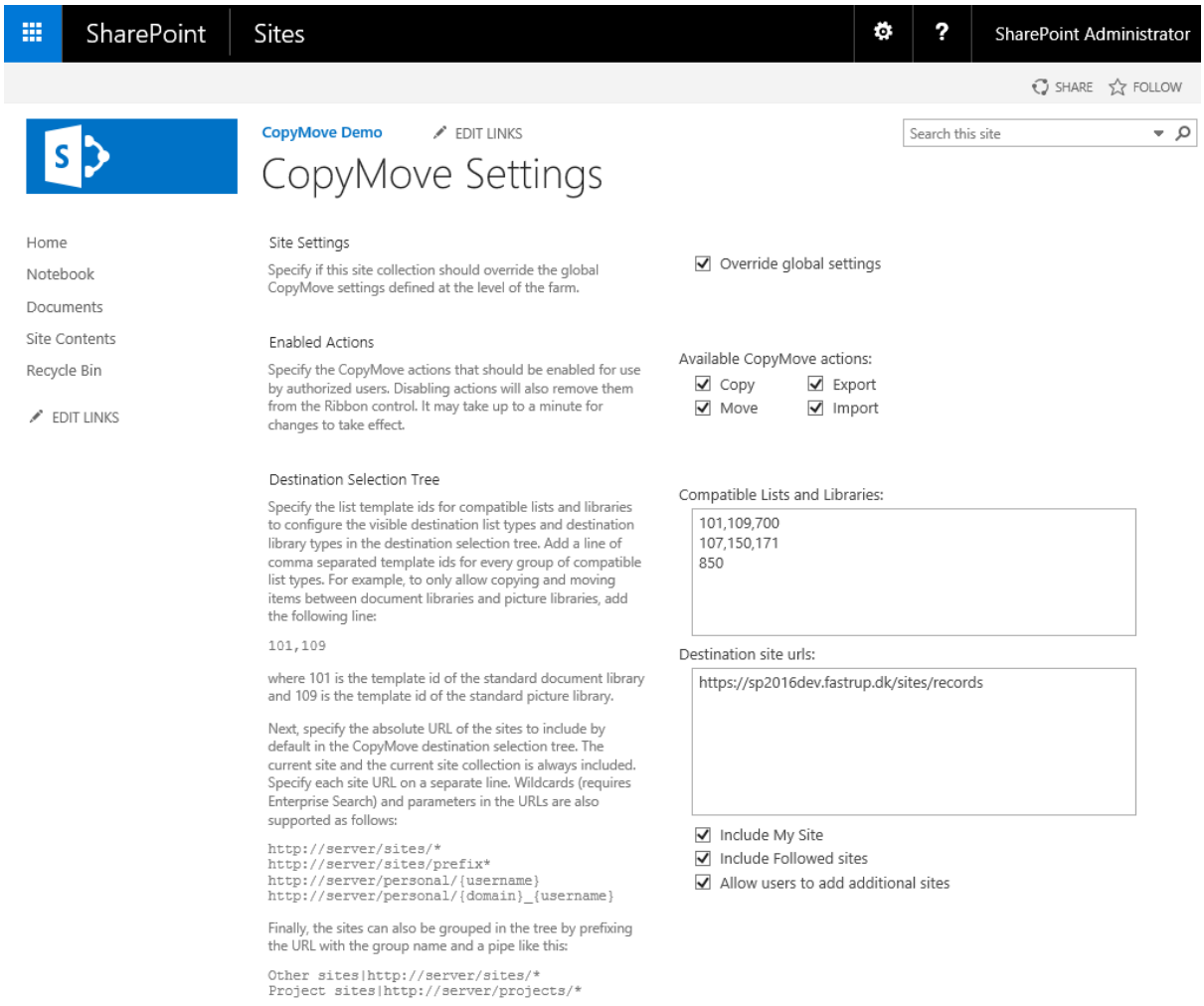
```
C:\Program Files\Common Files\Microsoft Shared\Web Server Extensions\15
\LOGS\SharePointProducts
```

The LOGS folder is ideal for CopyMove as the IIS application pool account(s) of the SharePoint Web applications has full access to this folder. Changing the temporary files location should be followed with a check to ensure that the application pool account(s) also have full access there.

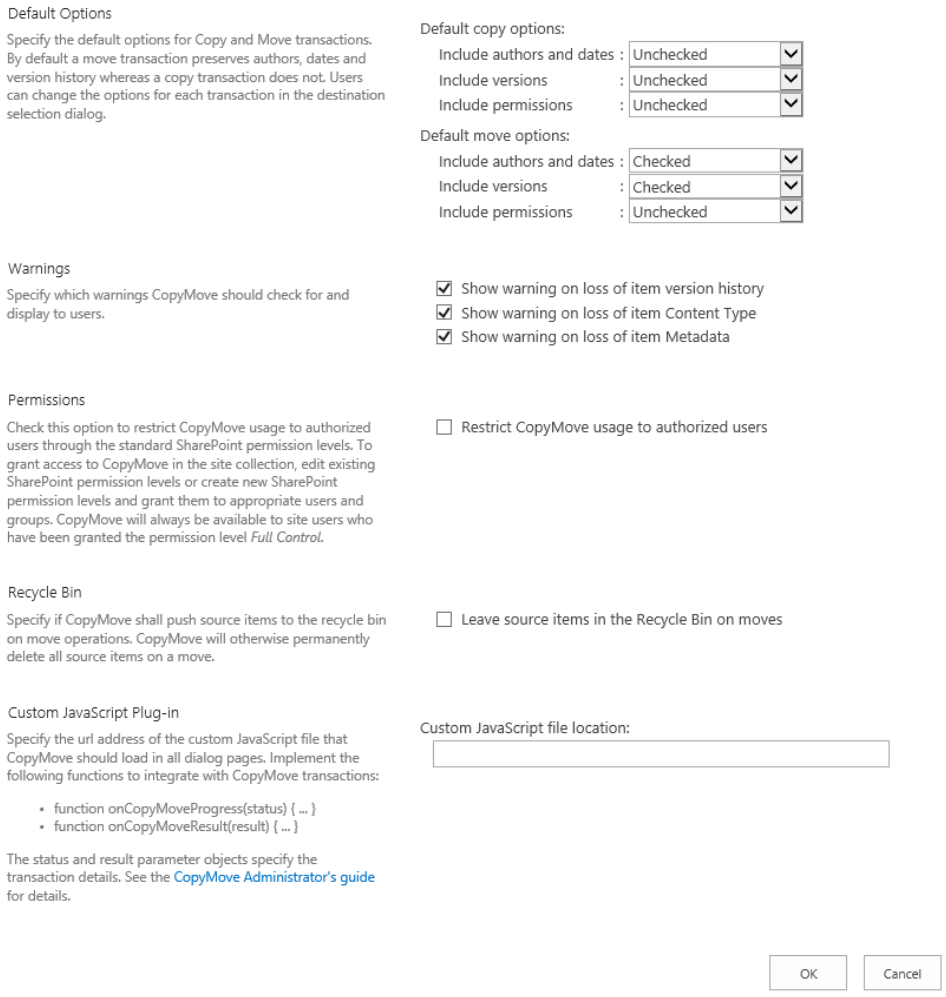
© 2008 - 2016 SharePoint Products

3.3 Site Collection Settings

The CopyMove Settings page shown below is available to site collection administrators in site collections where the CopyMove site collection feature is activated. The page is accessible through a link on the site settings page as shown in the Administration section. Site collection administrators can leverage the page to override the global settings for the context site collection. However, the transaction limits and the temporary files location cannot be overridden. These settings are only available to the farm administrator. The default behavior of all site collections is to inherit the global settings. To override the settings, simply check the *Override global settings* check box and all available options become enabled as illustrated by the following screen shot.



© 2008 - 2016 SharePoint Products



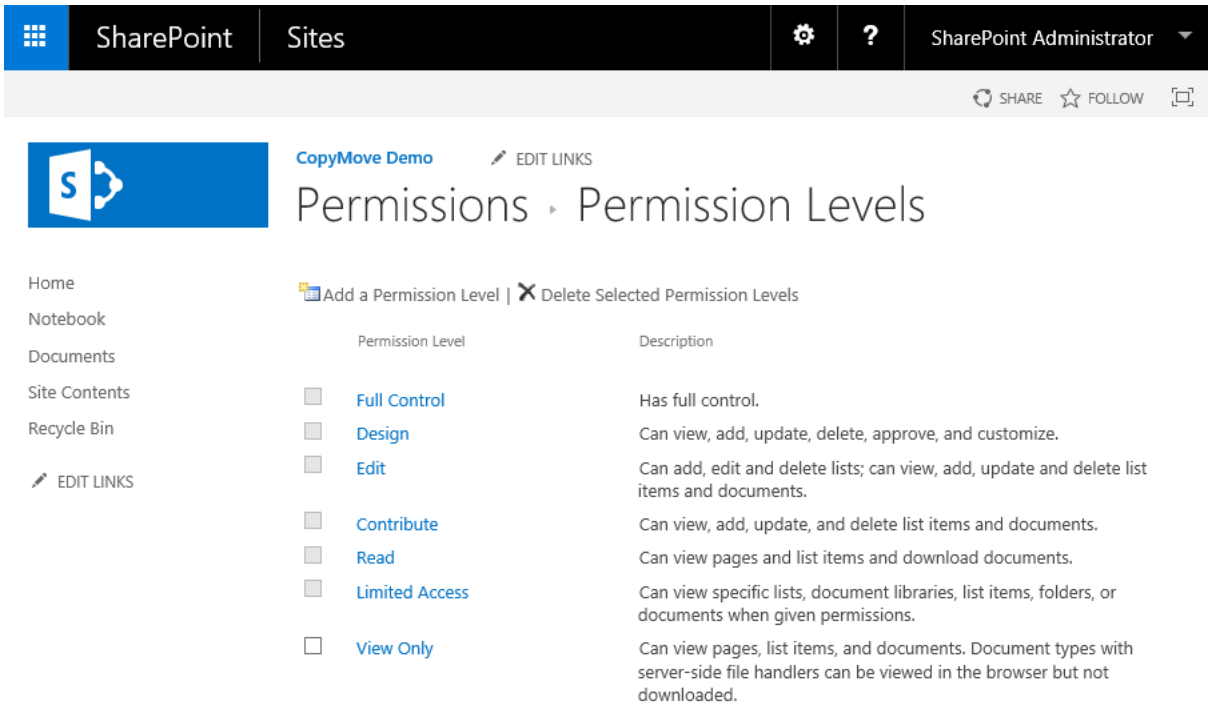
All the options on the page have identical effects as the options on the global settings page described in the Global Settings section. The only difference is that the settings are limited to the site collection.

© 2008 - 2016 SharePoint Products

3.4 Permissions Management

Site collection administrators, site owners and other users who can manage permissions in SharePoint also have the power to limit access to the CopyMove functionality. However, not before CopyMove has been configured to use permissions as described in the sections Global Settings and Site Collection Settings. Otherwise the CopyMove buttons are available to everyone with access to SharePoint sites where the CopyMove site collection feature is activated. CopyMove will of course always respect regular SharePoint permissions in terms of sufficient privileges to view, add and delete items in lists.

Users and groups with the *Full Control* permission level will always have access to CopyMove in site collections where the CopyMove feature is activated. But other users and groups will need to be granted permission to the appropriate CopyMove actions through SharePoint permission levels. The following screen shot shows the available permission levels in a default SharePoint 2016 team site.

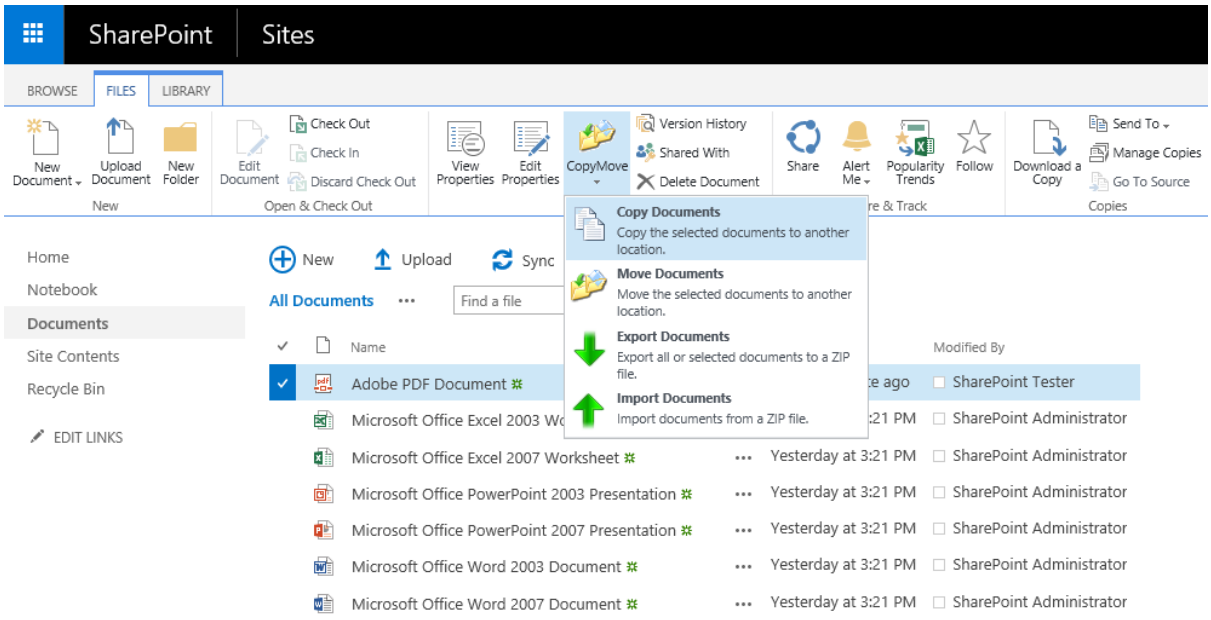


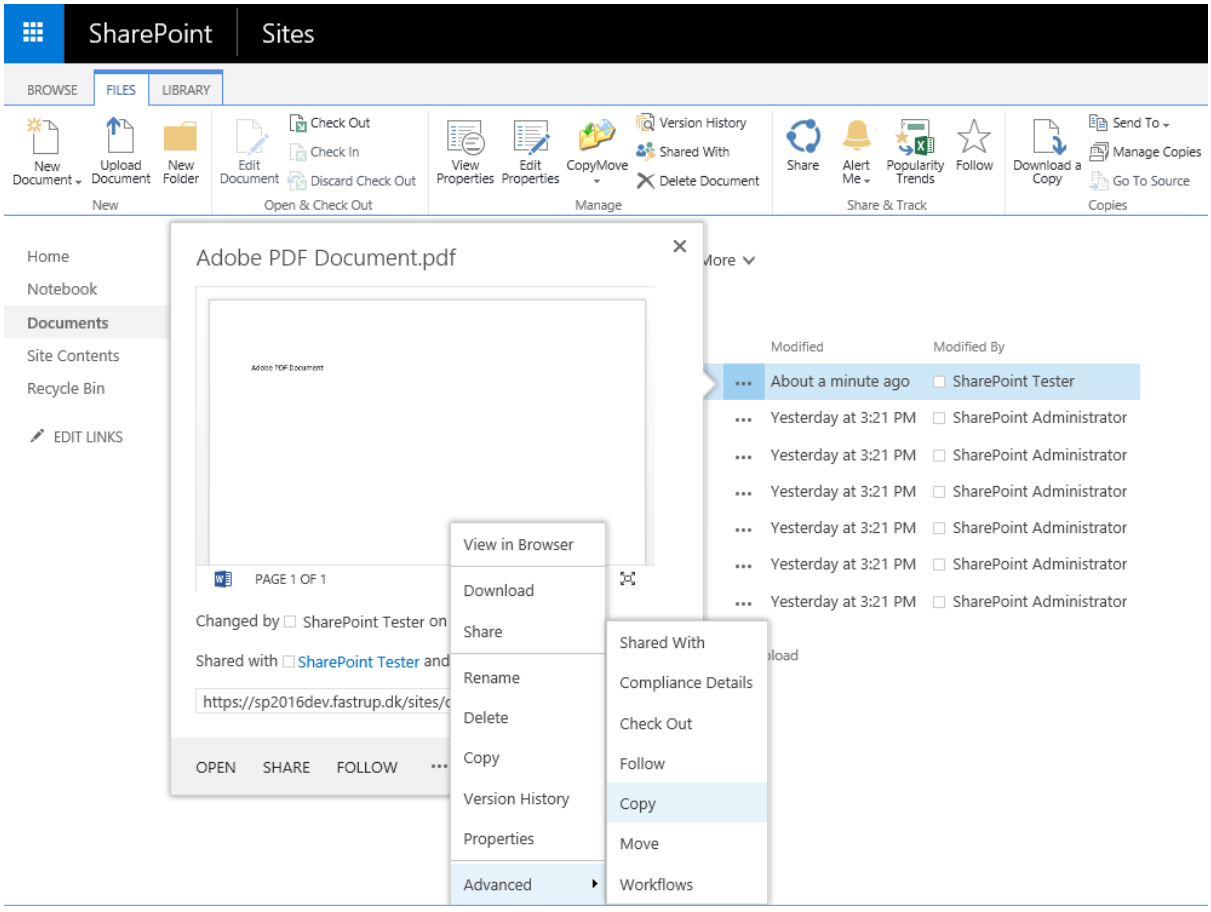
There is basically two ways of granting user permissions to CopyMove:

1. Add a new permission level for CopyMove and grant it to all appropriate users and groups.
2. Edit an existing permission level like *Contribute* and check the CopyMove permissions that apply to the level. Existing users who have already been granted the *Contribute* permission level, will in turn also get access to using CopyMove.

The permission levels will in turn trim the CopyMove buttons in the ribbon shown below in the first screen shot. The second screen shot show the SharePoint item context menu that CopyMove also integrates with. It is also trimmed according to the SharePoint permission level(s).

© 2008 - 2016 SharePoint Products

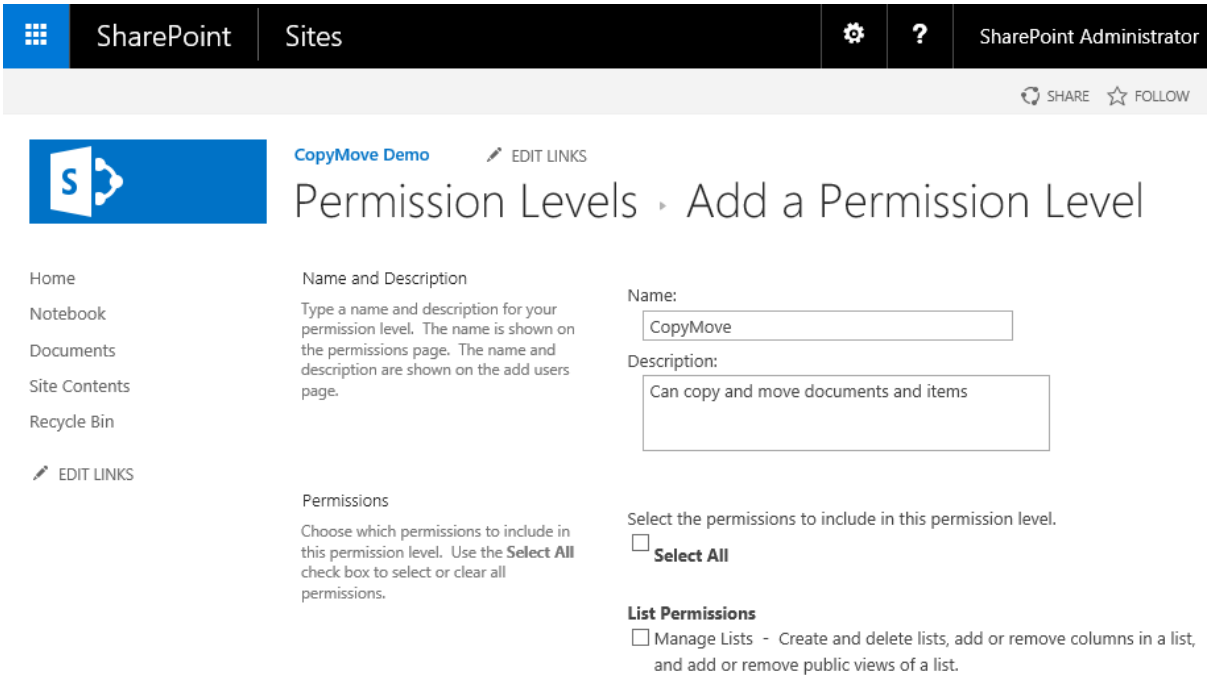




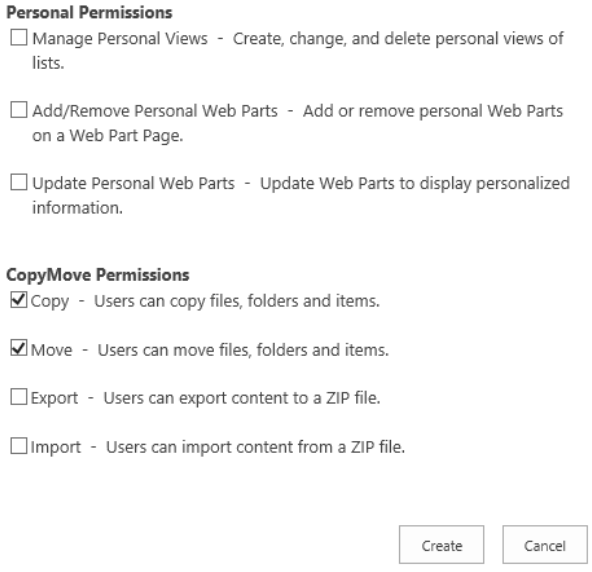
© 2008 - 2016 SharePoint Products

3.4.1 Granting access with a new CopyMove permission level

1. On the *Permission Levels* page shown above, click the link *Add a Permission Level*.
2. Next on the *Add a Permission Level* page, specify a name and a description for the new permission level - for example:



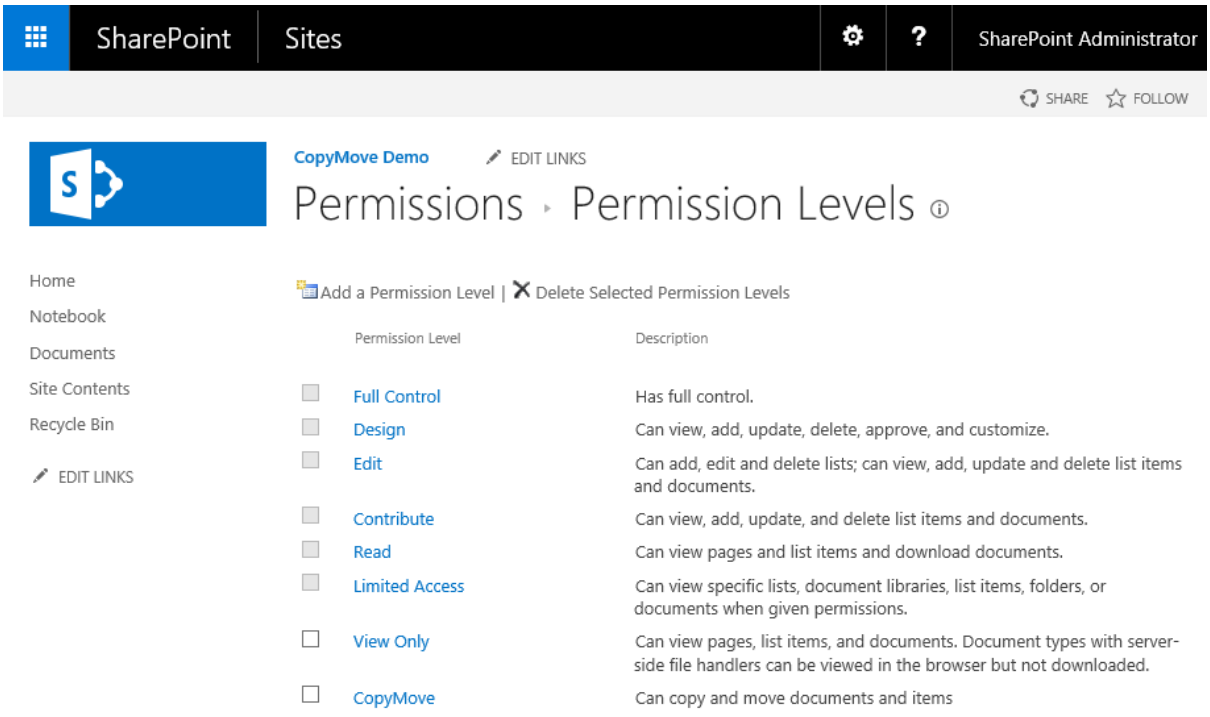
3. Then scroll down to the bottom of the page and specify the appropriate CopyMove permissions for the new permission level. The following screen shot shows an example of a new permission level that will include access to the *Copy* and *Move* functions but not the *Export* and *Import* functions.



© 2008 - 2016 SharePoint Products

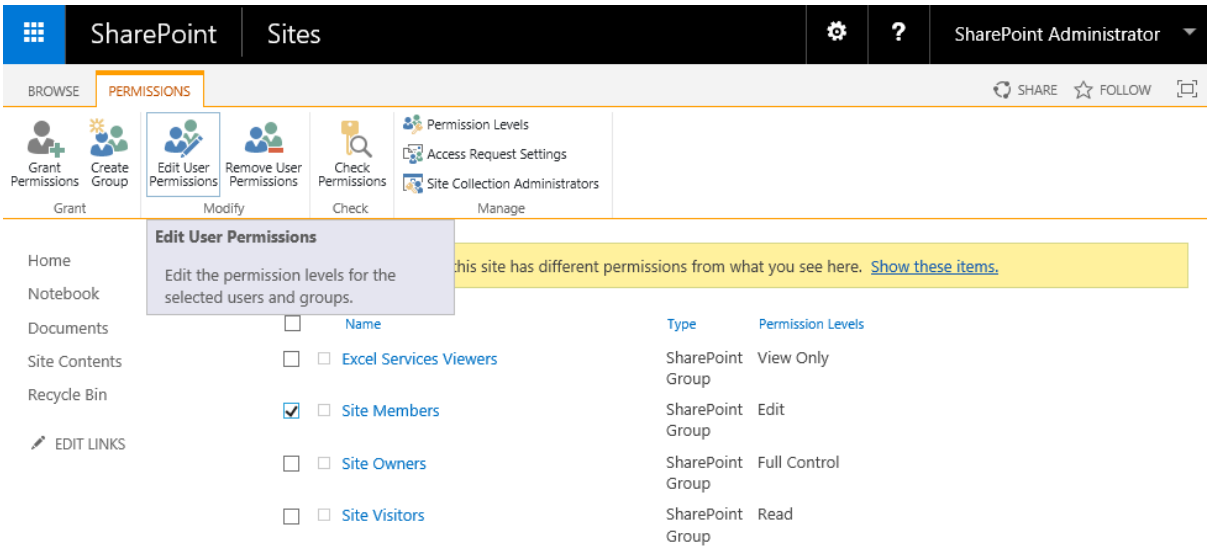
NOTE: If the *CopyMove Permissions* section is not shown then verify that the CopyMove site collection feature is activated and that the check box *Restrict CopyMove usage to authorized users* is checked on the CopyMove Site Collection Settings page.

4. Click the *Create* button to create the new CopyMove permission level. The list of permission levels now like this:

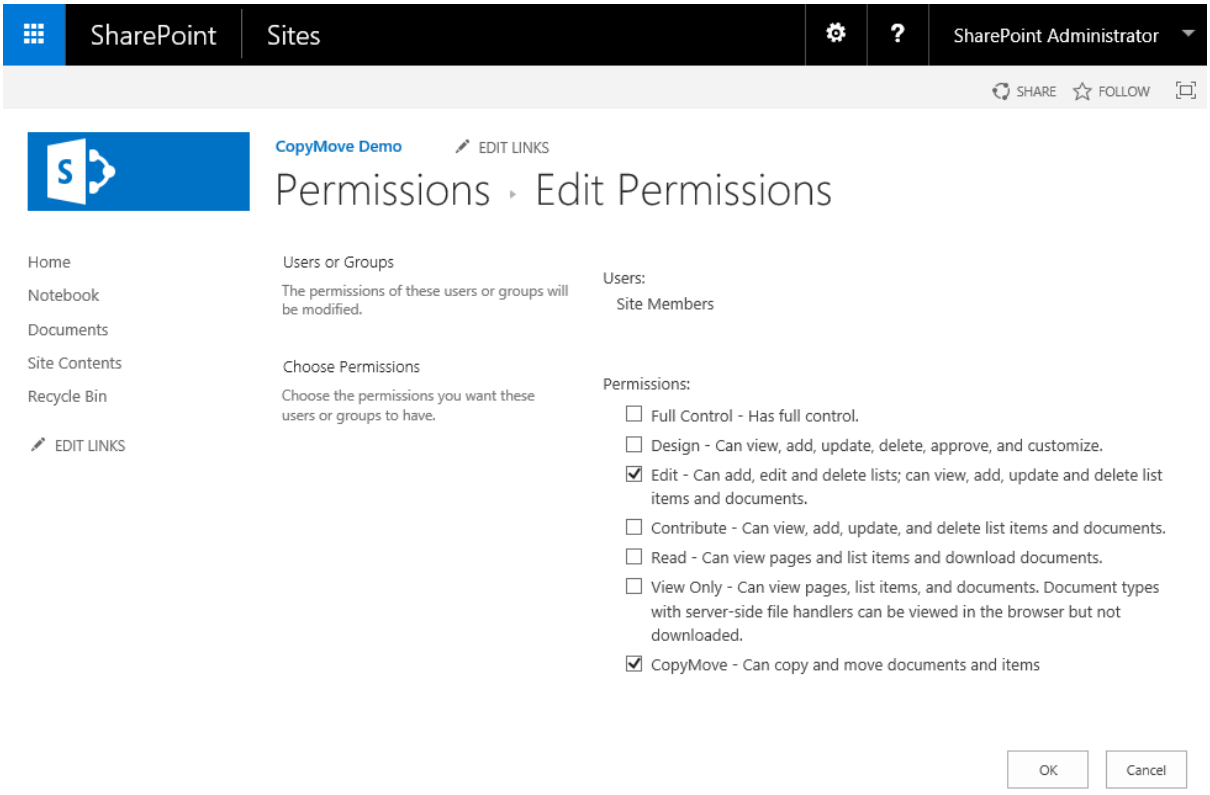


5. To grant permissions, manage permissions as usual in SharePoint and select the users or groups for which to change the permissions as shown in the screen shot below.

© 2008 - 2016 SharePoint Products



6. Click the *Edit User Permissions* button to edit the permissions for the selected users and groups. This opens the standard SharePoint permissions page as shown in this screen shot:



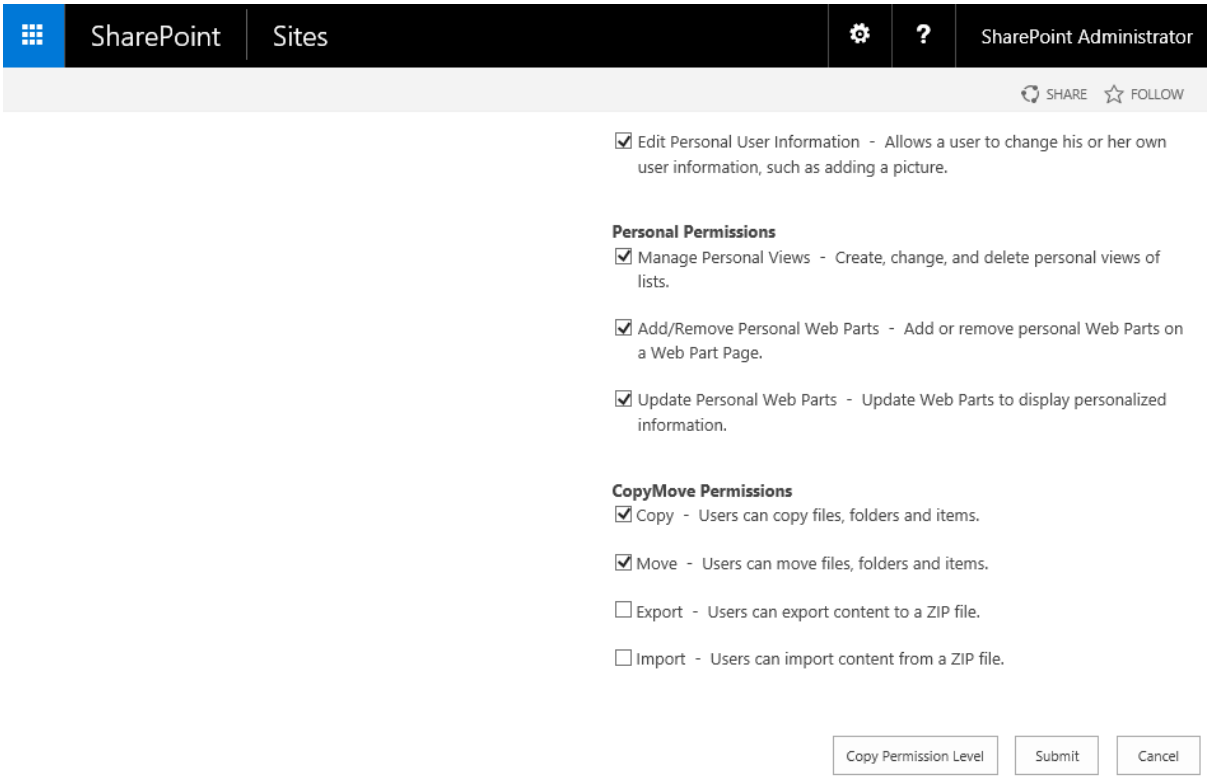
7. Check the appropriate permissions and click the *OK* button to apply the change.

8. Done.

© 2008 - 2016 SharePoint Products

3.4.2 Granting access through an existing permission level

1. On the *Permission Levels* page shown above, click the *Permission Level* to use for CopyMove access.
2. Scroll down to the bottom of the page and specify the appropriate CopyMove permissions for the new permission level. The following example will include access to the *Copy* and *Move* functions but not the *Export* and *Import* functions.



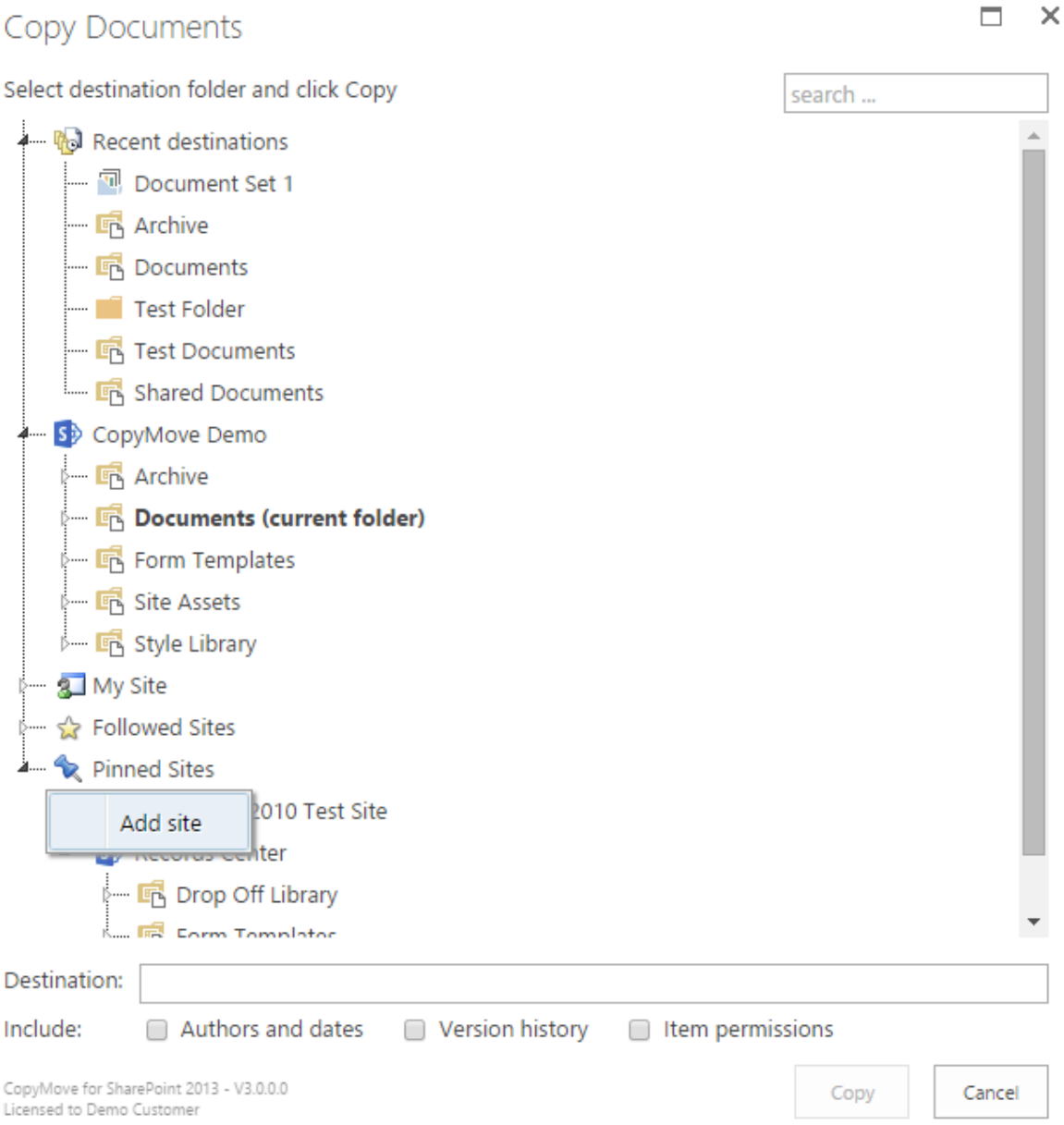
NOTE: If the *CopyMove Permissions* section is not shown then verify that the CopyMove site collection feature is activated and that the check box *Restrict CopyMove usage to authorized users* is checked on the CopyMove Site Collection Settings page.

3. Click the *Submit* button to update the permission level with the selected CopyMove permissions.
4. Done. Existing users and groups who have been granted the selected permission level will now have access to CopyMove as well.

© 2008 - 2016 SharePoint Products

3.5 Personal Links Location

The CopyMove destination selection dialog can track and show personal locations in the tree shown in the screen shot below. This includes list or folder links in the *Recent destinations* node and SharePoint sites in *Pinned Sites node*. The first shows recent destinations selected by the user for previous CopyMove transactions and the second shows SharePoint sites added by the user via the *Add site* action in the right-click context menu available on the *Pinned Sites* node.



From CopyMove version 3.8, personal links are stored in a SharePoint list. Earlier versions used a Web cookie to store the title and url for each entry. The default location of the new SharePoint list is:

```
~mysite/Lists/CopyMoveLinks
```

That is, CopyMove creates a list for each user in their personal site collection. The list is created on-

© 2008 - 2016 SharePoint Products

demand the first time the user copies or moves document(s). If the user does not have a personal site collection yet, or if My Sites has not been enabled for the SharePoint farm then CopyMove will fall back to using the current site collection. In that case all users will share the same list. They can in turn all browse the list but will only be able to see their own list items.

The default list location can be changed on the Global CopyMove settings page in SharePoint Central Administration or by the following PowerShell command from a SharePoint 2016 Management Shell:

```
Set-CopyMoveSettings -LinksListUrl <absolute url address of the
SharePoint list>
```

For example:

```
Set-CopyMoveSettings -LinksListUrl https://host/sites/copymove/lists/
links
```

Specifying one specific list this way will instruct CopyMove to only create this one list and store all personal links for all users here. The list is created on-demand by the first user that initiates a copy or a move. That first user must have access to the specified SharePoint site. Subsequent users will not need any access as CopyMove will automatically configure the list with contribute access to all users. Again, users will be able to browse the list but they will only be able to see their own items. However, the list is hidden and can only be accessed by typing or pasting the url. The list and it's items will also not show up in search results.

Link items for *pinned sites* are stored in the list until the user decides to delete them via the *Remove from tree* action available with the right-click context menu. Link items that represent recent destinations, are deleted after 90 days by a CopyMove timer job.

3.6 Auditing of Copy and Move events

Organizations often have policies and regulations that require them to track what documents users are accessing and when. For this reason, CopyMove also integrates with the built-in auditing facilities in SharePoint 2016. When auditing is enabled for copy and move events then CopyMove will log the following information to the SharePoint audit log for each item that is copied or moved:

- The date and time of the event.
- The id of the user who triggered the event.
- The id and the location URL of the item that was copied or moved.
- The type of the event (Copy or Move)
- The id and the location URL of the destination folder where the item was copied or moved to.

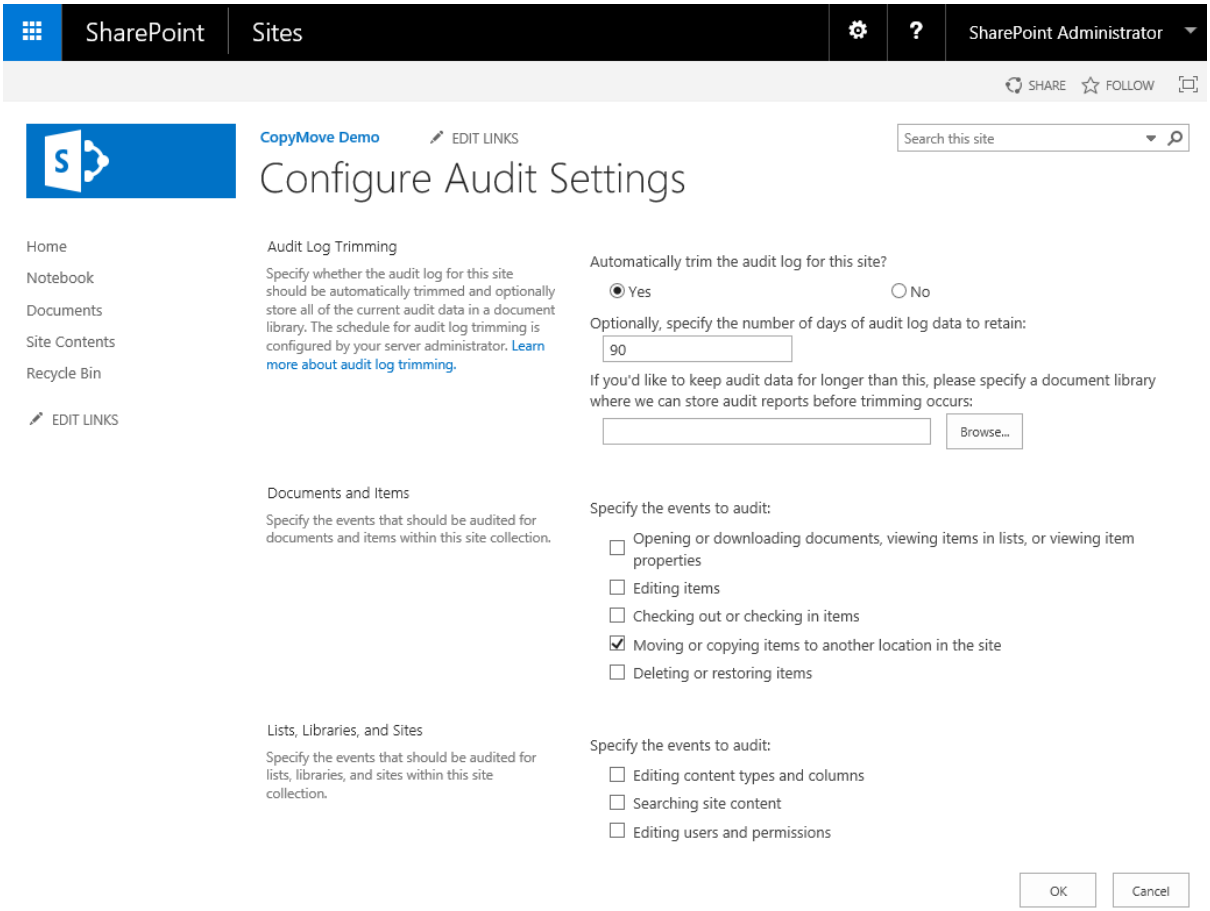
Again, note that this information is logged per item and not per transaction. To enable CopyMove auditing for a site collection, follow these steps:

1. Open the root site of the site collection for which to configure auditing.

© 2008 - 2016 SharePoint Products

2. Select *Site Settings* in the *Site Actions* menu.

3. On the *Site Settings* page, click the *Site collection audit settings* link in the *Site Collection Administration* group. This brings up the *Configure Audit Settings* page shown below.



4. Check the check box labeled *Moving or copying items to another location in the site* to include CopyMove events in the audit log.

5. Click the *OK* button to save the change.

Entries in the SharePoint audit log can in turn be viewed and analyzed using the built-in audit log report generator in SharePoint or any other tool capable of mining the SharePoint audit log. To generate a report for Copy and Move events in a site collection, follow the steps below:

1. Open the root site of the site collection for which to configure auditing.

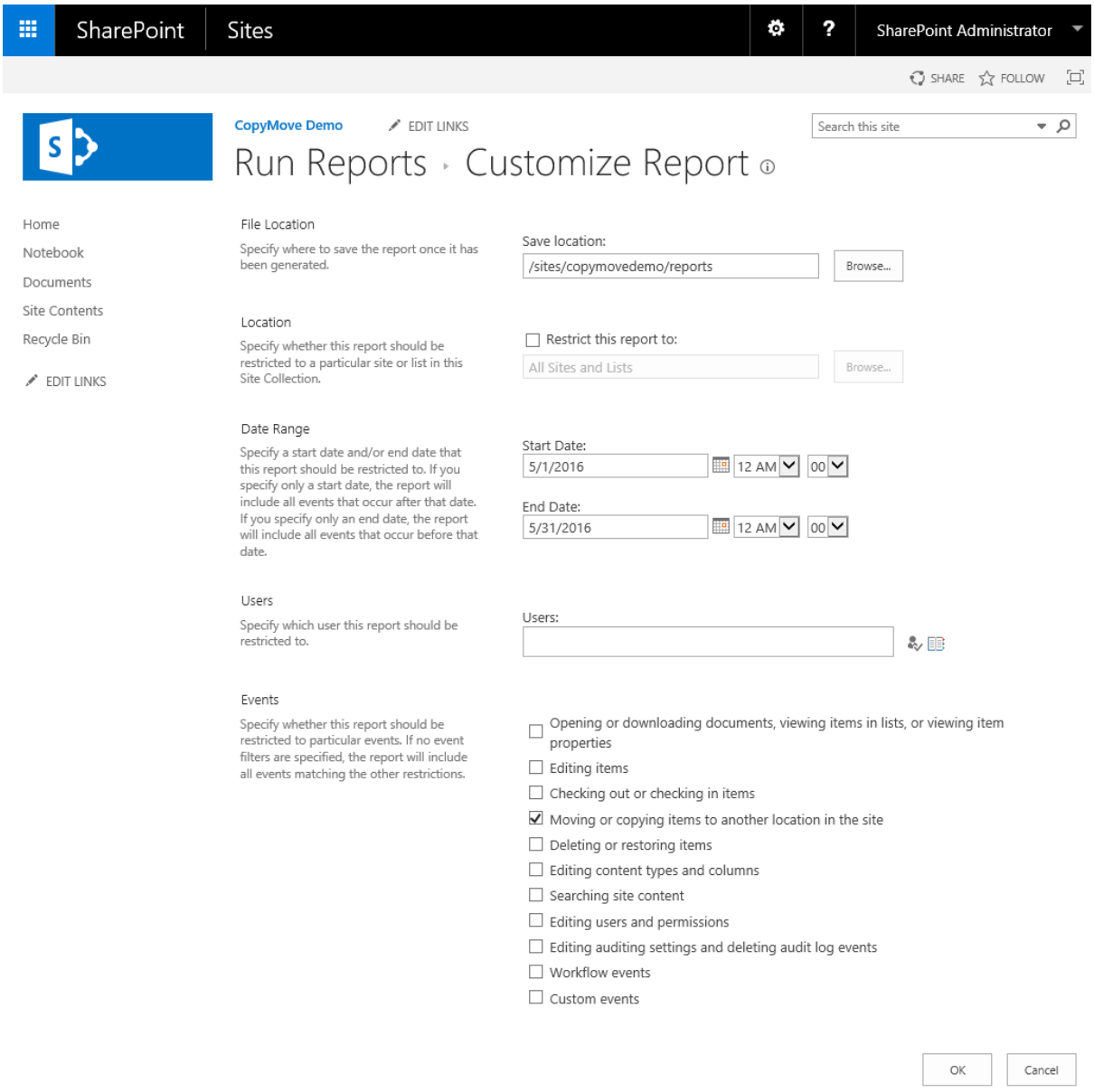
2. Select *Site Settings* in the *Site Actions* menu.

3. On the *Site Settings* page, click the *Audit log reports* link in the *Site Collection Administration* group.

4. On the *View Auditing Reports* page, select *Run a custom report*. This brings up the page shown

© 2008 - 2016 SharePoint Products

below.



5. Select a document library in the site collection where to save the report to.

6. Optionally, specify a data range.

7. Optionally, limit the report to one or more users.

8. Check the event labeled *Moving or copying items to another location in the site*.

9. Click the *OK* button to generate the report.

10. Open the resulting Excel document to view the result.

© 2008 - 2016 SharePoint Products

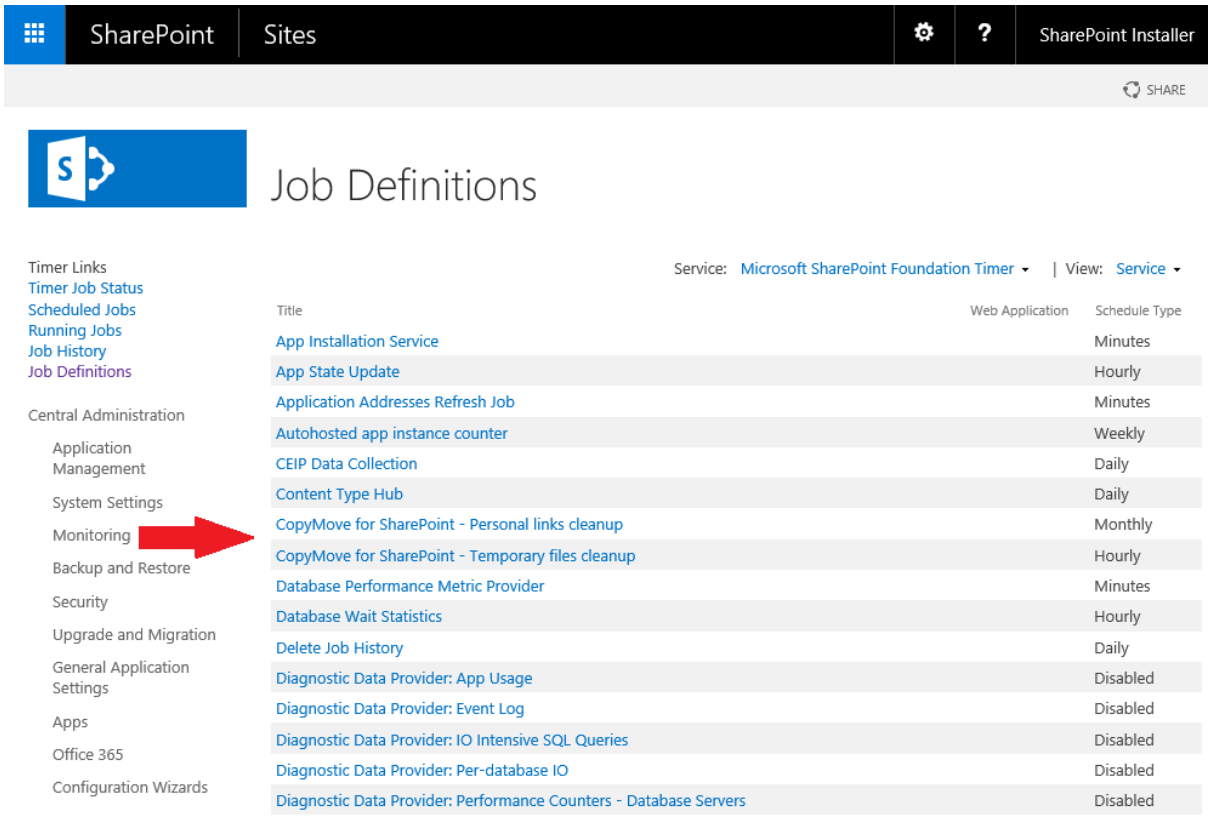
On a final note, make sure to trim the audit log in order to prevent it from growing into an unmanageable size. You can for instance limit the log to only retain events for the last 30 days as shown in the first screen shot in this section.

3.7 Timer Jobs

Activating the CopyMove farm feature installs these three timer jobs:

- **CopyMove for SharePoint - Upgrade site collections**. One-time job that upgrades the CopyMove site collection feature on all site collections where necessary. The job runs on the server where the farm feature was activated from.
- **CopyMove for SharePoint - Temporary files cleanup**. Hourly job that deletes temporary CopyMove files older than 12 hours. The job runs on all servers.
- **CopyMove for SharePoint - Personal links cleanup**. Monthly job that deletes expired destination links from all CopyMove Links lists in the farm. Destination links are expired when the last modified date is older than 90 days from today's date. The job runs on the server where the farm feature was activated from.

The CopyMove timer jobs can be viewed in the list of timer job definitions in SharePoint Central Administration as illustrated with the screen shot below.



© 2008 - 2016 SharePoint Products

3.8 Diagnostic Logging

CopyMove logs detailed information about every copy and move transaction triggered by the users. Errors and warnings are by default logged to the Windows Event log as well as to the SharePoint trace log stored in the following folder (with default configuration) on each server:

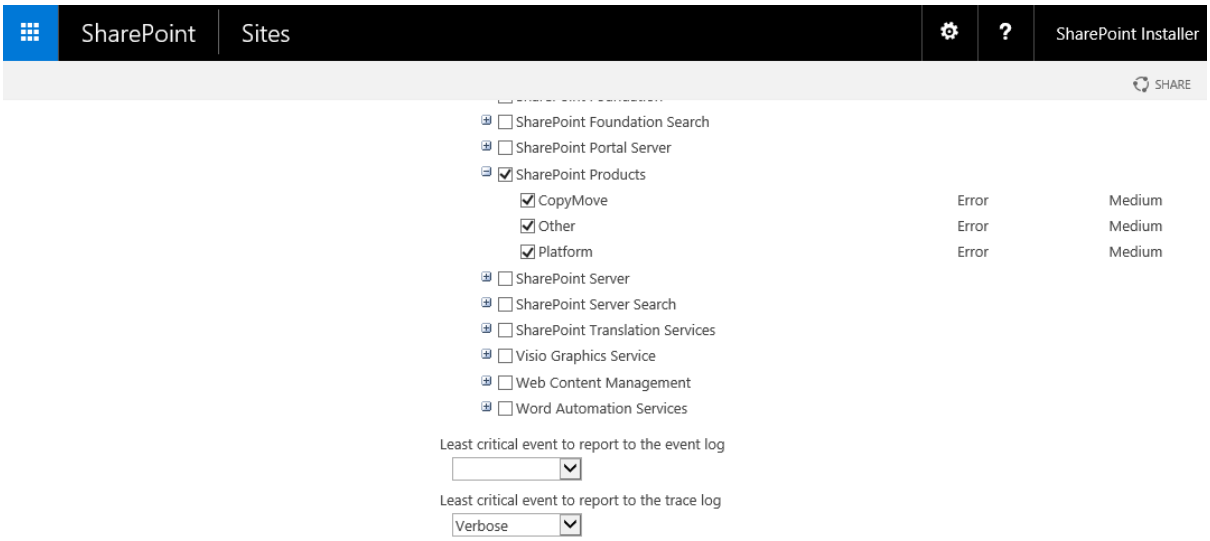
```
C:\Program Files\Common Files\Microsoft Shared\Web Server Extensions\15
\LOGS
```

But the detailed trace information is by default only logged to the SharePoint trace log in order not to flood the Windows Event log. Viewing the trace log files is possible with Notepad or any other text editor - but they are best examined with a dedicated tool like the free ULS Viewer available here:

http://archive.msdn.microsoft.com/ULSViewer

Just remember that each server in the farm has its own trace log file. They can, however, easily be merged into a single file using the PowerShell cmdlet `Merge-SPLogFile`. The CopyMove product download includes the PowerShell script `GetLogFile.ps1` that extracts all recent CopyMove events from all servers in the farm into a single file named *spp.log*. The script can also be launched using the Windows batch file `GetLogFile.cmd`. Recent events are defined as all the events produced within the past hour.

The detail level of the CopyMove diagnostics logging can be configured by SharePoint administrators from *Central Administration* > *Diagnostic Logging* as shown in the screen shot below.



Three diagnostic logging categories are registered for SharePoint Products; *CopyMove, Platform,* and *Other*. To enable verbose trace logging for CopyMove:

1. Select the categories *CopyMove* and *Platform* or simply select the parent category *SharePoint Products* to select all categories.

2. Select the *Verbose* option in the drop-down box labeled *Least critical event to report to the trace log*.

© 2008 - 2016 SharePoint Products

3. Click the *OK* button to save the change.

CopyMove will in turn start logging more details about each transaction.

© 2008 - 2016 SharePoint Products

Part IV

PowerShell Cmdlets

4 PowerShell Cmdlets

CopyMove 2016 includes a number of PowerShell Cmdlets to support CopyMove specific tasks from the PowerShell command-line or from PowerShell scripts. The following table provides an overview of the available cmdlets.

Cmdlets	Description
Get-SPPLicense	Retrieves the current license key in the farm
Set-SPPLicense	Applies a new license key to the farm
Get-CopyMoveSettings	Gets the CopyMove configuration settings
Set-CopyMoveSettings	Updates the CopyMove configuration settings
Remove-CopyMoveSettings	Removes all CopyMove settings from the specified site collection
Set-CopyMovePermissionLevel	Updates the CopyMove permissions for users in the specified site collection or sub-site
Copy-SPPItem	Copies documents, list items and folders
Move-SPPItem	Moves documents, list items and folders
Export-SPPItem	Exports documents, list items and folders from a SharePoint list/library to a ZIP file
Export-SPPList	Exports all documents, list items and folders from a SharePoint list/library to a ZIP file
Import-SPPItem	Imports documents, list items and folders from a ZIP file to a SharePoint list/ library

The cmdlets are registered with SharePoint on all servers when CopyMove is installed. Hereafter, they are immediately available for use within the SharePoint 2016 Management Shell. PowerShell scripts that are not executed from the Management Shell must first add the PowerShell Snapin Microsoft. SharePoint.PowerShell. The following PowerShell snippet loads it if necessary.

```
 If ((Get-PsSnapin |?{$_.Name -eq "Microsoft.SharePoint.PowerShell"})-eq
$null)
{
    Add-PsSnapin Microsoft.SharePoint.PowerShell -ErrorAction
SilentlyContinue | Out-Null
}
```

© 2008 - 2016 SharePoint Products

4.1 Get-SPPLicense

Retrieves the current license key for third-party products from www.sharepointproducts.com.

```
Get-SPPLicense [<CommonParameters>]
```

Example
Retrieve the license key and save it to a file.

```
Get-SPPLicense  | Out-File license.txt
```

4.2 Set-SPPLicense

Applies a license key for third-party products from www.sharepointproducts.com

```
Set-SPPLicense -License <String> [<CommonParameters>]
```

Parameter	Required	Type	Description
License	Yes	System.String	Specifies the license key string to apply

Example
Load the license key from a file and apply it to the farm.

```
$license = Get-Content license.txt | Out-String
Set-SPPLicense -License $license -Verbose
```

© 2008 - 2016 SharePoint Products

4.3 Get-CopyMoveSettings

Gets the global CopyMove farm settings or the local CopyMove settings for the specified site collection.

```
Get-CopyMoveSettings [-Url <String>] [<CommonParameters>]
```

Parameter	Required	Type	Description
Url	No	System.String	Specifies the URL of the site collection to retrieve CopyMove settings for. Omit to get the global CopyMove settings.

Example
Get the CopyMove settings for a specific site collection and save them to an XML file.

```
$settings = Get-CopyMoveSettings -SiteUrl "http://hostname/sites/
thesite"
$settings | Out-File CopyMoveSiteSettings.xml
```

4.4 Set-CopyMoveSettings

Updates new global CopyMove farm settings or new local CopyMove settings for the specified site collection.

Parameter Set 1

```
Set-CopyMoveSettings [-ConfigXml <String>] [-SiteUrl <String>]
   [<CommonParameters>]
```

Parameter	Required	Type	Description
ConfigXml	Yes	System.String	Specifies the CopyMove settings to apply.
SiteUrl	No	System.String	Specifies the URL address of the site collection to update CopyMove settings for. Omit to update the global CopyMove settings.

Parameter Set 2

```
Set-CopyMoveSettings [-SiteUrl <String>] [-LinksListUrl <String>] [-
   CustomJSUrl <String>] [-UseRecycleBin <SwitchParameter>] [-
   UsePermissionLevels <SwitchParameter>] [-ReportDetailedErrors
   <SwitchParameter>] [-EnableTransactionLimits <SwitchParameter>] [-
   InjectEventProperties <SwitchParameter>] [-MaxRecentFolders <Int32>]
   [-MaxListSubFolders <Int32>] [<CommonParameters>]
```

© 2008 - 2016 SharePoint Products

Parameter	Required	Type	Description
SiteUrl	No	System.String	Specifies the URL address of the site collection to update CopyMove settings for. Omit to update the global CopyMove settings.
LinksListUrl	No	System.String	Specifies the URL address of the SharePoint list where CopyMove stores personal links for the destination selection tree. The links include recent destinations and pinned SharePoint sites. Default value is ~mysite/Lists/CopyMoveLinks
CustomJSUrl	No	System.String	Specifies the URL address of a custom JavaScript file that CopyMove should include in all its dialog pages.
UseRecycleBin	No	SwitchParameter	Specifies if CopyMove should leave items in the recycle bin on moves. Default value is true.
UsePermissionLevels	No	SwitchParameter	Specifies if SharePoint permission levels should be used to restrict access to CopyMove. Default value is false.
ReportDetailedErrors	No	SwitchParameter	Determines if CopyMove should be allowed to reported detailed system errors in the user interface. Default value is true.
EnableTransactionLimits	No	SwitchParameter	Determines if CopyMove should enforce transaction limits. Default value is true.
InjectEventProperties	No	SwitchParameter	When enabled, CopyMove will inject the SPPCopyMoveEvent property into the property bag of SharePoint files during import. SharePoint event receivers will in turn be able to detect when CopyMove is importing a file and when it completes. Default value is false.
MaxRecentFolders	No	Int32	Specifies the maximum number of recent folders to track per user in the CopyMove destination selection tree.
MaxListSubFolders	No	Int32	Specifies the maximum number of sub-folders to query and show at the same level in the CopyMove destination selection tree.

Examples
Load CopyMove settings from an XML file and apply them to a specific site collection.

```
$settings = Get-Content CopyMoveSiteSettings.xml | Out-String
Set-CopyMoveSettings -ConfigXml $settings -SiteUrl "http://hostname/
sites/thesite" -Verbose
```

Updates the MaxRecentFolders property of the CopyMove settings in the site collection http://hostname/sites/thesite

© 2008 - 2016 SharePoint Products

```
Set-CopyMoveSettings -SiteUrl http://hostname/sites/thesite -
MaxRecentFolders 20
```

Updates the MaxRecentFolders property of the global CopyMove settings

```
Set-CopyMoveSettings -MaxRecentFolders 20
```

4.5 Remove-CopyMoveSettings

Removes all CopyMove settings from the specified site collection, which will in turn start inheriting the global CopyMove settings configured at the level of the farm.

```
Remove-CopyMoveSettings -SiteUrl <String> [-Force] [<CommonParameters>]
```

Parameter	Required	Type	Description
SiteUrl	Yes	System.String	Specifies the URL of the site collection for which to remove all CopyMove settings.
Force	No	SwitchParameter	Suppress confirmation before executing the command.

Example
Removes all CopyMove settings from the site collection http://hostname/sites/thesite.

```
Remove-CopyMoveSettings http://hostname/sites/thesite
```

© 2008 - 2016 SharePoint Products

4.6 Set-CopyMovePermissionLevel

Updates the CopyMove permissions for users in the specified site collection or sub-site. It is recommended to keep updates to the level of the site collection. The CopyMove permissions are applied to the specified SharePoint permission level, e.g. Contribute or Read. However, the permissions are only respected when CopyMove has also been configured to restrict access to authorized users. See the Global Settings section or the Site Collection Settings section in the Administration chapter. This effect of this PowerShell Cmdlet is identical to manually configuring CopyMove permissions as described in the Permissions Management section.

```
Set-CopyMovePermissionLevel -SiteUrl <String> -PermissionLevel <String>
[-AllowCopy <SwitchParameter>] [-AllowMove <SwitchParameter>] [-
AllowExport <SwitchParameter>] [-AllowImport <SwitchParameter>]
[<CommonParameters>]
```

Parameter	Required	Type	Description
SiteUrl	Yes	System.String	Specifies the URL address of the SharePoint site for which to updated permissions.
PermissionLevel	Yes	System.String	Specifies the name of the permission level to update.
AllowCopy	No	SwitchParameter	Allow users to copy items in the site.
AllowMove	No	SwitchParameter	Allow users to move items in the site.
AllowExport	No	SwitchParameter	Allow users to export items from the site.
AllowImport	No	SwitchParameter	Allow users to import items to the site.

Examples
Allow users with contribute permission to also copy, move, export and move items within the site collection http://hostname/sites/thesite.

```
Set-CopyMovePermssionLevel -SiteUrl "http://hostname/sites/thesite" -
PermissionLevel Contribute -AllowCopy -AllowMove -AllowExport -
AllowImport
```

Same as above but using the shorter parameter alias names.

```
Set-CopyMovePermssionLevel -Site "http://hostname/sites/thesite" -Level
Contribute -Copy -Move -Export -Import
```

Allow users with read only permission to copy and export items.

```
Set-CopyMovePermssionLevel -Site "http://hostname/sites/thesite" -Level
Read -Copy -Export
```

© 2008 - 2016 SharePoint Products

4.7 Copy-SPPItem

Copies list items, documents and folders across sites and Web applications.

```
Copy-SPPItem -SourceUri <String> -TargetUri <String> [-Item <String[]>]
[-HaltOnWarning <SwitchParameter>] [-IncludeTimeStamps
<SwitchParameter>] [-IncludeUserInfo <SwitchParameter>] [-
IncludeVersions <SwitchParameter>] [-IncludePermissions
<SwitchParameter>] [-Recursive <SwitchParameter>] [-
DisableTransactionLimits <SwitchParameter>] [<CommonParameters>]
```

Parameter	Required	Type	Description
SourceUri	Yes	System.String	The absolute URL address of the SharePoint folder to copy items from.
TargetUri	Yes	System.String	The absolute URL address of the SharePoint folder to copy items to.
Item	No	System.String[]	Array of source items to copy. The items can be specified by their URL, integer id or Guid. The URLs can be folder relative, site relative or server relative. Finally, it is also possible to include simple wildcards like * and *.* and *.pdf and *partialfilename*. Specify the Recursive switch parameter to also apply the wildcard(s) to items in sub folders.
HaltOnWarning	No	SwitchParameter	Aborts the copy transaction on warnings detected before copying starts. Default value is false.
IncludeTimeStamps	No	SwitchParameter	Preserves the item created time and item last modified time on the item copies. Default value is false.
IncludeUserInfo	No	SwitchParameter	Preserves the item created by user and item last modified by user on the item copies. Default value is false.
IncludeVersions	No	SwitchParameter	Preserves the complete version history on the item copies. Requires the destination list to have the same versioning settings as the source list. Default value is false.
IncludePermissions	No	SwitchParameter	Preserves any item level permissions on the item copies. Inherited permissions are not preserved. Default value is false.
Recursive	No	SwitchParameter	Applies any specified item wildcard(s) to items in sub folders besides items in the specified source folder. Default value is false.
DisableTransaction Limits	No	SwitchParameter	Disables the CopyMove transaction limits for the total item count and the total item size in MB that can be copied in one transaction. Use this switch when copying large lists and

© 2008 - 2016 SharePoint Products

Parameter	Required	Type	Description
			libraries that exceed the transaction limits configured on the global CopyMove settings page in Central Administration. However, the transaction may as a result take a long time to complete and consume a significant amount of memory.

Examples
Copy a single document and rename it in the destination.

```
Copy-SPPItem -SourceUri "http://hostname/sites/thesite1/Shared
Documents/Document.docx" -TargetUri "http://hostname/sites/thesite2/
Shared Documents/New Document.docx"
```

Copy two documents with full version history from a document library in one site to a compatible document library in another site.

```
$items = @("Document.docx", "Presentation.pptx")

Copy-SPPItem
  -SourceUri "http://hostname/sites/thesite1/Shared Documents"
  -TargetUri "http://hostname/sites/thesite2/Shared Documents"
  -Item $items
  -IncludeVersions
```

Recursively copy all Microsoft Word and Microsoft PowerPoint documents with full version history from a document library in one site to a compatible document library in another site.

```
$items = @("*.docx", "*.pptx")

Copy-SPPItem
  -SourceUri "http://hostname/sites/thesite1/Shared Documents"
  -TargetUri "http://hostname/sites/thesite2/Shared Documents"
  -Item $items
  -Recursive
  -IncludeVersions
```

© 2008 - 2016 SharePoint Products

4.8 Move-SPPItem

Moves list items, documents and folders across sites and Web applications.

```
Move-SPPItem -SourceUri <String> -TargetUri <String> [-Item <String[]>]
[-HaltOnWarning <SwitchParameter>] [-IncludeTimeStamps
<SwitchParameter>] [-IncludeUserInfo <SwitchParameter>] [-
IncludeVersions <SwitchParameter>] [-IncludePermissions
<SwitchParameter>] [-Recursive <SwitchParameter>] [-
DisableTransactionLimits <SwitchParameter>] [<CommonParameters>]
```

Parameter	Required	Type	Description
SourceUri	Yes	System.String	The absolute URL address of the SharePoint folder to move items from.
TargetUri	Yes	System.String	The absolute URL address of the SharePoint folder to move items to.
Item	No	System.String[]	Array of source items to move. The items can be specified by their URL, integer id or Guid. The URLs can be folder relative, site relative or server relative. Finally, it is also possible to include simple wildcards like * and *.* and *.pdf and *partialfilename*. Specify the Recursive switch parameter to also apply the wildcard(s) to items in sub folders.
HaltOnWarning	No	SwitchParameter	Aborts the move transaction on warnings detected before moving starts. Default value is false.
IncludeTimeStamps	No	SwitchParameter	Preserves the item created time and item last modified time on the destination items. Default value is true.
IncludeUserInfo	No	SwitchParameter	Preserves the item created by user and item last modified by user on the destination items. Default value is true.
IncludeVersions	No	SwitchParameter	Preserves the complete version history on the destination items. Requires the destination list to have the same versioning settings as the source list. Default value is true.
IncludePermissions	No	SwitchParameter	Preserves any item level permissions on the destination items. Inherited permissions are not preserved. Default value is false.
Recursive	No	SwitchParameter	Applies any specified item wildcard(s) to items in sub folders besides items in the specified source folder. Default value is false.
DisableTransaction Limits	No	SwitchParameter	Disables the CopyMove transaction limits for the total item count and the total item size in MB that can be moved in one transaction. Use this switch when moving large lists and libraries

© 2008 - 2016 SharePoint Products

Parameter	Required	Type	Description
			that exceed the transaction limits configured on the global CopyMove settings page in Central Administration. However, the transaction may as a result take a long time to complete and consume a significant amount of memory.

Examples
Move a single document and rename it in the destination.

```
Move-SPPItem -SourceUri "http://hostname/sites/thesite1/Shared
Documents/Document.docx" -TargetUri "http://hostname/sites/thesite2/
Shared Documents/New Document.docx"
```

Move two documents with full fidelity from a document library in one site to a compatible document library in another site.

```
$items = @("Document.docx", "Presentation.pptx")

Move-SPPItem
  -SourceUri "http://hostname/sites/thesite1/Shared Documents"
  -TargetUri "http://hostname/sites/thesite2/Shared Documents"
  -Item $items
  -IncludePermissions:$true
```

Recursively move all Microsoft Word and Microsoft PowerPoint documents with full version history from a document library in one site to a compatible document library in another site.

```
$items = @("*.docx", "*.pptx")

Move-SPPItem
  -SourceUri "http://hostname/sites/thesite1/Shared Documents"
  -TargetUri "http://hostname/sites/thesite2/Shared Documents"
  -Item $items
  -Recursive
  -IncludeVersions
```

© 2008 - 2016 SharePoint Products

4.9 Export-SPPItem

Exports list items, documents and folders from a SharePoint list/library to a ZIP file on disk.

```
Export-SPPItem -SourceUri <String> -TargetUri <String> -Item <String[]>
[-HaltOnWarning <SwitchParameter>] [-IncludeVersions <SwitchParameter>]
[-IncludePermissions <SwitchParameter>] [-DisableTransactionLimits
<SwitchParameter>] [<CommonParameters>]
```

Parameter	Required	Type	Description
SourceUri	Yes	System.String	The absolute URL address of the SharePoint list or list folder to export items from.
TargetUri	Yes	System.String	The absolute or relative path and filename of the ZIP file to export items to.
Item	Yes	System.String[]	Array of source items to export. The items can be specified by their URL, integer id or Guid. The URLs can be folder relative, site relative or server relative. Finally, it is also possible to include simple wildcards like * and *.* and *.pdf and *partialfilename*. Specify the Recursive switch parameter to also apply the wildcard(s) to items in sub folders.
HaltOnWarning	No	SwitchParameter	Aborts the export transaction on warnings detected before exporting starts. Default value is false.
IncludeVersions	No	SwitchParameter	Preserves the complete version history on the exported items. Default value is false.
IncludePermissions	No	SwitchParameter	Preserves any item level permissions on the exported items. Inherited permissions are not preserved. Default value is false.
DisableTransaction Limits	No	SwitchParameter	Disables the CopyMove transaction limits for the total item count and the total item size in MB that can be exported in one transaction. Use this switch when exporting large lists and libraries that exceed the transaction limits configured on the global CopyMove settings page in Central Administration. However, the transaction may as a result take a long time to complete and consume a significant amount of memory.

Examples
Exports two documents and a complete folder branch with full version history from a document library.

```
$items = @("Document.docx", "Presentation.pptx", "FolderName")
```

© 2008 - 2016 SharePoint Products

```
Export-SPPItem
  -SourceUri "http://hostname/sites/thesite1/Shared Documents"
  -TargetUri "SharedDocuments.zip"
  -Item $items
  -IncludeVersions
```

Recursively exports all Microsoft Word and Microsoft PowerPoint documents with full version history from a document library.

```
$items = @("*.docx", "*.pptx")

Export-SPPItem
  -SourceUri "http://hostname/sites/thesite1/Shared Documents"
  -TargetUri "SharedDocuments.zip"
  -Item $items
  -Recursive
  -IncludeVersions
```

© 2008 - 2016 SharePoint Products

4.10 Export-SPPList

Exports all list items, documents and folders from a SharePoint list/library to a ZIP file on disk.

```
Export-SPPList -SourceUri <String> -TargetUri <String> [-HaltOnWarning
<SwitchParameter>] [-IncludeVersions <SwitchParameter>] [-
IncludePermissions <SwitchParameter>] [-DisableTransactionLimits
<SwitchParameter>] [<CommonParameters>]
```

Parameter	Required	Type	Description
SourceUri	Yes	System.String	The absolute URL address of the SharePoint list or list folder to export items from.
TargetUri	Yes	System.String	The absolute or relative path and filename of the ZIP file to export items to.
HaltOnWarning	No	SwitchParameter	Aborts the export transaction on warnings detected before exporting starts. Default value is false.
IncludeVersions	No	SwitchParameter	Preserves the complete version history on the exported items. Default value is false.
IncludePermissions	No	SwitchParameter	Preserves any item level permissions on the exported items. Inherited permissions are not preserved. Default value is false.
DisableTransaction Limits	No	SwitchParameter	Disables the CopyMove transaction limits for the total item count and the total item size in MB that can be exported in one transaction. Use this switch when exporting large lists and libraries that exceed the transaction limits configured on the global CopyMove settings page in Central Administration. However, the transaction may as a result take a long time to complete and consume a significant amount of memory.

Example
Exports all documents and folders in a document library with full version history.

```
Export-SPPList
  -SourceUri "http://hostname/sites/thesite1/Shared Documents"
  -TargetUri "SharedDocuments.zip"
  -IncludeVersions
```

© 2008 - 2016 SharePoint Products

4.11 Import-SPPItem

Imports list items, documents and folders from a ZIP file on disk to a SharePoint list or library. The ZIP file can either be a file that was previously exported using the CopyMove export function or it can be any ZIP file containing files and folders. Version history and author information can only be preserved with ZIP files exported by CopyMove.

```
Import-SPPItem -SourceUri <String> -TargetUri <String> -Item <String[]>
[-HaltOnWarning <SwitchParameter>] [-IncludeTimeStamps
<SwitchParameter>] [-IncludeUserInfo <SwitchParameter>] [-
IncludeVersions <SwitchParameter>] [-IncludePermissions
<SwitchParameter>] [-DisableTransactionLimits <SwitchParameter>]
[<CommonParameters>]
```

Parameter	Required	Type	Description
SourceUri	Yes	System.String	The absolute or relative path and filename of the ZIP file to import items from.
TargetUri	Yes	System.String	The absolute URL address of the SharePoint list or list folder to import items to.
Item	No	System.String[]	Array of source items to include from the ZIP. The items can be specified by their URL, integer id or Guid. The URLs can be folder relative, site relative or server relative. Omit this parameter to import all content from the ZIP.
HaltOnWarning	No	SwitchParameter	Aborts the import transaction on warnings detected before importing starts. Default value is false.
IncludeTimeStamps	No	SwitchParameter	Preserves the item created time and item last modified time on the imported items. Default value is false.
IncludeUserInfo	No	SwitchParameter	Preserves the item created by user and item last modified by user on the imported items. Default value is false.
IncludeVersions	No	SwitchParameter	Preserves the complete version history on the imported items. Requires versioning to be enabled in the destination list. Default value is false.
IncludePermissions	No	SwitchParameter	Preserves any item level permissions on the imported items. Default value is false.
DisableTransaction Limits	No	SwitchParameter	Disables the CopyMove transaction limits for the total item count and the total item size in MB that can be imported in one transaction. Use this switch when importing large lists and libraries that exceed the transaction limits configured on the global CopyMove settings page in Central Administration. However, the transaction may as a result take a long time to

© 2008 - 2016 SharePoint Products

Parameter	Required	Type	Description
			complete and consume a significant amount of memory.

Example 1
Imports all files and folders from the *SharedDocuments.zip* archive to a SharePoint document library while preserving authoring information and full version history.

```
Import-SPPItem
  -SourceUri "SharedDocuments.zip"
  -TargetUri "http://hostname/sites/thesite/Shared Documents"
  -IncludeTimeStamps
  -IncludeUserInfo
  -IncludeVersions
```

Example 2
Imports selected documents from the *SharedDocuments.zip* archive to a SharePoint document library while preserving authoring information and full version history.

```
$items = @("Document.docx", "Presentation.pptx", "Folder/Accouting.xls")

Import-SPPItem
  -SourceUri "SharedDocuments.zip"
  -TargetUri "http://hostname/sites/thesite/Shared Documents"
  -Item $items
  -IncludeTimeStamps
  -IncludeUserInfo
  -IncludeVersions
```

© 2008 - 2016 SharePoint Products

Part V

Extending CopyMove

5 Extending CopyMove

CopyMove can be extended by SharePoint developers to accommodate special business requirements like support for custom list templates and integration with list event receivers. Developers can also invoke CopyMove from custom code locally through the available .NET API or remotely through the SOAP web service or the REST web service. Finally, workflow designers can also leverage CopyMove in SharePoint 2010 and SharePoint 2013 Workflows created with Visual Studio or SharePoint Designer 2013. The following sections in this chapter documents the different ways by which it is possible to extend CopyMove.

5.1 API

The Application Programming Interface for CopyMove enables SharePoint software developers to integrate CopyMove with their own SharePoint applications. Basically, the API supports the operations; copy, move, export and import. They can respectively be leveraged to copy, move, export and import one or more folders, documents and list items at a time. The items are handled with full fidelity including Metadata, version history and item level permissions. In this context, the Metadata includes the created and modified timestamps, the created by and modified by users, other system Metadata and custom Metadata.

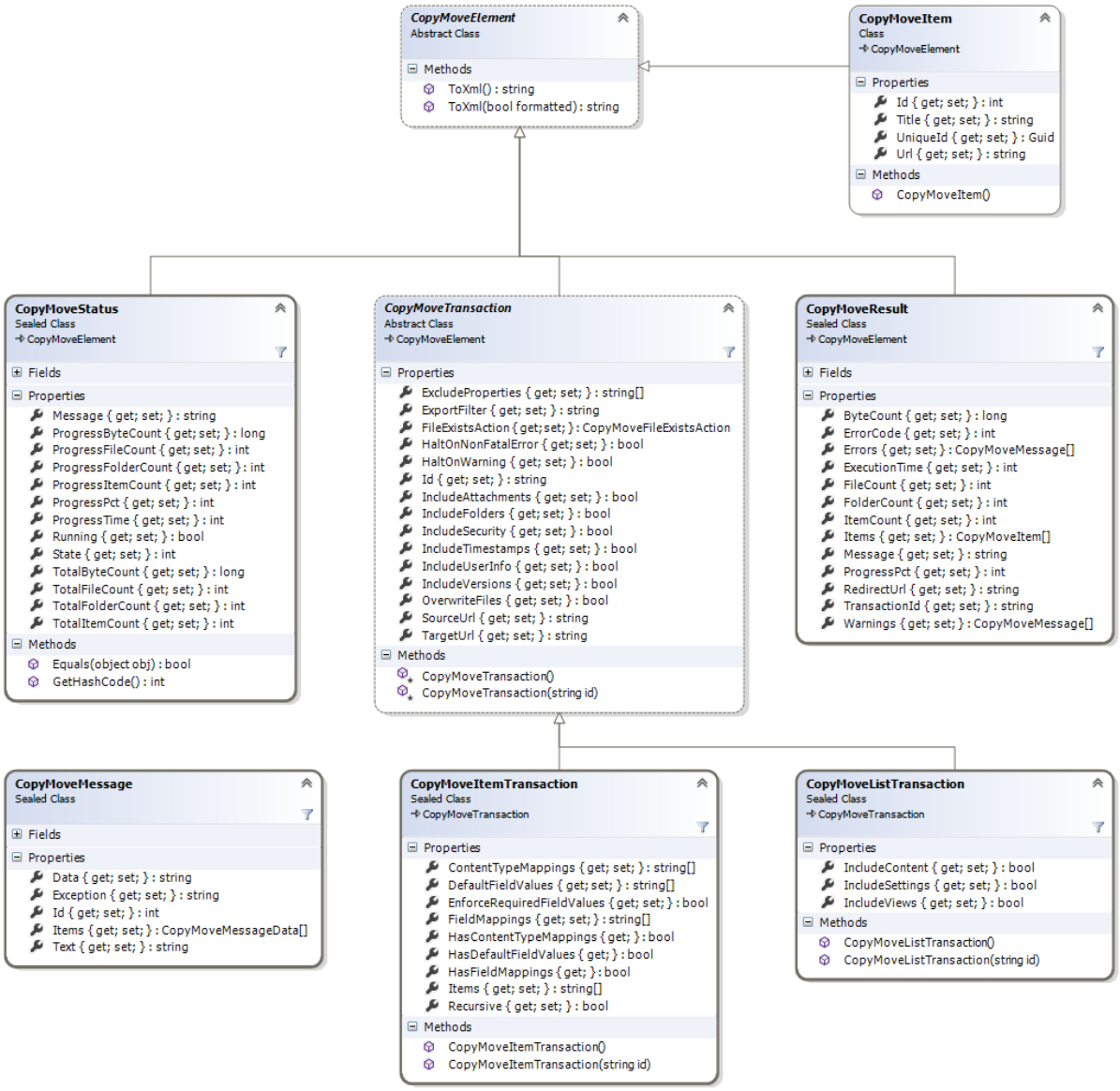
The API is exposed as a .NET object model for use in server side code and as a WCF service for remote client use. The .NET API exposes the following methods:

```
CopyMoveResult Copy(CopyMoveItemTransaction transaction);

CopyMoveResult Move(CopyMoveItemTransaction transaction);

CopyMoveResult Export(CopyMoveItemTransaction transaction);

CopyMoveResult Import(CopyMoveItemTransaction transaction);

CopyMoveStatus GetStatus(string transactionId);
```

where the WCF service is currently only exposing the Copy, Move and GetStatus methods. There are also some differences with respect to asynchronous invocation of the operations - but more about this later.

The *Copy, Move, Export* and *Import* methods all take an instance of the *CopyMoveItemTransaction* class as input and returns an instance of the *CopyMoveResult* class when the transaction is completed or aborted by errors or warnings. Larger transactions can take some time to complete and the *GetStatus* method can then be used to track the progress from another thread or from the same thread if the operations are invoked asynchronously as demonstrated later in this chapter. The following class diagram outlines the details of the input and output parameter classes in the API. The facade classes are documented in the next two sections.

© 2008 - 2016 SharePoint Products



The following tables documents each class and its properties shown in the class diagram above.

CopyMoveStatus

An instance of this class is returned by the *GetStatus* method and it contains progress information for the specified transaction.

Property	Type	Description
Message	string	Gets a textual message about the current progress of the copy or move transaction. The value of this property is designed to be used directly in a user interface. CopyMove displays the message together with the progress bar during a transaction.
ProgressByteCount	long	Gets the number of bytes copied or moved so far. Only document library files and list item attachments count.

© 2008 - 2016 SharePoint Products

CopyMoveStatus An instance of this class is returned by the *GetStatus* method and it contains progress information for the specified transaction.		
ProgressFileCount	int	Gets the number of files copied or moved so far by the transaction.
ProgressFolderCount	int	Gets the number of folders copied or moved so far by the transaction.
ProgressItemCount	int	Gets the number of list items copied or moved so far by the transaction.
ProgressPct	int	Gets the progress in percent.
ProgressTime	int	Gets the number of miliseconds that the transaction has been working.
Running	bool	Gets a value indicating whether the transaction is still running.
State	int	Gets the current state of the transaction.
TotalByteCount	long	Gets the total number of bytes to be copied or moved. Only document library files and list item attachments count.
TotalFileCount	int	Gets the total number of files to be copied or moved by the transaction.
TotalFolderCount	int	Gets the total number of folders to be copied or moved by the transaction.
TotalItemCount	int	Gets the total number of list items to be copied or moved by the transaction.

CopyMoveTransaction The base class for all CopyMove transaction types. Currently there is only one type but future versions of CopyMove might include a site and a list transaction type as well.		
Property	**Type**	**Description**
Id	string	Gets or sets the transaction id.
IncludeTimestamps	bool	Gets or sets a value indicating whether to include and preserve the created time and modified time on items. The default value is *true*.
IncludeUserInfo	bool	Gets or sets a value indicating whether to include and preserve the created by user and the last modified by user. The default value is *true*.
IncludeVersions	bool	Gets or sets a value indicating whether to include and preserve version history, if any. The default value is *true*.
IncludeSecurity	bool	Gets or sets a value indicating whether to include item level permissions or not. The default value is *false*.
IncludeFolders	bool	Gets or sets a value indicating whether to include sub-folders when copying, moving or exporting content. The default value is *false*.
ExcludeProperties	string[]	Optional. Gets or sets an array of Metadata property names to exclude. To exclude values for a specific list column, include the internal name of that column here.
OverwriteFiles	bool	Gets or sets a value indicating whether to overwrite existing files. **Deprecated - instead use the FileExistsAction property.**
FileExistsAction	enum	Gets or sets the action for existing files. Possible values are: 1 = CopyMoveFileExistsAction.Skip = The existing destination file is left as is and the source file is not copied/moved. 2 = CopyMoveFileExistsAction.Overwrite = The existing destination file is deleted and replaced with a copy of the source file. 3 = CopyMoveFileExistsAction.Rename = The existing destination file is

© 2008 - 2016 SharePoint Products

CopyMoveTransaction The base class for all CopyMove transaction types. Currently there is only one type but future versions of CopyMove might include a site and a list transaction type as well.		
		left as is and the source file is renamed with a new unique file name. 4 = CopyMoveFileExistsAction.AddAsNewVersion = The source file is added as a new version to the existing destination file. However, only if versioning is enabled.
ExportFilter	string	Optional. Get or sets a SharePoint CAML query to only export matching items. Applies to exports only.
HaltOnNonFatalError	bool	Gets or sets a value indicating if the transaction should abort on non fatal errors.The default is *true*.
HaltOnWarning	bool	Gets or sets a value indicating if the transaction should abort on warnings. A typical warning is when copying or moving a file and it already exists in the destination. The CopyMove UI always starts with this property set to *true*. Any warnings will abort the transaction and CopyMove displays them to the user for confirmation. If the user confirms the warnings then CopyMove restarts the transaction with the HaltOnWarning property set to *false*. The default value is *true*.
SourceUrl	string	Gets or sets the absolute URL of the source list or the source folder in SharePoint. For import transactions this property must specify the Windows server file path to the ZIP archive or file system folder containing the content to import.
TargetUrl	string	Gets or sets the absolute URL of the destination list or the destination folder in SharePoint. For export transactions this property must specify the Windows server path to a folder where the resulting ZIP archive should be saved to.

CopyMoveItemTransaction Specifies a transaction for one or more folders, documents or list items.		
Property	**Type**	**Description**
ContentTypeMapings	string[]	Gets or sets an array of content type mappings in the string format <SourceContentTypeId>:<TargetContentTypeId>
DefaultFieldValues	string[]	Gets or sets an array of default values for required fields where the source item does not provide a value for the column in the destination.
EnforceRequiredFieldValues	bool	Gets or sets a value that determines whether CopyMove will enforce missing values on required fields. The default value is *true* .
Items	string[]	Gets or sets an array of the folders, documents or items to include in the transaction. Each source item can be specified by its absolute URL, server relative URL, filename, list item GUID or list item ID. All items must exist in the source folder specified by the SourceUrl property.
Recursive	bool	Obsolete. Use the *IncludeFolders* property instead.
SourceFolder	string	Obsolete. Gets or sets the absolute URL of the source list or source folder.

© 2008 - 2016 SharePoint Products

CopyMoveItemTransaction Specifies a transaction for one or more folders, documents or list items.		
TargetFolder	string	Obsolete. Gets or sets the absolute URL of the destination list or source folder. It can be any compatible folder in the SharePoint farm.

CopyMoveResult An instance of this class is returned by the Copy and Move methods when the transaction ends.		
Property	**Type**	**Description**
ByteCount	long	Gets the total number of bytes copied or moved. Only document library files and list item attachments count.
ErrorCode	int	Gets the overall status of the completed transaction. Possible values are: 0 = Ok 1 = Transaction aborted with warnings 2 = Transaction aborted with errors
Errors	CopyMoveMessage[]	Gets an array of detailed error messages if the ErrorCode=2
ExecutionTime	int	Gets the total number of miliseconds it took to execute the transaction.
FileCount	int	Gets the total number of files that were copied or moved by the transaction.
FolderCount	int	Gets the total number of folders that were copied or moved by the transaction.
ItemCount	int	Gets the total number of list items that were copied or moved by the transaction.
Items	CopyMoveItem[]	Gets an array holding one entry with information for each destination item that was created during a copy, move or import transaction.
Message	string	Gets a textual message describing the outcome of the transaction. The value of this property is designed to be used directly in a user interface. CopyMove displays this message to the user when a transaction completes.
TransactionId	string	Gets the id of the transaction that produced the result.
Warnings	CopyMoveMessage[]	Gets an array of detailed warning messages if the ErrorCode=1

CopyMoveMessage Represents a message from CopyMove.		
Property	**Type**	**Description**
Exception	Exception	If the message represents an error then this property contains the Exception that triggered it.
Id	int	Gets the message id, which can be any of the following values defined as public constants in the class:

© 2008 - 2016 SharePoint Products

CopyMoveMessage Represents a message from CopyMove.		
		Warning_FileExist = 101; Warning_FilesExist = 102; Warning_LossOfMajorVersionHistory = 110; Warning_LossOfMinorVersionHistory = 111; Warning_LossOfContentType = 112; Warning_LossOfListFieldValue = 113; Warning_LossOfPageLayout = 115; Warning_LossOfPartialMajorVersionHistory = 116; Warning_LossOfWebPart = 117; Warning_MoveInPlaceRecord = 145; Warning_MoveItemOnHold = 146; Error_InvalidTarget = 200; Error_SiteNotFound = 201; Error_ListNotFound = 202; Error_FolderNotFound = 203; Error_ItemNotFound = 204; Error_SourceNotFound = 205; Error_InvalidSource = 206; Error_MissingManifest = 207; Error_SQL = 208; Error_IO = 209; Error_AccessDenied_OpenSite = 210; Error_AccessDenied_CopyItem = 211; Error_AccessDenied_MoveItem = 212; Error_AccessDenied_ExportItem = 215; Error_AccessDenied_ImportItem = 216; Error_AccessDenied_SourceList = 217; Error_AccessDenied_SourceFolder = 218; Error_AccessDenied_TargetList = 219; Error_AccessDenied_TargetFolder = 220; Error_AccessDenied_OverwriteTargetItem = 221; Error_FileTargetUrlTooLong = 230; Error_FolderTargetUrlTooLong = 231; Error_IncompatibleTargetList = 232; Error_TransactionItemCountExceeded = 234; Error_TransactionFileCountExceeded = 235; Error_TransactionFolderCountExceeded = 236; Error_TransactionFileSizeExceeded = 237; Error_TargetFolderIsLinkStub = 238; Error_CannotCreateFoldersInDocumentSet = 239; Error_OverwriteCheckedOutFile = 240; Error_OverwriteLockedFile = 241; Error_OverwriteInPlaceRecord = 242; Error_MoveCheckedOutFile = 243; Error_MoveLockedFile = 244; Error_MoveInPlaceRecord = 245; Error_MoveFolderToSelf = 246; Error_MoveItemOnHold = 247; Error_TargetFolderCreationNotAllowed = 248;

© 2008 - 2016 SharePoint Products

CopyMoveMessage Represents a message from CopyMove.		
		Error_DuplicateTargetFiles = 249; Error_CannotDeleteSourceItem = 250; Error_ConcurrentMove = 251; Error_NoItems = 252; Error_CannotExportWebPart = 253; Error_CannotImportWebPart = 254; Error_FileSizeExceeded = 255; Error_FileVersionSizeExceeded = 256; Error_StorageQuotaExceeded = 257; Error_TargetSiteNoPublishing = 260; Error_MissingRequiredField = 301;
Text	string	Gets the message text.

© 2008 - 2016 SharePoint Products

5.1.1 CopyMove .NET API

Integrating server side code with CopyMove is best done using the CopyMove .NET API, which is exposed through the *SharePointProducts.CopyMove.CopyMoveProcessor* class in addition to the input and output parameter classes outlined in the previous section. To work with the API in Visual Studio it is necessary to add a reference to the following .NET assemblies located in the Global Assembly Cache:

- SharePointProducts.Platform.dll
- SharePointProducts.CopyMove.dll

The *CopyMoveProcessor* class contains the public methods listed below. The *Copy, Move, Export* and *Import* methods are synchronous and will not return before the transaction completes whereas the *CopyAsync, MoveAsync, ExportAsync* and *ImportAsync* methods are asynchronous. The latter methods launches a background worker thread and returns to the caller with the transaction id. The method *GetStatus* can in turn be used to track the progress of a specific transaction without blocking the caller. Finally, the *GetResult* method is used to obtain the result of an asynchronous transaction. Calling the method before the worker thread has completed the transaction blocks the caller until the transaction has completed and an instance of the *CopyMoveResult* class is ready to return.

```
public CopyMoveResult Copy(CopyMoveItemTransaction transaction)
public string CopyAsync(CopyMoveItemTransaction transaction)

public CopyMoveResult Move(CopyMoveItemTransaction transaction)
public string MoveAsync(CopyMoveItemTransaction transaction)

public CopyMoveResult Export(CopyMoveItemTransaction transaction)
public CopyMoveResult Export(CopyMoveListTransaction transaction)
public string ExportAsync(CopyMoveItemTransaction transaction)
public string ExportAsync(CopyMoveListTransaction transaction)

public CopyMoveResult Import(CopyMoveItemTransaction transaction)
public string ImportAsync(CopyMoveItemTransaction transaction)

public CopyMoveStatus GetStatus(string transactionId)
public CopyMoveResult GetResult(string transactionId)
```

How-to instantiate the CopyMoveProcessor

Creating a new instance with the default constructor yields a new instance using the CopyMove settings stored in SharePoint and which will run as the current SharePoint user if the current thread has a SharePoint context. Otherwise, it will run as the current Windows user.

```
var processor = new CopyMoveProcessor();
```

There is also an overloaded constructor that makes it possible to specify the settings to user and the user to run as.

```
var settings = new CopyMoveSettings {
    TemporaryFilesLocation = @"D:\CopyMoveTemp",
    UseRecycleBin = false,
    KeepTemporaryFilesUntilCleanup = true
};
var processor = new CopyMoveProcessor(settings, @"SHAREPOINT\SYSTEM");
```

© 2008 - 2016 SharePoint Products

How-to Copy/Move a single file

The sample code below demonstrates how to use the CopyMove .NET API to move a single file from one location to another within the same SharePoint farm.

```
// Create a new instance of the CopyMoveProcessor class
var processor = new CopyMoveProcessor();

// Prepare new CopyMove item transaction
var transaction = new CopyMoveItemTransaction();
transaction.FileExistsAction = CopyMoveFileExistsAction.Overwrite;
transaction.IncludeTimestamps = true;
transaction.IncludeUserInfo = true;
transaction.IncludeVersions = true;
transaction.IncludeSecurity = true;
transaction.HaltOnWarning = false;

// The absolute URL of the source file and the target file
transaction.SourceUrl = "https://host/sites/site1/doclib/folder/document.ext";
transaction.TargetUrl = "https://host/sites/site2/doclib/folder/newdocument.ext";

// Move the document and wait for the transaction to complete
CopyMoveResult result = processor.Move(transaction);

if (result.ErrorCode == 0)
{
    // Success - the transaction completed without errors or warnings
} else if (result.ErrorCode == 1)
{
    // Warning - the transaction completed with warnings
} else if (result.ErrorCode == 2)
{
    // Error - the transaction was aborted with errors
}
```

How-to Copy/Move multiple files

This example demonstrates how to use the CopyMove .NET API to move a folder and two documents from one document library to another document library within the same SharePoint farm.

```
// Create a new instance of the CopyMoveProcessor class
var processor = new CopyMoveProcessor();

// Prepare new CopyMove item transaction
var transaction = new CopyMoveItemTransaction();
transaction.FileExistsAction = CopyMoveFileExistsAction.Overwrite;
transaction.IncludeTimestamps = true;
transaction.IncludeUserInfo = true;
transaction.IncludeVersions = true;
transaction.IncludeSecurity = false;
transaction.HaltOnWarning = false;

// The absolute URL of the source folder or source list
transaction.SourceUrl = "https://host/sites/site1/doclib/folder";
```

© 2008 - 2016 SharePoint Products

```
// The absolute URL of the target folder or target list
transaction.TargetUrl = "https://host/sites/site2/doclib/folder";

// The source items to copy. Each item can be specified by its
// absolute URL, server relative URL, filename, list item GUID or list item ID.
// In this example, we just specify the filenames of a folder and two documents
// in the source folder specified above.
transaction.Items = new string[]
{
    "Test Folder", "Test Document.doc", "Test Document.pdf"
};

// Optional: Exclude one or more list columns
transaction.ExcludeProperties = new string[] { "FieldName1", "FieldName2" };

// Start the Move transaction in a separate thread. The MoveAsync method returns
// immediately after launching the background thread that does the actual
// copying of items.
string transactionId = processor.MoveAsync(transaction);

// CopyMove keeps track of the progress in the worker thread
// and you can optionally ask for it using the GetStatus method.
CopyMoveStatus status;
do
{
    Thread.Sleep(500);
    status = processor.GetStatus(transactionId);

    // TODO: Display progress to user
} while (status.IsRunning)

// Get the final result of the transaction. The GetResult
// method blocks until the transaction has completed or
// was aborted by an error
CopyMoveResult result = processor.GetResult(transactionId);

if (result.ErrorCode == 0)
{
    // Success - the transaction completed without errors or warnings
} else if (result.ErrorCode == 1)
{
    // Warning - the transaction completed with warnings
} else if (result.ErrorCode == 2)
{
    // Error - the transaction was aborted with errors
}
```

The sample code shown above can also be found in the `Form1.cs` file of the Visual Studio project named *API Sample*. It is included in the CopyMove product download.

How-to Export Files to ZIP

This example demonstrates how to use the CopyMove .NET API to export a folder and two documents

© 2008 - 2016 SharePoint Products

from a document library to a ZIP file on disk.

```
// Create a new instance of the CopyMoveProcessor class
var processor = new CopyMoveProcessor();

// Prepare new CopyMove item transaction
var transaction = new CopyMoveItemTransaction();
transaction.IncludeTimestamps = true;
transaction.IncludeUserInfo = true;
transaction.IncludeVersions = true;
transaction.IncludeSecurity = true;
transaction.HaltOnWarning = false;

// The absolute URL of the source folder or source list
transaction.SourceUrl = "https://host/sites/site1/doclib/folder";

// The file path of the zip file to export
transaction.TargetUrl = @"D:\CopyMove\Export.zip";

// The source items to copy. Each item can be specified by its
// absolute URL, server relative URL, filename, list item GUID or list item ID.
// In this example, we just specify the filenames of a folder and two documents
// in the source folder specified above.
transaction.Items = new string[]
{
    "Test Folder", "Test Document.doc", "Test Document.pdf"
};

// Export the items and wait for the transaction to complete
CopyMoveResult result = processor.Export(transaction);
```

How-to Import Files from ZIP

Importing the files again is just as easy:

```
// Create a new instance of the CopyMoveProcessor class
var processor = new CopyMoveProcessor();

// Prepare new CopyMove item transaction
var transaction = new CopyMoveItemTransaction();
transaction.IncludeTimestamps = true;
transaction.IncludeUserInfo = true;
transaction.IncludeVersions = true;
transaction.IncludeSecurity = true;
transaction.HaltOnWarning = false;

// The file path of the zip file to import
transaction.SourceUrl = @"D:\CopyMove\Export.zip";

// The absolute URL of the destination folder
transaction.TargetUrl = "https://host/sites/site1/doclib/folder";

// Export the items and wait for the transaction to complete
CopyMoveResult result = processor.Import(transaction);
```

© 2008 - 2016 SharePoint Products

5.1.2 CopyMove SOAP Web Service

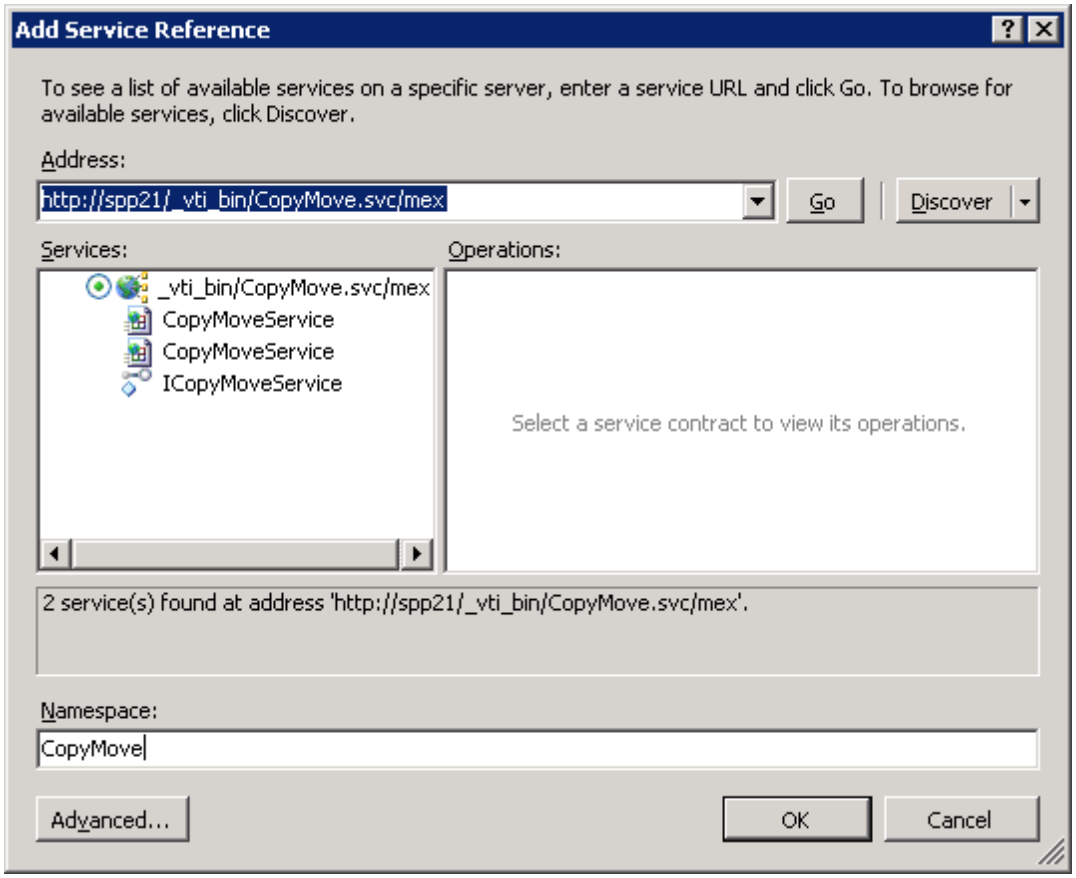
Remote clients can also leverage CopyMove through its integrated WCF SOAP Web Service that wraps the .NET API. The service is available at URLs constructed from the following pattern:

<protocol>://<Web application>/<Context Site>/_vti_bin/CopyMove.svc

where the protocol can be HTTP or HTTPS depending on the Web application setup. For example:

http://server/_vti_bin/CopyMove.svc or http://server/sites/site/_vti_bin/CopyMove.svc

The service is sensitive to the SharePoint site context and it is therefore important to always connect with the proper SharePoint site url address. From Visual Studio it is easy to connect to the service and build the required client classes. Just note that you need to add /mex to the service URL as shown in the screen shot below.



Once Visual Studio has successfully connected to the service, click *OK* to generate the client code that can be used to invoke the service. To include support for asynchronous invocation, first click the *Advanced* button and check the *Generate asynchronous operations* check box. Visual Studio will in turn generate a class named *CopyMoveServiceClient* with the following methods:

```
CopyMoveResult Copy(CopyMoveItemTransaction transaction);
```

© 2008 - 2016 SharePoint Products

```
IAsyncResult BeginCopy(CopyMoveItemTransaction transaction, AsyncCallback callback,
object asyncState);

CopyMoveResult EndCopy(IAsyncResult result);

CopyMoveResult Move(CopyMoveItemTransaction transaction);

IAsyncResult BeginMove(CopyMoveItemTransaction transaction, AsyncCallback callback,
object asyncState);

CopyMoveResult EndMove(IAsyncResult result);

CopyMoveStatus GetStatus(string transactionId);

string GetVersion();
```

See http://msdn.microsoft.com/en-us/library/ms734691.aspx for more information about accessing WCF services. Connecting to the CopyMove WCF SOAP service in code goes as in the following example with NTLM authentication. For other types of supported authentication schemes, please refer to the Microsoft WCF documentation on MSDN.

```
//
// Create proxy that connects to the service and authenticates with current user.
//
var binding = new BasicHttpBinding();
binding.Security.Mode = BasicHttpSecurityMode.TransportCredentialOnly;
binding.Security.Transport.ClientCredentialType = HttpClientCredentialType.Ntlm;
var endpoint = new EndpointAddress("http://server/sites/site/_vti_bin/CopyMove.svc");
CopyMove.CopyMoveServiceClient proxy = new CopyMove.CopyMoveServiceClient(binding,
endpoint);
proxy.ClientCredentials.Windows.AllowedImpersonationLevel = System.Security.Principal.
TokenImpersonationLevel.Impersonation;

//
// Get the CopyMove version number. This service call will fail if a
// connection to the WCF Service cannot be established.
//
string version = proxy.GetVersion();
```

From here the code is virtually identical to that of the .NET API. The following example moves a folder and two documents from one document library to another document library within the same SharePoint farm.

```
// Prepare new CopyMove item transaction
var transaction = new CopyMoveItemTransaction();
transaction.FileExistsAction = CopyMoveFileExistsAction.Overwrite;
transaction.IncludeTimestamps = true;
transaction.IncludeUserInfo = true;
transaction.IncludeVersions = true;
transaction.IncludeSecurity = false;
transaction.HaltOnWarning = false;
```

© 2008 - 2016 SharePoint Products

```
// The absolute URL of the source folder or source list
transaction.SourceUrl = "http://server/sites/site1/doclib/folder";

// The absolute URL of the target folder or target list
transaction.TargetUrl = "http://server/sites/site2/doclib/folder";

// The source items to copy. Each item can be specified by its
// absolute URL, server relative URL, filename, list item GUID or list item ID.
// In this example, we just specify the filenames of a folder and two documents
// in the source folder specified above.
transaction.Items = new string[]
{
    "Test Folder", "Test Document.doc", "Test Document.pdf"
};

// Optional: Exclude one or more list columns
transaction.ExcludeProperties = new string[] { "FieldName1", "FieldName2" };

// Run the specified move transaction.
CopyMoveResult result = proxy.Move(transaction);

if (result.ErrorCode == 0)
{
    // Success - the transaction completed without errors or warnings
} else if (result.ErrorCode == 1)
{
    // Warning - the transaction completed with warnings
} else if (result.ErrorCode == 2)
{
    // Error - the transaction was aborted with errors
}
```

The sample code shown above can also be found in the `Form1.cs` file of the Visual Studio project named *WCF Sample*. It is included in the CopyMove product download.

© 2008 - 2016 SharePoint Products

5.1.3 CopyMove REST Web Service

CopyMove also features a REST Web service that remote clients can call. The service is available at URLs constructed from the following pattern:

<protocol>://<Web application>/<Context Site>/_vti_bin/CopyMoveRest.svc

where the protocol can be HTTP or HTTPS depending on the SharePoint Web application setup. For example:

https://server/_vti_bin/CopyMoveRest.svc or https://server/sites/site/_vti_bin/CopyMoveRest.svc

The service is sensitive to the SharePoint site context and it is therefore important to always connect with the proper SharePoint site url address. The input/output format is always XML regardless of the HTTP Accept header. To work with JSON data instead of XML, simply use the equivalent CopyMoveJSON.svc REST service. The XML service has these seven end points:

Name	Method	Url
Copy	POST	~site/_vti_bin/CopyMoveRest.svc/Copy
Move	POST	~site/_vti_bin/CopyMoveRest.svc/Move
Result	GET	~site/_vti_bin/CopyMoveRest.svc/Result/{transactionid}
Status	GET	~site/_vti_bin/CopyMoveRest.svc/Status/{transactionid}
SampleItemTransaction	GET	~site/_vti_bin/CopyMoveRest.svc/SampleItemTransaction
Settings	GET	~site/_vti_bin/CopyMoveRest.svc/Settings
Version	GET	~site/_vti_bin/CopyMoveRest.svc/Version

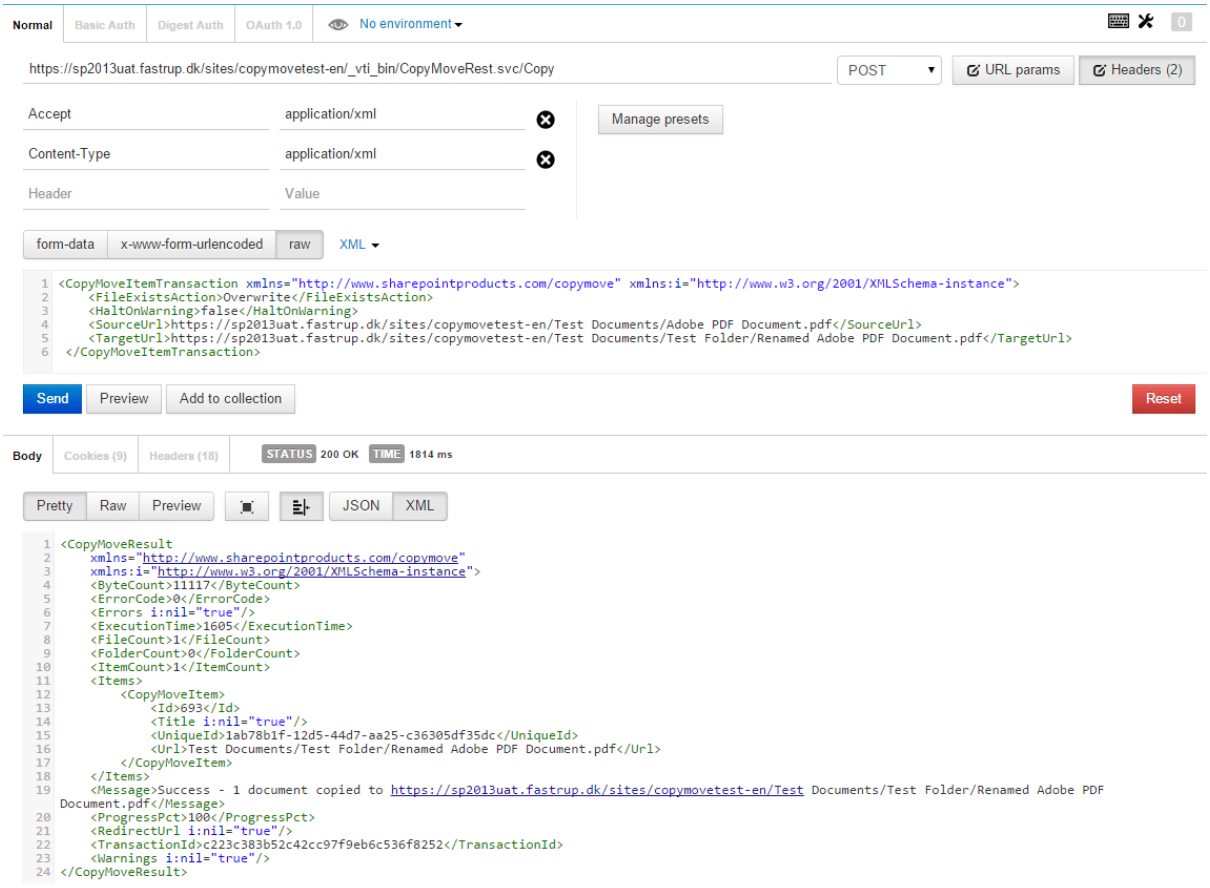
The JSON service has the same end points:

Name	Method	Url
Copy	POST	~site/_vti_bin/CopyMoveJSON.svc/Copy
Move	POST	~site/_vti_bin/CopyMoveJSON.svc/Move
Result	GET	~site/_vti_bin/CopyMoveJSON.svc/Result/{transactionid}
Status	GET	~site/_vti_bin/CopyMoveJSON.svc/Status/{transactionid}
SampleItemTransaction	GET	~site/_vti_bin/CopyMoveJSON.svc/SampleItemTransaction
Settings	GET	~site/_vti_bin/CopyMoveJSON.svc/Settings
Version	GET	~site/_vti_bin/CopyMoveJSON.svc/Version

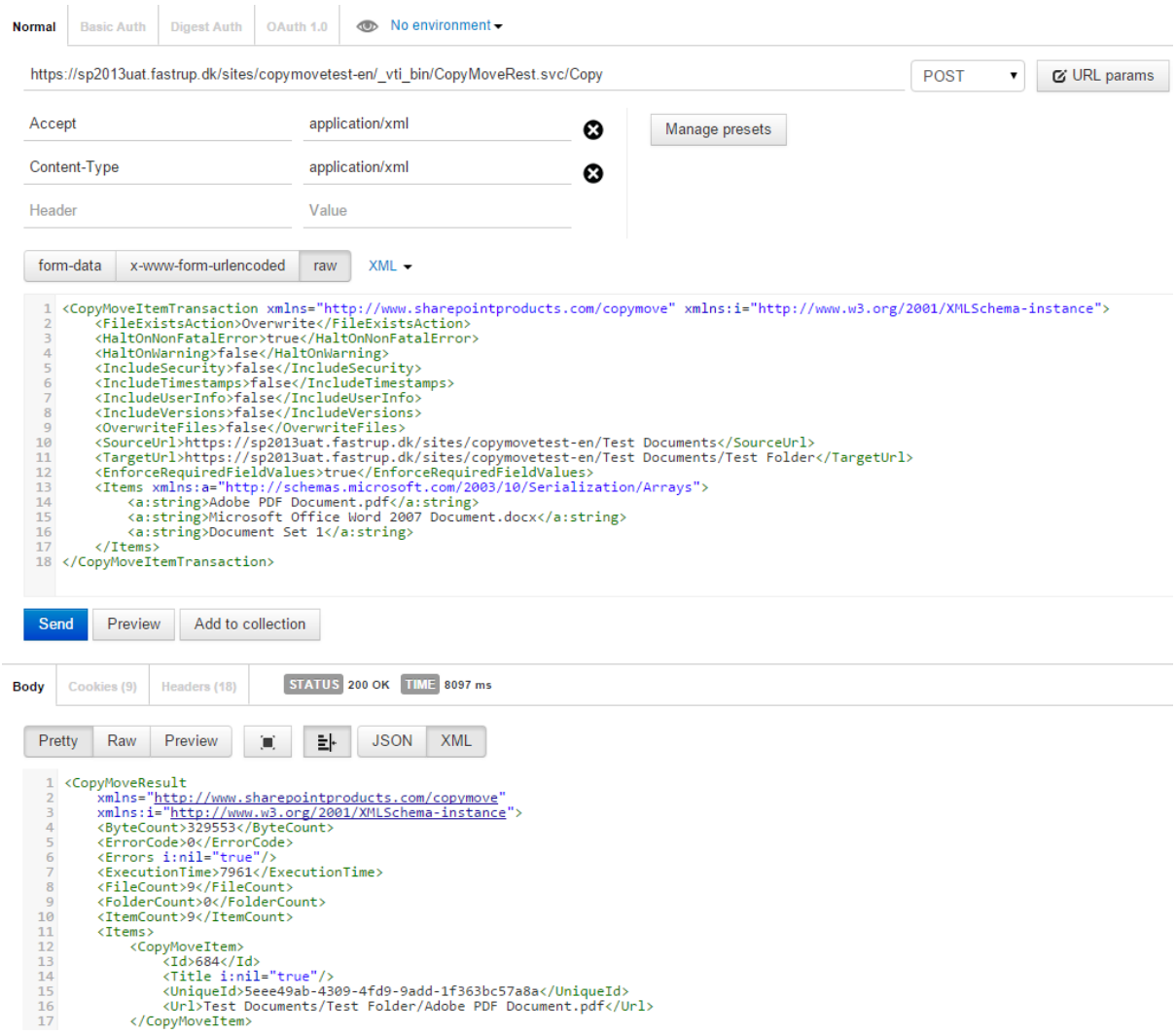
Use the *SampleItemTransaction* endpoint to request a sample XML / JSON structure accepted by the *Copy* and *Move* end points.

© 2008 - 2016 SharePoint Products

Copying a single file only requires a few XML tags as shown in the screen shot below. It shows how to quickly build and test a web request with the Postman app for Google Chrome.



The next Postman screen shot shows how to copy multiple files using the *Items* XML element.



© 2008 - 2016 SharePoint Products

5.1.4 CopyMove JavaScript API

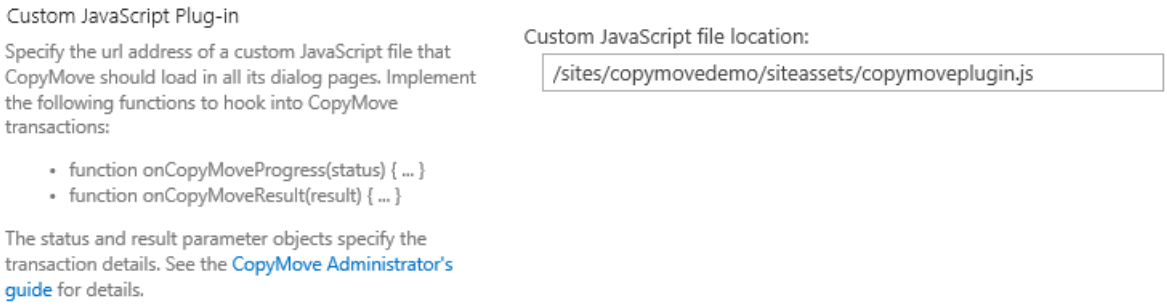
Several clients have over time asked us for a way to plug-in to the CopyMove Web user interface and the transaction flow. To meet these requests, we have in earlier versions of CopyMove 2013 implemented a server-side .NET API, which is still there. However, we are now recommending to use the new client-side JavaScript API presented here. It is easier to work with and does not have a strong coupling to the CopyMove runtime as the server-side API does. The typical scenarios for wanting to plug-in to the CopyMove UI includes:

- Custom validation of the selected destination.
- Redirection to a custom results page.
- Logging of CopyMove transactions to a custom store - e.g to a SharePoint list via the SharePoint JSOM.

To demonstrate how the API works, let us create a skeleton JavaScript file CopyMovePlugin.js with two methods as shown in the code snippet below:

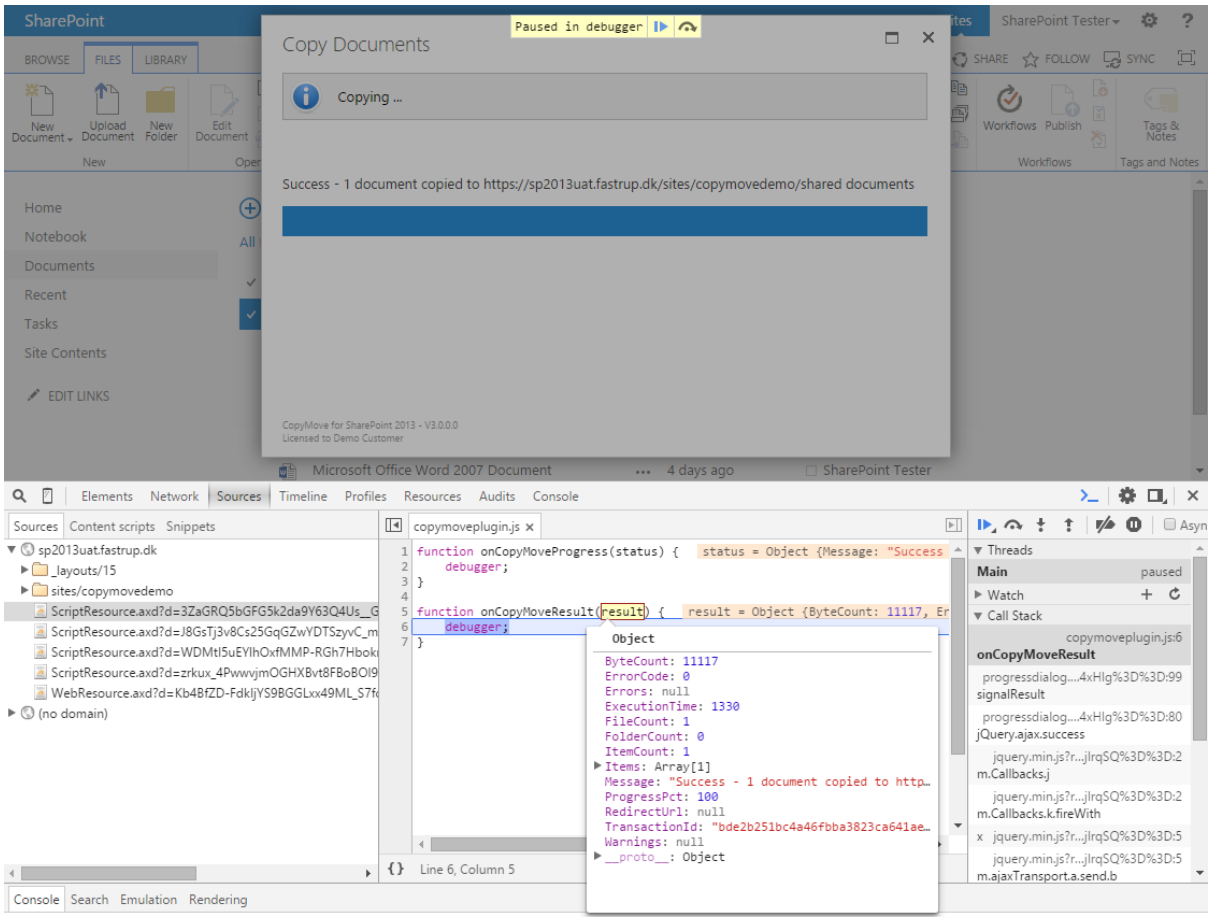
```
function onCopyMoveProgress(status) {
    debugger;
}

function onCopyMoveResult(result) {
    debugger;
}
```

Upload the file to the SiteAssets library and specify the url address on the CopyMove site collection settings page.



Now, open the SharePoint site in the Google Chrome Web Browser and press F12 to open the Developer Tools. Then copy one or more documents with CopyMove. Chrome will in turn break at the *debugger* statement when CopyMove calls the *onCopyMoveProgress* method and the *onCopyMoveResult* method. Hover over the *status/result* variable to inspect the properties of the JavaScript object supplied by CopyMove.

© 2008 - 2016 SharePoint Products



Other Web browsers like Microsoft Internet Explorer, Microsoft Edge and Mozilla FireFox can of course also be used to debug the JavaScript code. However, they do not recognize the debugger keyword as a break point.

© 2008 - 2016 SharePoint Products

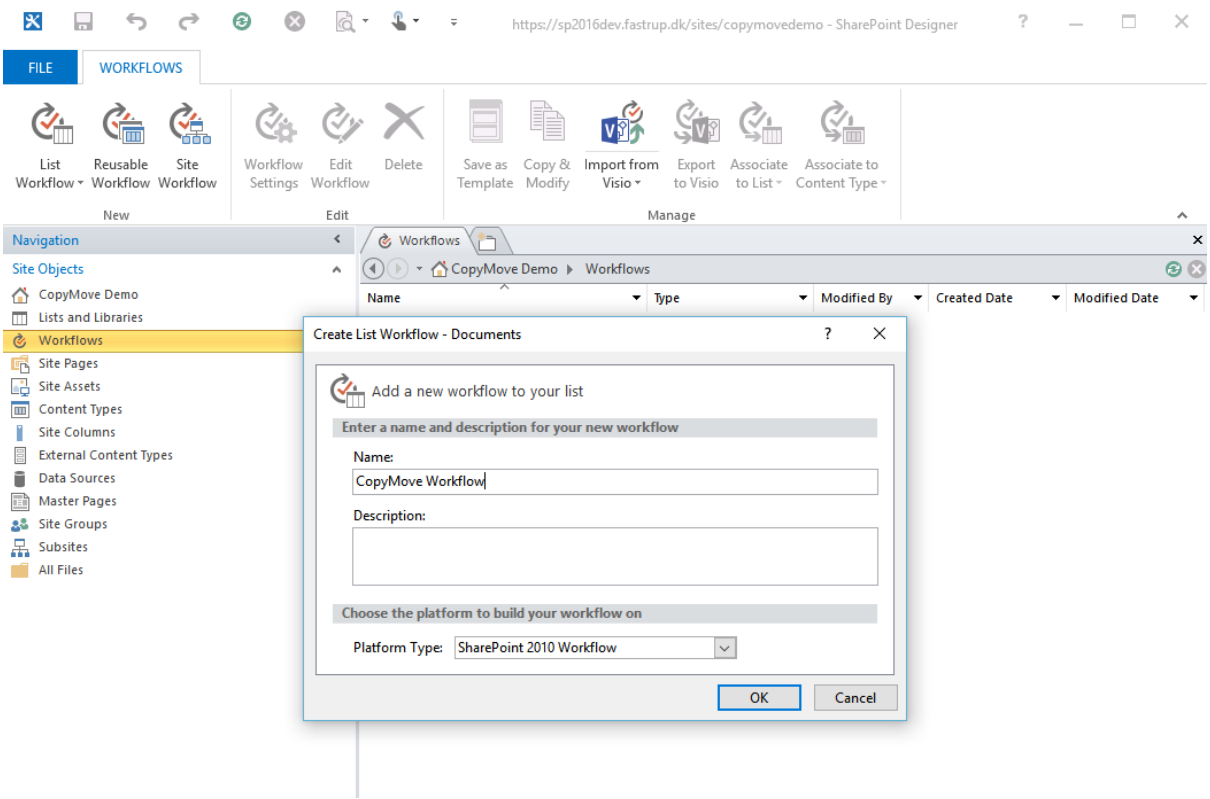
5.2 Using CopyMove in SharePoint 2010 Workflows

SharePoint software developers building SharePoint 2010 type workflows with Visual Studio, can simply leverage the CopyMove .NET API to execute copy, move, export and import operations. Workflows created with SharePoint Designer 2013 can also leverage CopyMove, which installs two workflow actions for the Copy and Move operations. It is in other words possible to build a no-code workflow that copies or moves documents, items and folders with full fidelity. Let us start SharePoint Designer and create a simple SharePoint 2010 type workflow that copies the current item to another location.

1. With SharePoint Designer 2013, open the SharePoint site to create a new workflow in.

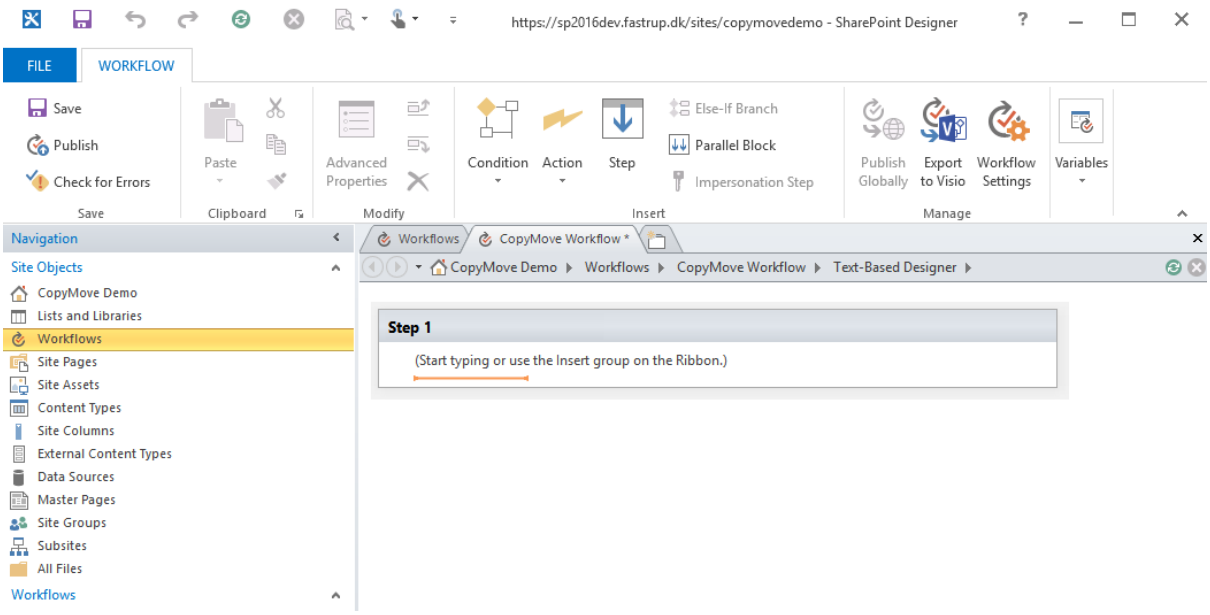
2. Select the *Workflows* node in the left hand navigation pane.

3. Select a document library or list from the *List Workflow* button in the ribbon. This opens the *Create List Workflow* dialog as shown in the following screen shot:

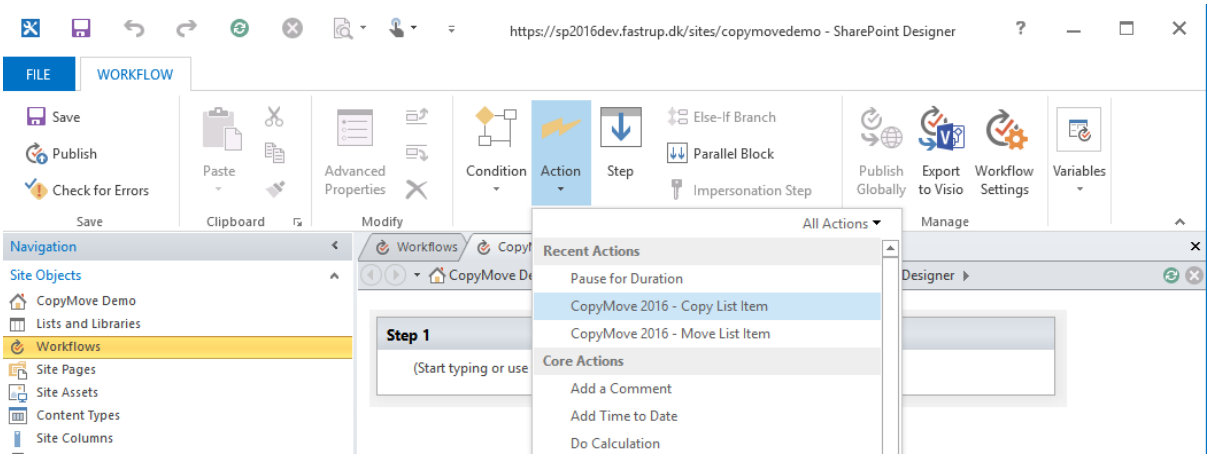


4. Specify a name for the workflow, e.g. CopyMove Workflow and choose *SharePoint 2010 Workflow* as the *Platform Type*. Then click the *OK* button to create a blank workflow as shown below.

© 2008 - 2016 SharePoint Products

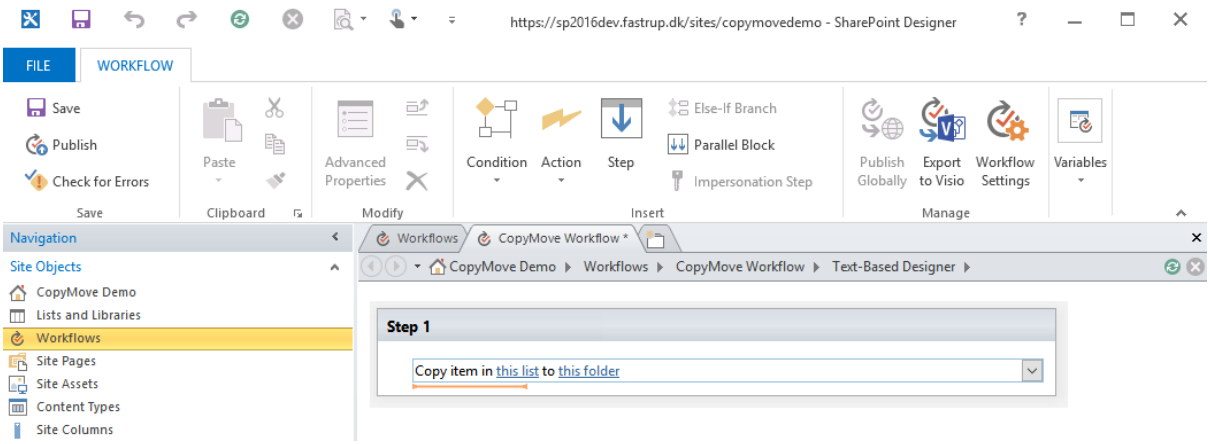


5. We will keep the workflow simple and just design a one that copies the current item as soon as the workflow is started. Select *Step 1* to enable the *Action* button in the Ribbon. Then select the action named *CopyMove 2016 - Copy List Item* to add it to *Step 1* in the workflow.

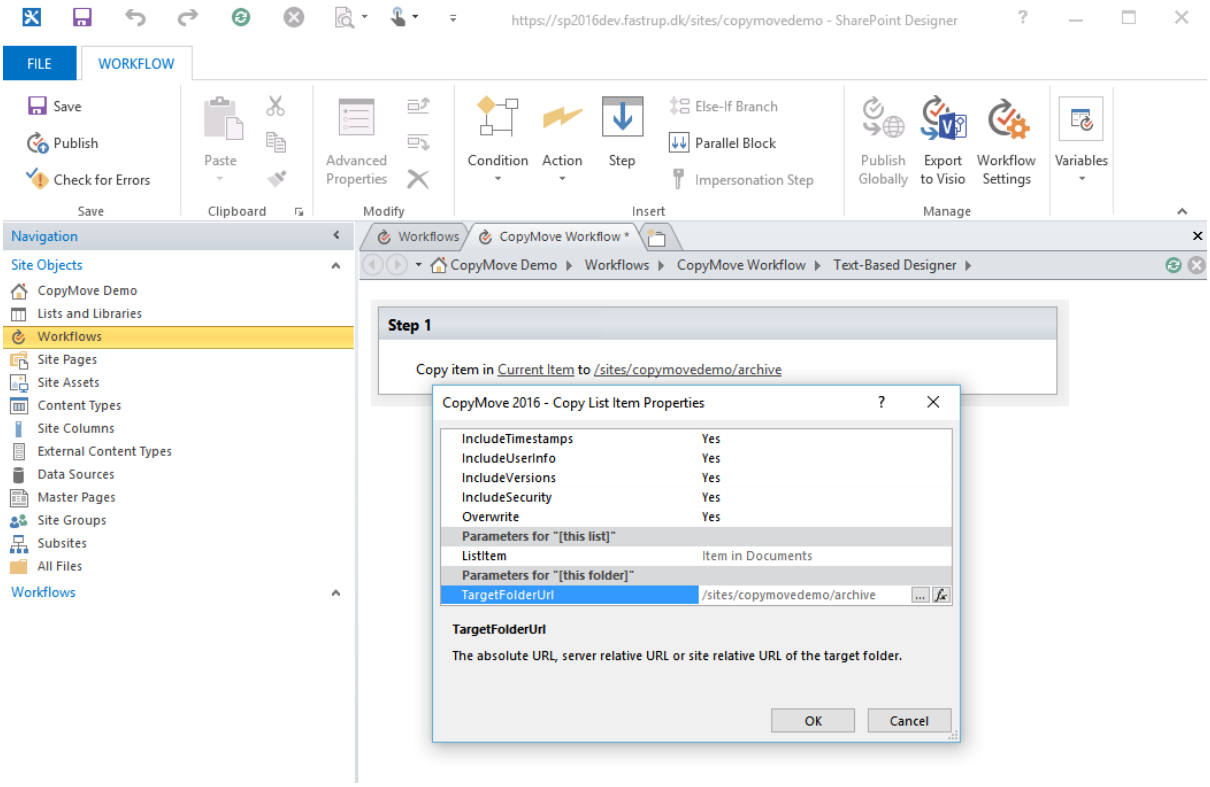


6. The CopyMove action is now added to the workflow and is ready to be configured.

© 2008 - 2016 SharePoint Products



7. Configure the workflow action to copy the *Current Item* to another list or another folder in the same list. Specify the server relative url of the list or folder, e.g. /sites/copymovedemo/archive. Finally, open the properties dialog of the CopyMove workflow action and configure the properties; *IncludeTimestamps*, *IncludeUserInfo*, *IncludeVersions*, *IncludeSecurity* and *Overwrite*.

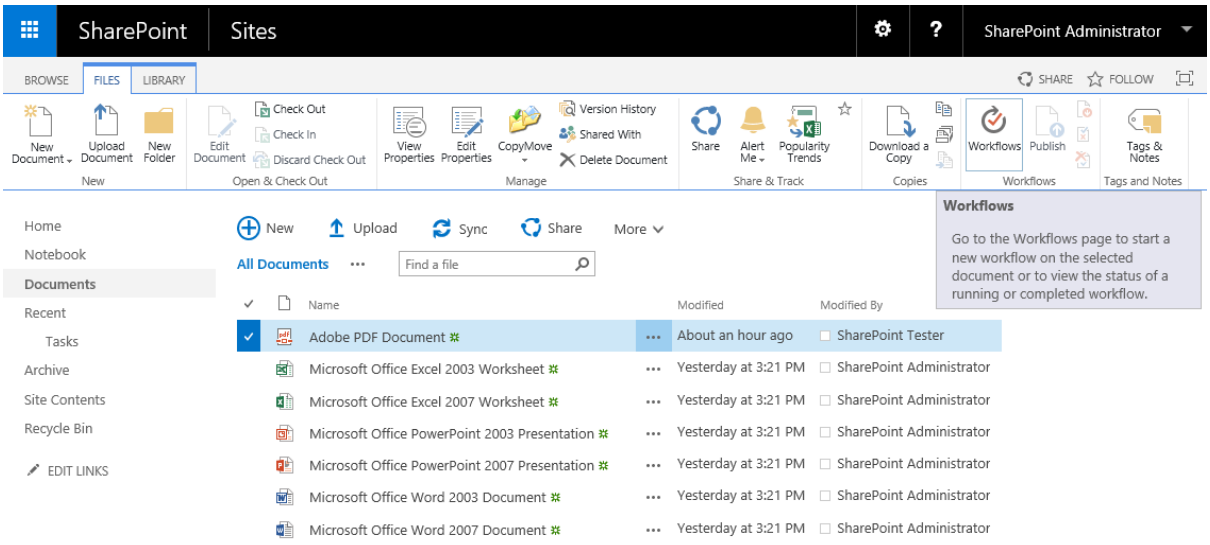


Specify *IncludeTimestamps=Yes* to preserve the created timestamp and the last modified timestamp of the item.
Specify *IncludeUserInfo=Yes* to preserve the created by user and the modified by user.
Specify *IncludeVersions=Yes* to preserve the complete version history of the item.
Specify *IncludeSecurity=Yes* to preserve any item level permissions.
Specify *Overwrite=Yes* to allow existing documents to be overwritten in document libraries.

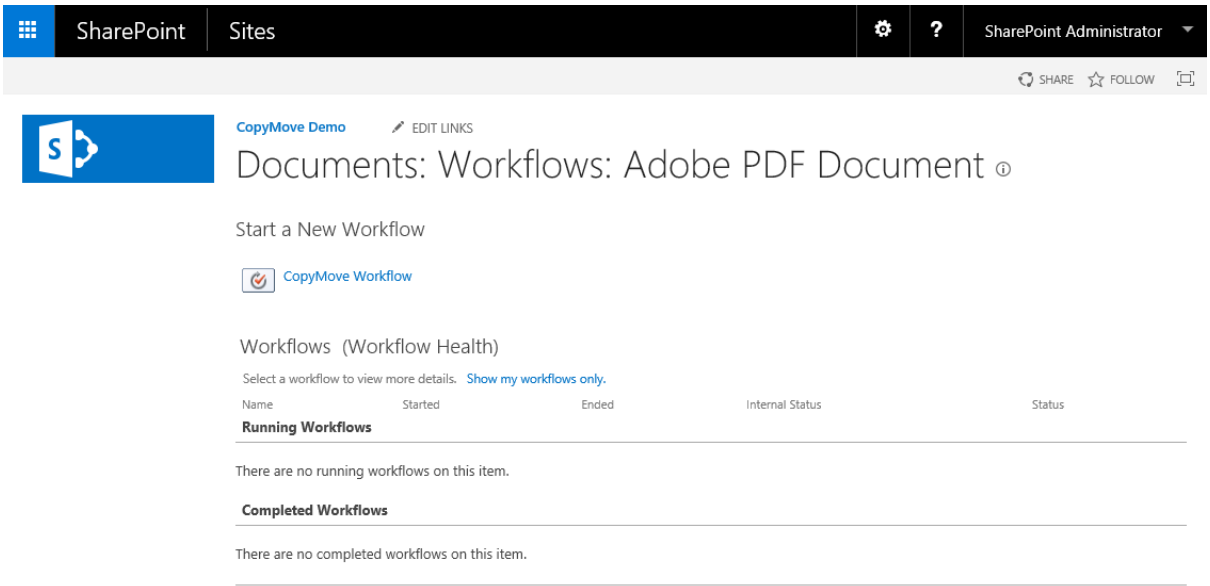
© 2008 - 2016 SharePoint Products

8. The workflow is now ready to be deployed to SharePoint. Verify that it does not contain any errors by clicking the *Check for Errors* button in the ribbon. Then click the Publish button to deploy it to SharePoint.

9. Open the SharePoint site and navigate to the document library or list that the workflow was created for. Then select the item and click the Workflows button in the ribbon as shown below.

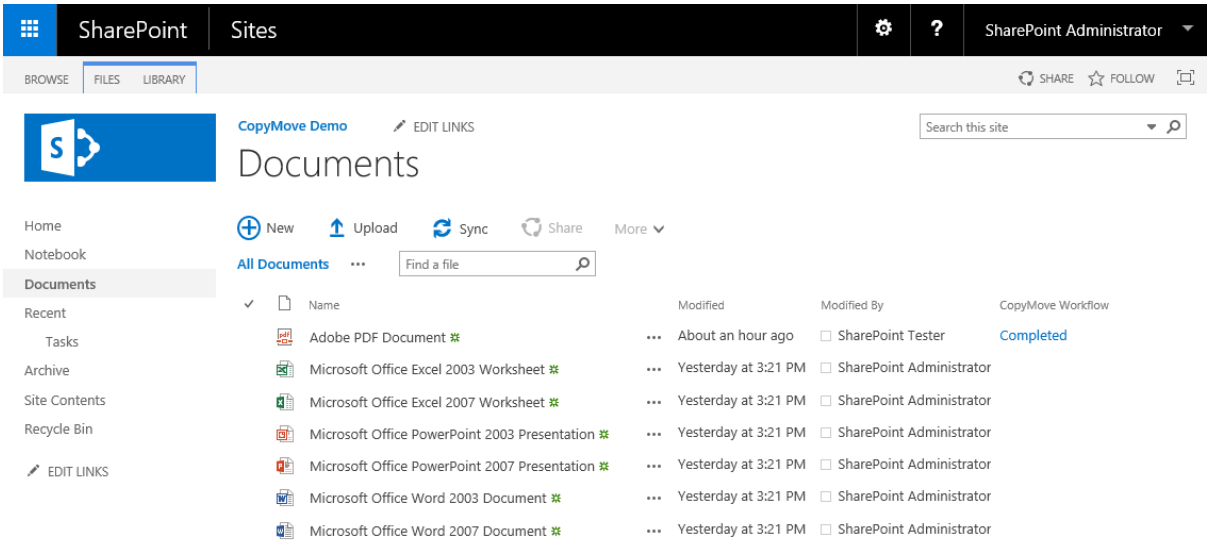


10. Start the workflow by clicking the link the section *Start a New Workflow*.



11. The workflow copies the selected document to the destination location specified for the workflow action in SharePoint Designer. Once the workflow completes, it adds a new status column to the document library as shown in the following screen shot.

© 2008 - 2016 SharePoint Products



12. Done

© 2008 - 2016 SharePoint Products

5.3 Using CopyMove in SharePoint 2013 Workflows

CopyMove 2013 version 3.8.0 and later can be used with the new declarative SharePoint 2013 workflow model. Earlier versions of CopyMove 2013 will fail with an error. Basically, a SharePoint 2013 workflow must use the **Call HTTP Web Service (Http Send)** workflow action to send a Copy or Move Web request to the CopyMove REST Web Service. Specifically, use the endpoint ~site/_vti_vin/CopyMoveJSON.svc/Copy or ~site/_vti_vin/CopyMoveJSON.svc/Move to invoke a Copy or a Move transaction respectively. A transaction has succeed when the resulting HTTP status code is 200 (OK) and the CopyMove error code in the resulting JSON response equals zero. Transactions can of course also fail and can do so in two ways; first it can fail with an HTTP error code other than 200 (OK) and secondly it can also fail with a CopyMove error code in the resulting JSON response. Consequently, workflows should always check both to determine if the transaction completed without errors. The CopyMove error code can be retrieved from the ErrorCode property in the resulting JSON object. It will assume one of the following three integer values:

- 0 = Success
- 1 = Transaction aborted with warning(s)
- 2 = Transaction aborted with error(s)

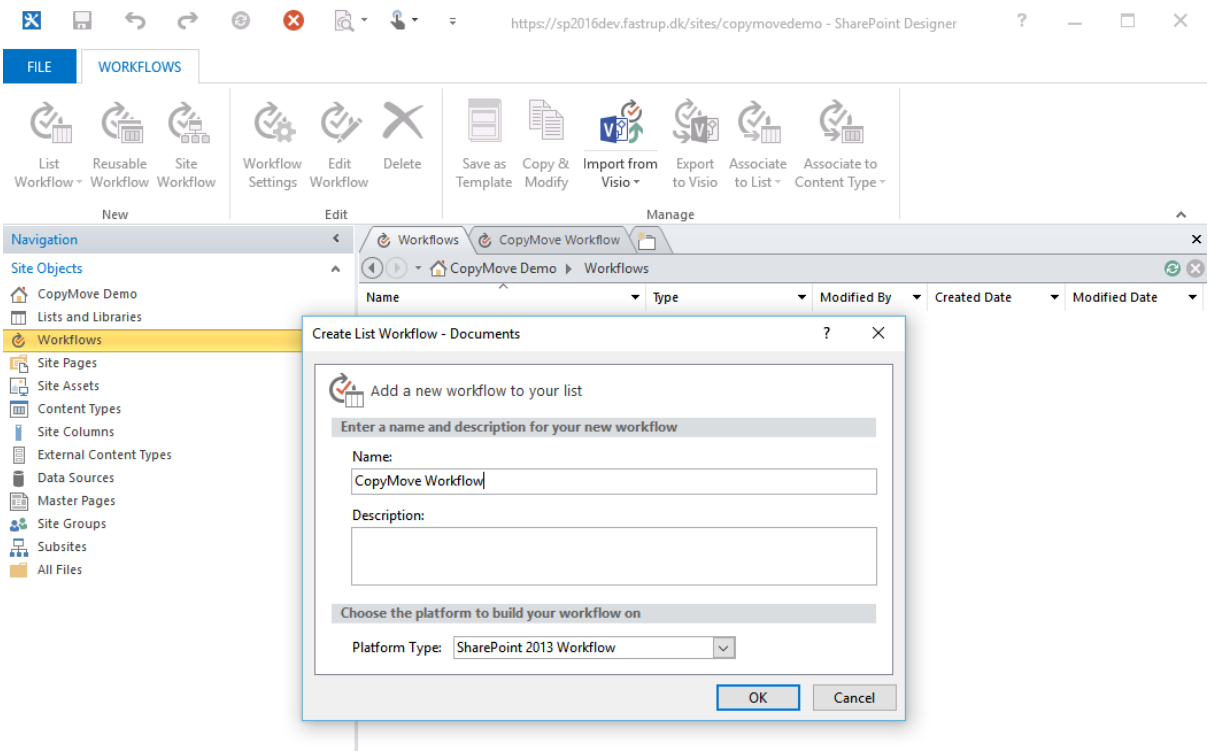
See the API documentation for more details on the CopyMoveResult class which the JSON result represents. To illustrate the process of creating a workflow, let us now try and create a simple SharePoint 2013 type workflow that copies the current document to another document library in the same SharePoint web site.

1. With SharePoint Designer 2013, open the SharePoint site to create a new workflow in.

2. Select the *Workflows* node in the left hand navigation pane.

3. Select a document library or list from the *List Workflow* button in the ribbon. This opens the *Create List Workflow* dialog as shown in the following screen shot:

© 2008 - 2016 SharePoint Products



4. Specify a name for the workflow, e.g. CopyMove Workflow and choose *SharePoint 2013 Workflow* as the *Platform Type*. Then click the *OK* button to create a blank workflow.

5. Rename *Stage 1* to *Copying* and add two more stages named *Success* and *Error*.

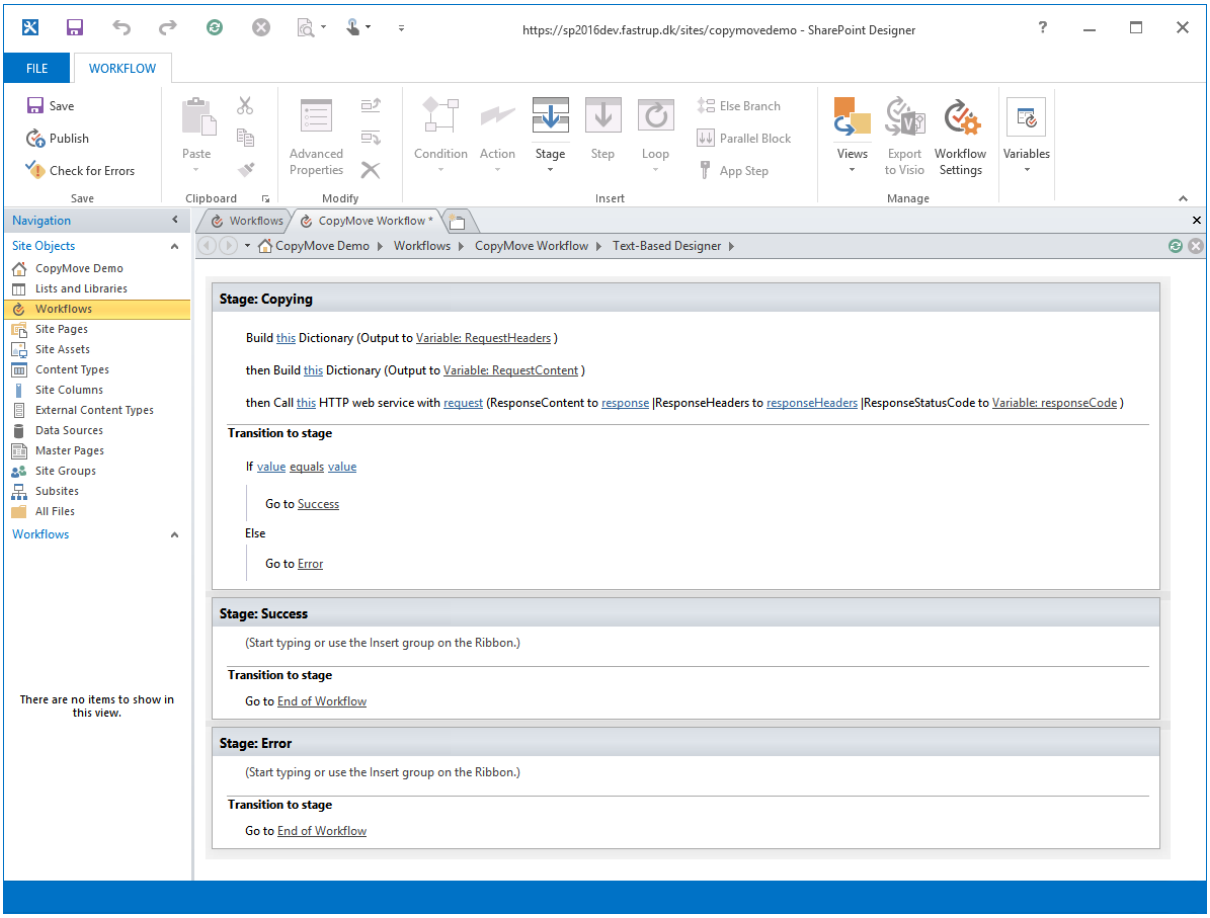
6. In the *Copying* stage, add two *Build Dictionary* actions and let them output to two new dictionary variables named *RequestHeaders* and *RequestContent* respectively.

7. After these, add an instance of the action named *Call HTTP Web Service*.

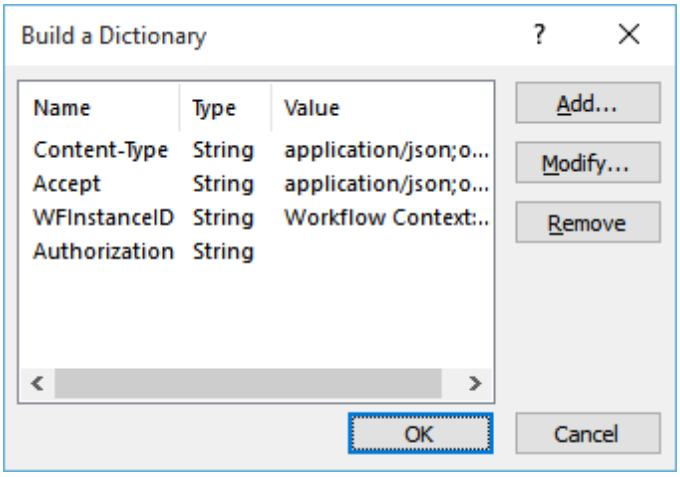
8. Finally, connect the three stages through an IF condition in the *Copying* stage and *Go to Stage* action in the *Success* and *Error* stages. The IF condition should transition to *Success* when it evaluates to true and to *Error* when false. The *Success* and *Error* stages should both transition to the *End of Workflow*.

9. The workflow, should with the completed steps above, now look like this:

© 2008 - 2016 SharePoint Products



10. Next steps is to configure all actions. First, configure the dictionary action that builds the *RequestHeaders* dictionary and add the following entries:

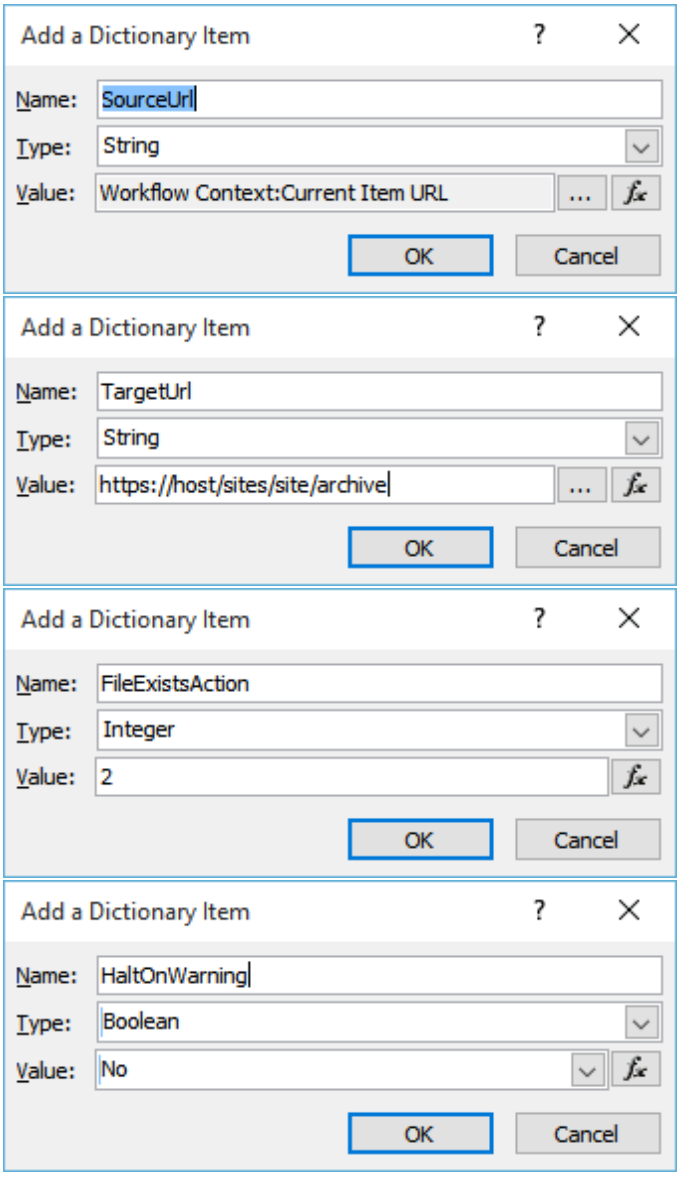


Name	Value	Description
Accept	application/json;odata=verbose	

© 2008 - 2016 SharePoint Products

Name	Value	Description
Content-Type	application/json;odata=verbose	
WFInstanceID	Workflow Context:Instance ID	CopyMove needs the workflow instance id to impersonate the user who initiated the workflow.
Authorization		Empty value, which is critical for the workflow to authenticate with SharePoint.

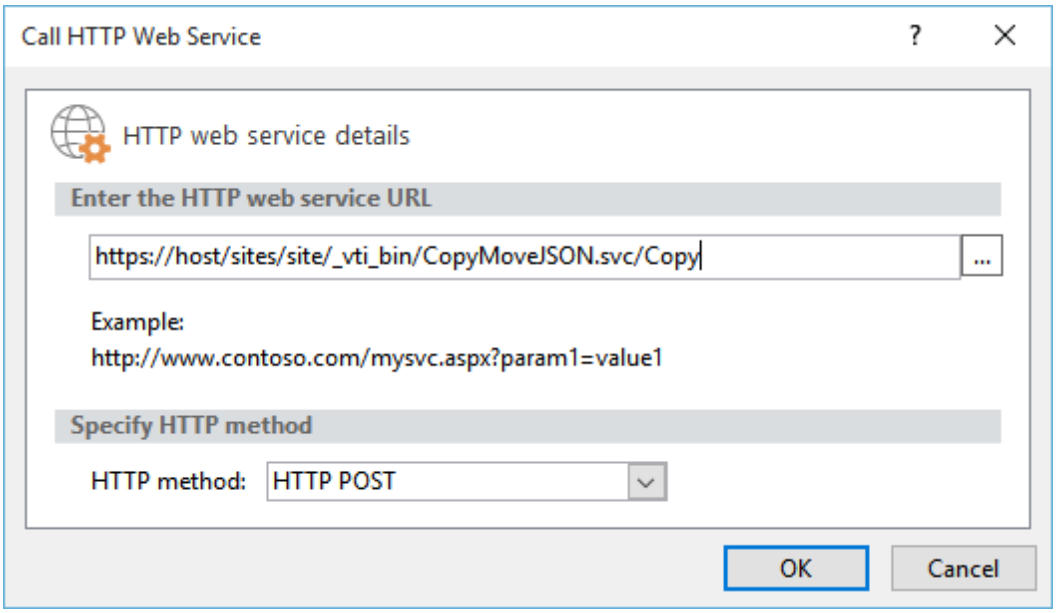
11. Then configure the *RequestContent* dictionary with four entries named *SourceUrl*, *TargetUrl*, *FileExistsAction* and *HaltOnWarning* as shown in these four screen shots from SharePoint Designer 2013:



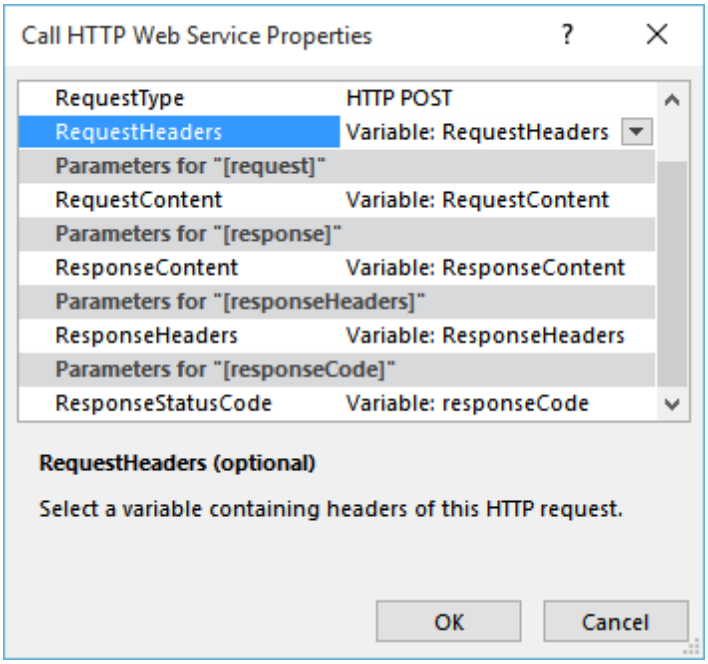
12. Configure the *Call HTTP Web Service* action to call the Copy endpoint at <your site>/_vti_bin/

© 2008 - 2016 SharePoint Products

CopyMoveJSON.svc/Copy as illustrated by this dialog:

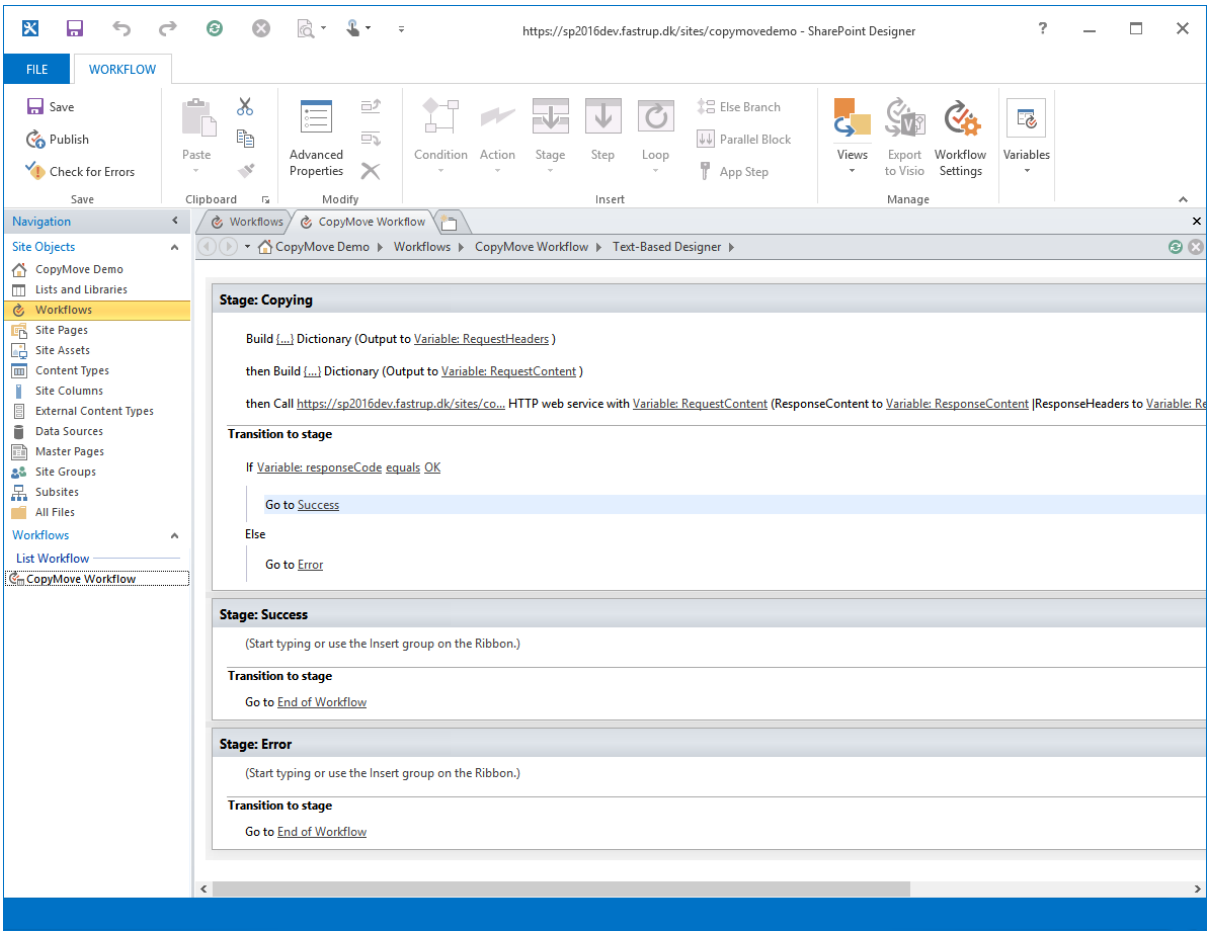


13. Right click the *Call HTTP Web Service* action and select *Properties* to open the Properties dialog. Then link the parameters *RequestHeaders*, *RequestContent*, *ResponseContent* and *ResponseHeaders* to dictionary variables as shown here:



14. Finally, configure the IF condition in the *Transition to Stage* section to evaluate of the variable named *responseCode* equals the string OK. The workflow is now complete and should look as illustrated by the following screen shot.

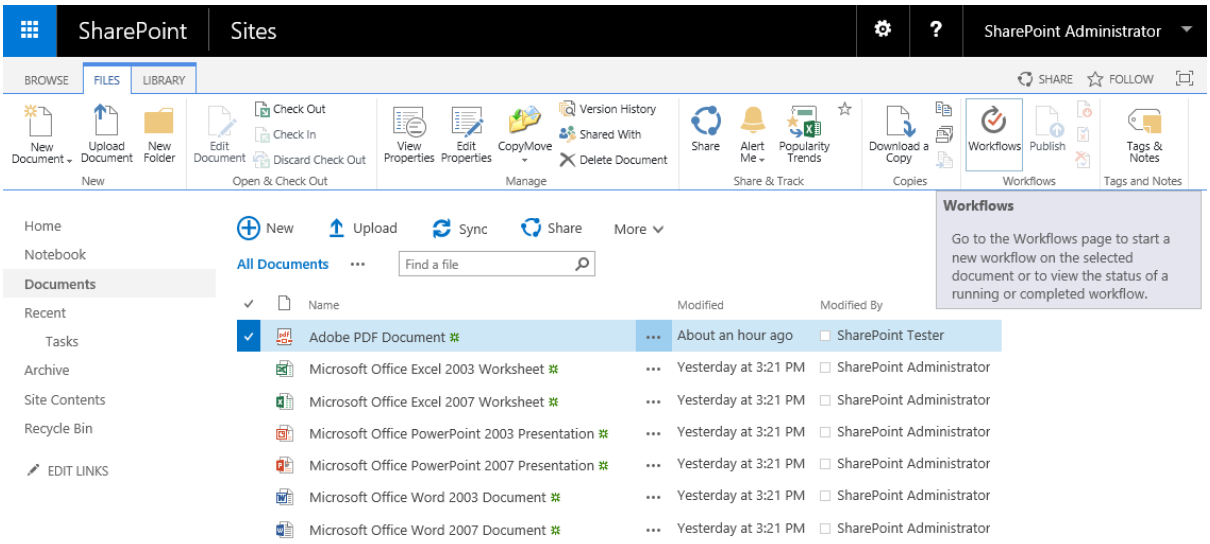
© 2008 - 2016 SharePoint Products



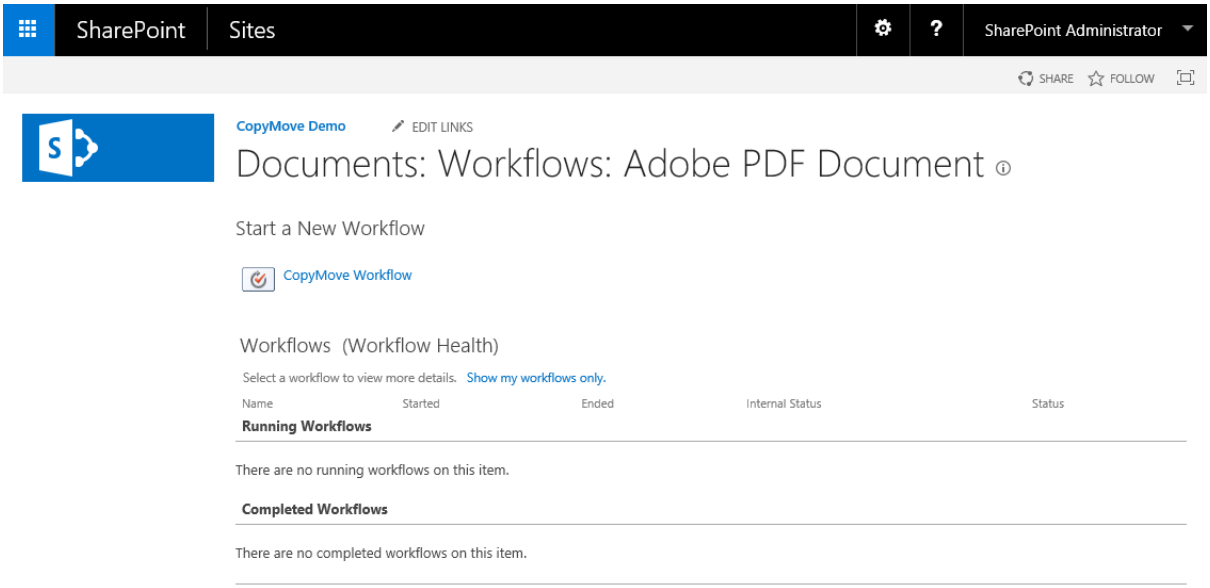
15. Verify that the workflow does not contain any errors by clicking the *Check for Errors* button in the ribbon. Then click the *Publish* button to deploy it to SharePoint.

16. Next, let's test the workflow in SharePoint by opening the SharePoint site and navigate to the document library or list that the workflow was created for. Then select an item and click the *Workflows* button in the ribbon as shown below.

© 2008 - 2016 SharePoint Products

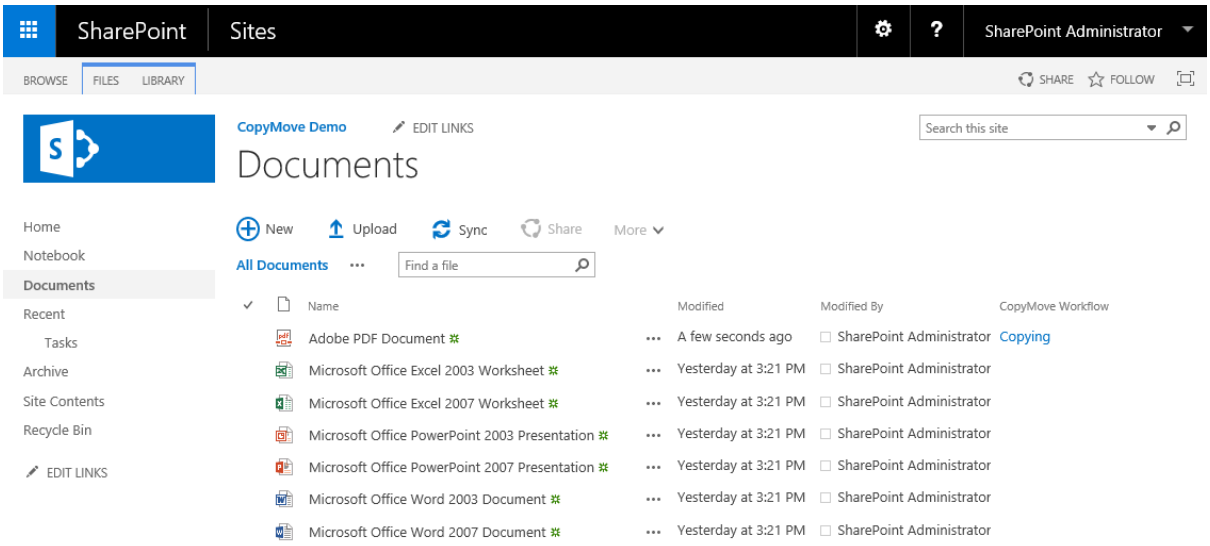


17. Start the workflow by clicking the *CopyMove Workflow* link the section *Start a New Workflow*.

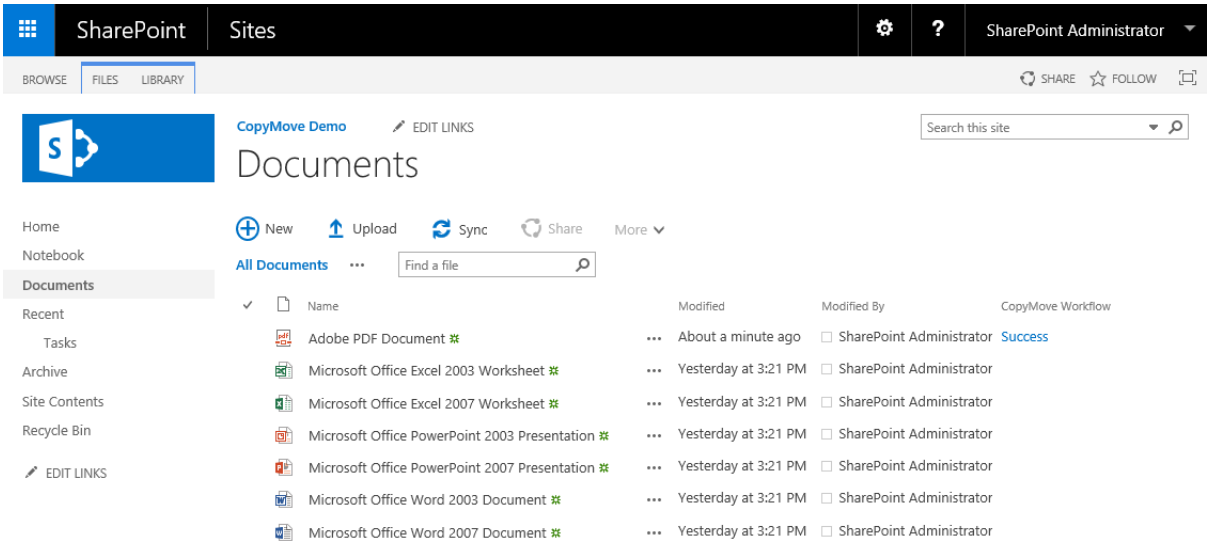


18. Once, the workflow starts it will update the *CopyMove Workflow* column with the name of the currently executing stage. First it will say Copying.

© 2008 - 2016 SharePoint Products



19. When completed - it will show Success or Error.



20. This completes the tutorial on building SharePoint 2013 workflows that leverage the CopyMove 2016 REST API. From here you can extend and optimize the workflow to meet your business requirements.

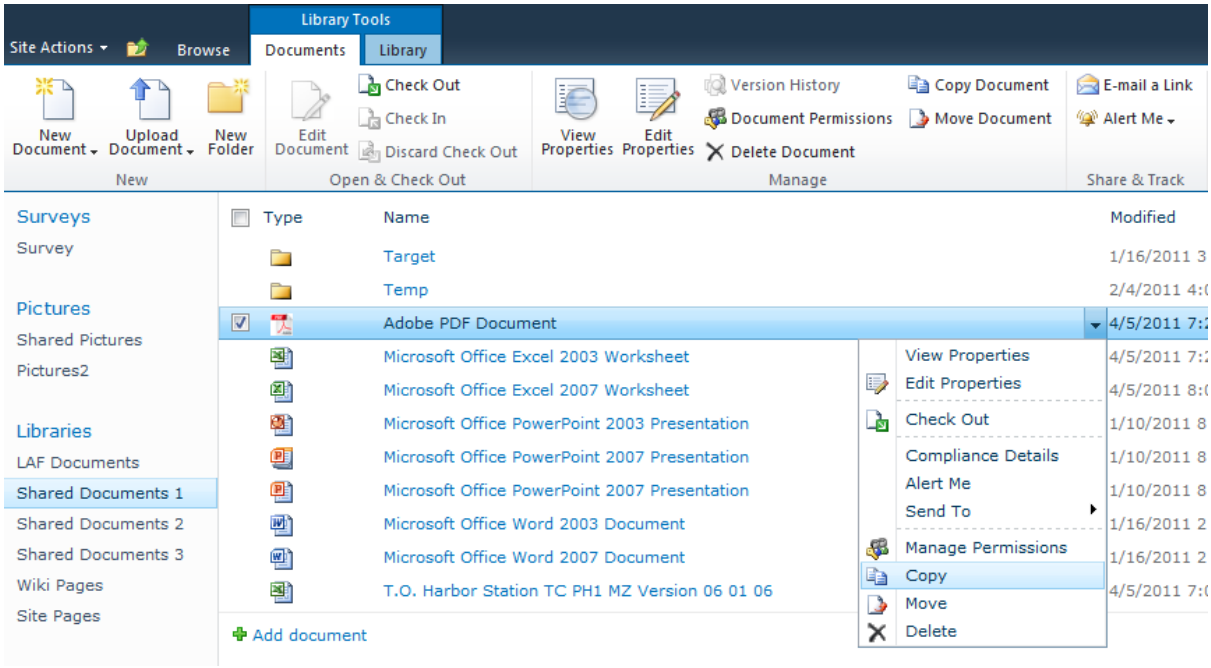
© 2008 - 2016 SharePoint Products

5.4 Registering CopyMove with Custom Lists

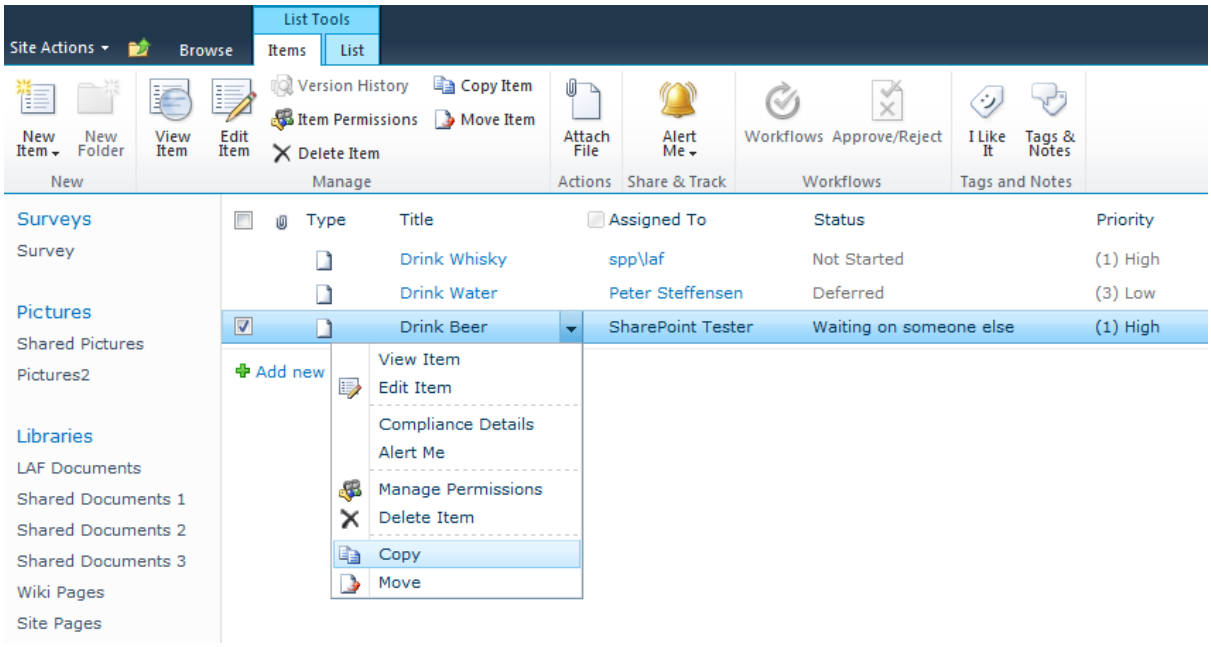
Activating the CopyMove feature as described in the section **Activating the CopyMove Features** registers the Copy and Move actions with document libraries and lists created from standard SharePoint list templates. See the section **Supported List Types** for a complete overview of the list types that CopyMove support out of the box.

Document libraries and lists created from custom list templates will not integrate with CopyMove without some customization as outlined in this section. This is by design as SharePoint requires custom UI actions in a list view to be registered independently for each list type. It is simply not possible to register custom actions in one go for all list types. Consequently, you will need to register the CopyMove UI actions for each custom list template that CopyMove must also integrate with. The recommended way to do this is to register the actions via a custom SharePoint feature deployed through a SharePoint 2016 solution package created with Visual Studio 2015. The CopyMove 2016 product download includes a sample Visual Studio project named *Custom List Template* that registers CopyMove with a custom document library template and a custom list template.

The *Custom List Template* sample demonstrates how to register CopyMove with a custom document library template and a custom list template as illustrated by the following two screen shots respectively. The sample registers a *Copy* action and a *Move* action with the ribbon and with the item drop-down menu.



© 2008 - 2016 SharePoint Products



To register the *Copy* action with a custom document library template, simply reference the XML snippet below in a SharePoint feature as shown by the *Custom List Template* sample project. The <*CustomAction*> element registers the *Copy* action with the ribbon in two locations; the first <*CommandUIDefinition*> element registers it with the ribbon in the *View Item* form and the second <*CommandUIDefinition*> element registers it with the ribbon in document library views.

```
<CustomAction
  Id="CustomDocumentLibrary.Ribbon.Copy"
  RegistrationId="101" RegistrationType="List"
  Location="CommandUI.Ribbon" Sequence="5"
  Rights="ViewListItems" UIVersion="4">
  <CommandUIExtension>
    <CommandUIDefinitions>
      <CommandUIDefinition
        Location="Ribbon.ListForm.Display.Manage.Controls._children">
        <Button
          Id="SCustomDocumentLibrary.DispForm.CopyButton"
          Command="CustomDocumentLibrary.CopyDocument"
          Image32by32="/_layouts/15/images/spp/Copy32.gif"
          Image16by16="/_layouts/15/images/copy16.gif"
          LabelText="Copy Item" ToolTipTitle="Copy Item"
          ToolTipDescription="Copy the selected items."
          Sequence="100" TemplateAlias="o2" />
      </CommandUIDefinition>
      <CommandUIDefinition
        Location="Ribbon.Documents.Manage.Controls._children">
        <Button
          Id="CustomDocumentLibrary.AllItems.CopyButton"
          Command="SharePointProducts.CopyMove.CustomDocumentLibrary.CopyDocuments"
          Image32by32="/_layouts/15/images/spp/Copy32.gif"
          Image16by16="/_layouts/15/images/copy16.gif"
          LabelText="Copy Document" ToolTipTitle="Copy Document"
          ToolTipDescription="Copy the selected documents."
          Sequence="100" TemplateAlias="o2" />
```

© 2008 - 2016 SharePoint Products

```
      </CommandUIDefinition>
    </CommandUIDefinitions>
    <CommandUIHandlers>
      <CommandUIHandler
        Command="CustomDocumentLibrary.CopyDocument"
        CommandAction="javascript:SPP_CopyItem('{SiteUrl}','Copy Document')" />
      <CommandUIHandler
        Command="CustomDocumentLibrary.CopyDocuments"
        CommandAction="javascript:SPP_CopyItems('{SiteUrl}','Copy Documents')"
        EnabledScript="javascript:SPP_EnableCopyCommand()" />
    </CommandUIHandlers>
  </CommandUIExtension>
</CustomAction>
```

Similarly, use the following XML to register the *Move* action:

```
<CustomAction
  Id="CustomDocumentLibrary.Ribbon.Move"
  RegistrationId="101" RegistrationType="List"
  Location="CommandUI.Ribbon" Sequence="5"
  Rights="ViewListItems" UIVersion="4">
  <CommandUIExtension>
    <CommandUIDefinitions>
      <CommandUIDefinition
        Location="Ribbon.ListForm.Display.Manage.Controls._children">
        <Button
          Id="CustomDocumentLibrary.DispForm.MoveButton"
          Command="CustomDocumentLibrary.MoveDocument"
          Image32by32="/_layouts/15/images/spp/Move32.gif"
          Image16by16="/_layouts/15/images/upload.gif"
          LabelText="Move Item" ToolTipTitle="Move Item"
          ToolTipDescription="Move the selected items."
          Sequence="101" TemplateAlias="o2" />
      </CommandUIDefinition>
      <CommandUIDefinition
        Location="Ribbon.Documents.Manage.Controls._children">
        <Button
          Id="CustomDocumentLibrary.AllItems.MoveButton"
          Command="CustomDocumentLibrary.MoveDocuments"
          Image32by32="/_layouts/15/images/spp/Move32.gif"
          Image16by16="/_layouts/15/images/upload.gif"
          LabelText="Move Document" ToolTipTitle="Move Document"
          ToolTipDescription="Move the selected documents."
          Sequence="101" TemplateAlias="o2" />
      </CommandUIDefinition>
    </CommandUIDefinitions>
    <CommandUIHandlers>
    <CommandUIHandler
      Command="CustomDocumentLibrary.MoveDocument"
      CommandAction="javascript:SPP_MoveItem('{SiteUrl}','Move Document')"/>
    <CommandUIHandler
      Command="CustomDocumentLibrary.MoveDocuments"
      CommandAction="javascript:SPP_MoveItems('{SiteUrl}','Move Documents')"
      EnabledScript="javascript:SPP_EnableMoveCommand()" />
    </CommandUIHandlers>
  </CommandUIExtension>
</CustomAction>
```

© 2008 - 2016 SharePoint Products

Adapt the XML as follows for each custom document library template that you need to support:

1. Update the *RegistrationId* attribute to match the template id of your custom document library template.
2. Update the *Id* attributes with unique values across templates.
3. Update the *Command* attributes with unique values across templates.

Finally, be aware that the custom actions outlined above rely on the availability of several Javascript methods including; *SPP_CopyItem, SPP_CopyItems, SPP_MoveItem, SPP_MoveItems, SPP_EnableCopyCommand and SPP_EnableMoveCommand*. The methods will be available if you have activated the CopyMove site collection feature or the CopyMove Web application feature. But should you decide to only activate CopyMove through a custom feature then the following XML must also be referenced in the feature to include the CopyMove Javascript in Web pages.

```
<CustomAction ScriptSrc="SPP/Platform.js" Location="ScriptLink" Sequence="100">
</CustomAction>
<CustomAction ScriptSrc="SPP/CopyMove.js" Location="ScriptLink" Sequence="200">
</CustomAction>
```

A good source for further inspiration is the feature XML files deployed by CopyMove to the following locations on each farm server:

```
C:\Program Files\Common Files\Microsoft Shared\Web Server Extensions\15
\TEMPLATE\FEATURES\SPPCopyMoveSite
C:\Program Files\Common Files\Microsoft Shared\Web Server Extensions\15
\TEMPLATE\FEATURES\SPPCopyMoveWebApp
```

Here you can see exactly how CopyMove registers with all the list types listed in the **Supported List Types** section.

© 2008 - 2016 SharePoint Products

5.5 Detecting CopyMove in SharePoint Event Receivers

Custom event receivers or third-party event receivers in SharePoint lists will conflict with CopyMove if they make any updates in asynchronous events like ItemAdded and ItemUpdated before CopyMove is done with the item. This is especially true when copying or moving documents and items with more than one version. CopyMove does have some retry logic to recover from conflicts - but it does not cover all possible scenarios. To avoid conflicts, custom event receivers must implement CopyMove awareness to defer all work until CopyMove has imported the last document or item version.

The CopyMove awareness can be implemented by monitoring the value of the item property named *SPPCopyMoveEvent* that CopyMove injects into property bag of each list item version being imported. However, the property injection is not enabled by default. It must first be enabled with the following PowerShell command in a SharePoint Management Shell on the SharePoint server:

```
Set-CopyMoveSettings -InjectEventProperties
```

Once enabled, CopyMove will inject the property to all items being copied and moved. The property can assume the different values shown in the table below.

Value	Events	Description
0	ItemAdded ItemCheckedIn	File was copied (last version will have this value)
1	ItemAdded ItemCheckedIn	File was moved (last version will have this value)
2	ItemUpdated	Move in progress (more versions to be imported)
3	ItemUpdated	Copy in progress (more versions to be imported)
4	ItemUpdated	The very last event after a copy
5	ItemUpdated	The very last event after a move

Code wise, the *SPPCopyMoveEvent* property can be monitored as shown in the following C# code sample.

```
public class SampleEventReceiver : SPItemEventReceiver
{
    public override void ItemAdded(SPItemEventProperties properties)
    {
        HandleCopyMoveEvent(properties.AfterProperties);
    }

    public override void ItemUpdated(SPItemEventProperties properties)
    {
        HandleCopyMoveEvent(properties.AfterProperties);
    }

    private void HandleCopyMoveEvent(SPItemEventDataCollection properties)
    {
        object sppCopyMoveEventObj = properties["SPPCopyMoveEvent"];
        if (sppCopyMoveEventObj != null)
        {
```

© 2008 - 2016 SharePoint Products

```
            int copyMoveEventId;
            if (Int32.TryParse(sppCopyMoveEventObj.ToString(), out copyMoveEventId))
            {
                HandleCopyMoveEvent(copyMoveEventId);
            }
        }
    }

    private void HandleCopyMoveEvent(int copyMoveEventId)
    {
        switch (copyMoveEventId)
        {
            case 0:   // Copying last item version
                break;

            case 1:   // Moving last item version
                break;

            case 2:   // Move in progress - more versions to come.
                break;

            case 3:   // Copy in progress - more versions to come.
                break;

            case 4:   // Copy complete.
                break;

            case 5:   // Move complete.
                break;

        }
    }
}
```

© 2008 - 2016 SharePoint Products

Part VI

Troubleshooting

6 Troubleshooting

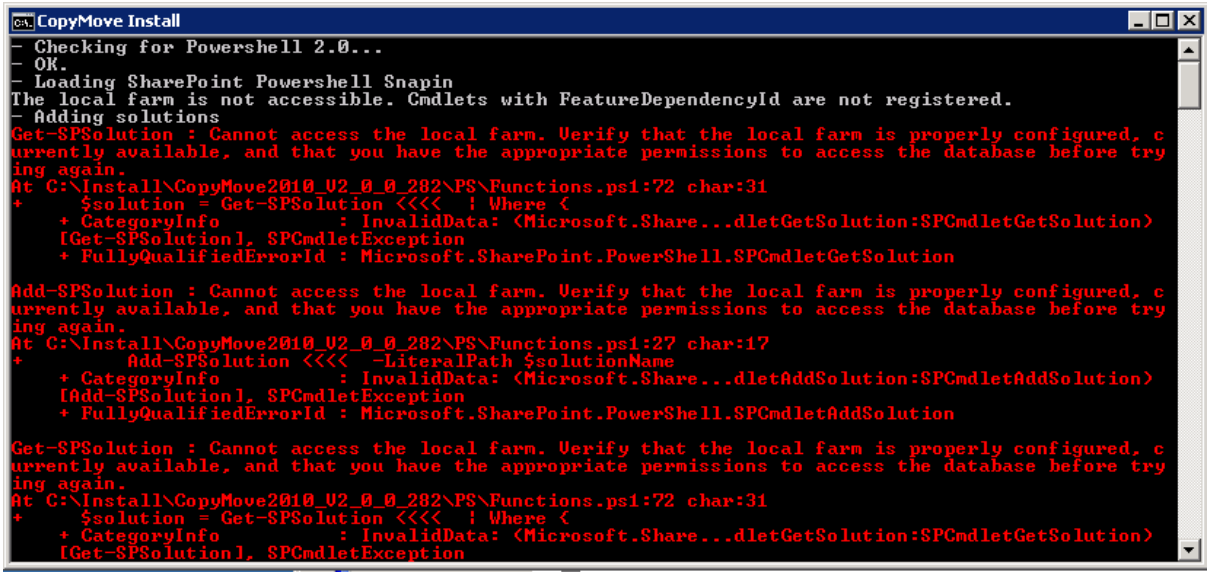
SharePoint is a very complex platform to setup and operate as well as it is a complex platform to develop applications for. CopyMove for SharePoint 2016 has been tested rigorously with SharePoint Server 2016 in different configurations. But this does unfortunately not mean that you will never encounter any problems when installing and using CopyMove. Most problems are related to the way SharePoint is configured as well as lack of technical insight into the SharePoint product (it simply takes time, skill and effort to really understand and master SharePoint). On rare occasions, you might encounter a genuine bug in CopyMove when it is faced with special conditions that we have not tested for. In this case, please report the problem and we will do our best to understand, reproduce and fix the problem for you.

For the reasons above, we strongly recommend to evaluate and test CopyMove in a test environment before deploying it to a production environment. The test environment should be a separate SharePoint farm in a configuration that resembles that of the production farm.

This chapter seeks to help you understand and correct common problems. It also provides a guideline for reporting problems to our support team when you encounter a problem that you cannot resolve yourself.

6.1 Installation Error - Cannot access the local farm

The error *Cannot access the local farm* shows when trying to install CopyMove to a SharePoint farm. The error causes the installation script to abort as shown in the screen shot below.



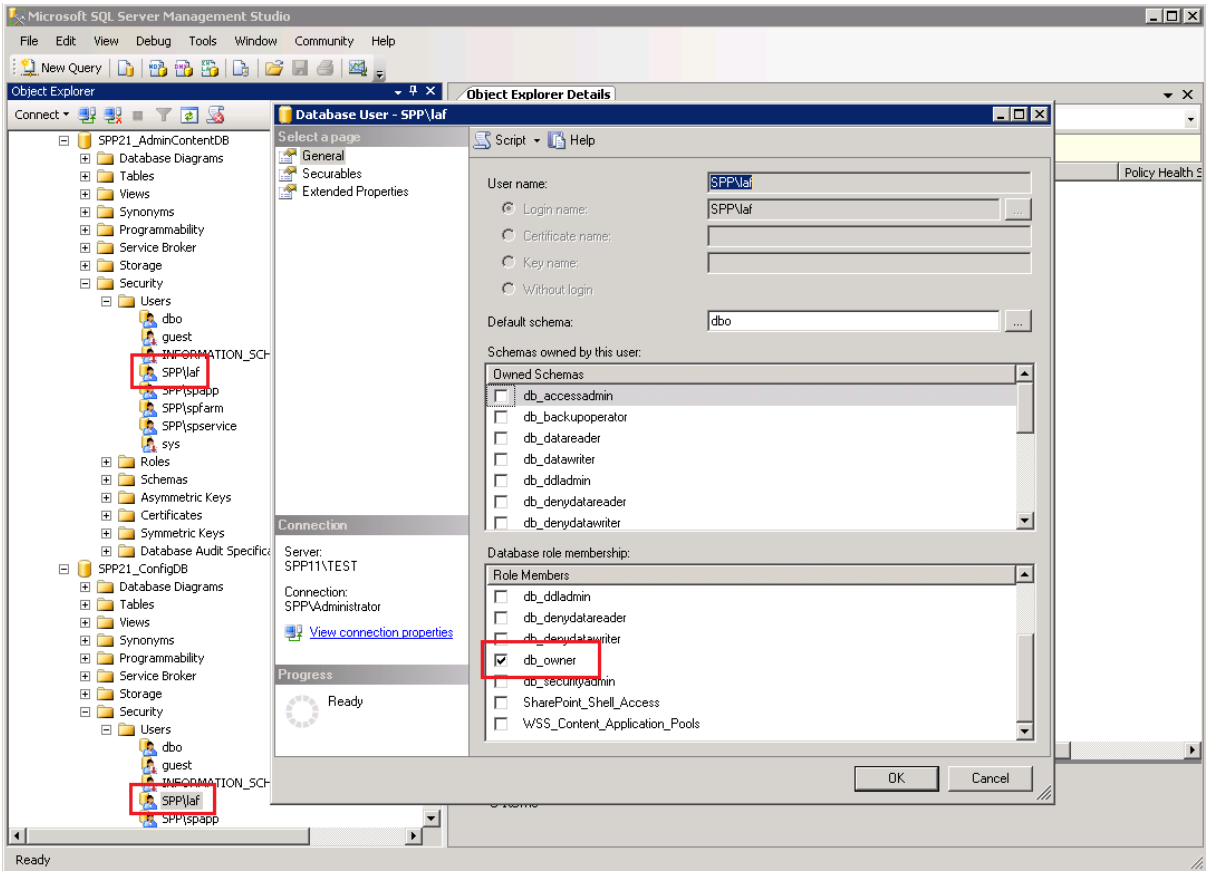
CAUSE
You have logged in to the SharePoint server with a Windows account that does not have full access to the SharePoint configuration database and the SharePoint administration content database. This is required by SharePoint in order to add and deploy solution packages in the farm.

RESOLUTION

© 2008 - 2016 SharePoint Products

Log out from the SharePoint server and log in again with the credentials of a user having full access to the SharePoint configuration database and the SharePoint administration content database. If possible, use the same account as used by the IIS application pool account for the SharePoint Central Administration Web site. Alternatively, login as domain administrator if you have the password to this account.

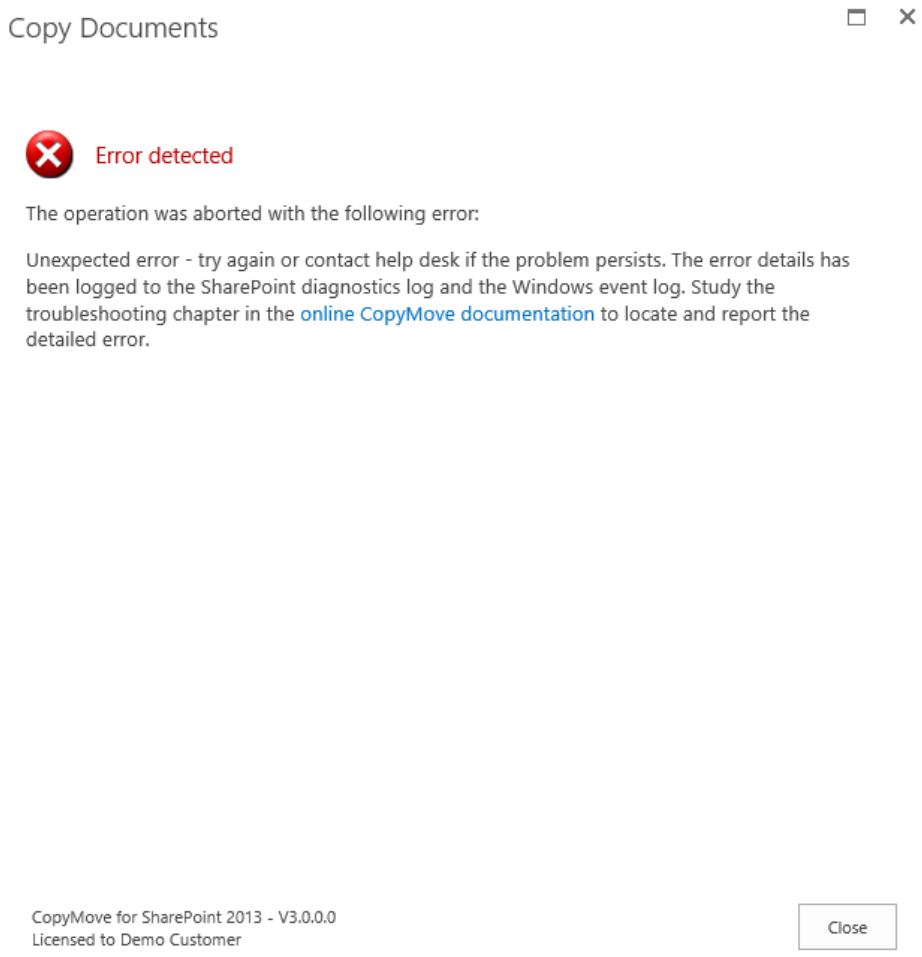
Finally, you can also grant access to the appropriate account using the Microsoft SQL Server Management Studio as illustrated below. However, this operation requires administrative access to the SharePoint databases on the SQL Server. Add the account to the list of users in each of the two databases and grant it the *db_owner* role.



© 2008 - 2016 SharePoint Products

6.2 Unexpected error in the CopyMove dialog

Users experience the following error when trying to copy or more one or more selected items.



CAUSE
This is an unexpected error and could be caused by a SharePoint configuration problem or by a bug in CopyMove. The error message does for security concerns not reveal system details to end users.

RESOLUTION
Inspect the SharePoint diagnostics log or the Windows event log for more details about the error. See the **Diagnostic Logging** section for more information how to configure and inspect the log. You can also update the web.config file as follows to get CopyMove to display the detailed error directly in the dialog instead of the general error message.

1. Open the web.config file for the affected Web application.
2. Change the attribute **CallStack="false"** to **CallStack="true"** on the **<SafeMode>** element.
3. Change **<customErrors mode="On" />** to **<customErrors mode="Off" />**

© 2008 - 2016 SharePoint Products

If the detailed error information cannot help you understand and resolve the problem then contact CopyMove support for assistance. See the section **How to report a problem to the CopyMove support team**.

6.3 How to report a problem to the CopyMove support team

If you as a SharePoint administrator encounter any undocumented problems with CopyMove then you can report it to our support team who will do their best to help you. New support cases can be opened by email to support@sharepointproducts.com or by submitting the support form at www.sharepointproducts.com/support. We do not offer phone support and conference calls with desktop sharing before strictly necessary. All support cases must start with a written description of the problem.

To help us understand and reproduce the problem, please provide as much relevant information as possible. Please do not just state that it does not work or that you get this and that error message. We will in turn just ask you for more information causing a delay in the resolution process. Study the following check list to see what kind of information you can include in the support email:

1. Description of the expected behavior.

2. Description and screen shots of the unexpected behavior.

3. Answer to the question: "Can you reproduce the problem anywhere else in SharePoint or is it restricted to a specific site, file, list or document library?"

4. Answer to the question: "Can you reproduce the problem in all your SharePoint environments including Development, Test and Production?"

5. Description of all the steps required to reproduce the problem.

6. Description and screen shots of relevant configuration settings, like Content Types and Columns in a document library.

7. If possible and relevant, a copy of the document library or list where CopyMove fails to copy or move one or more items. Save the list as a template and include the .STP file in the support email.

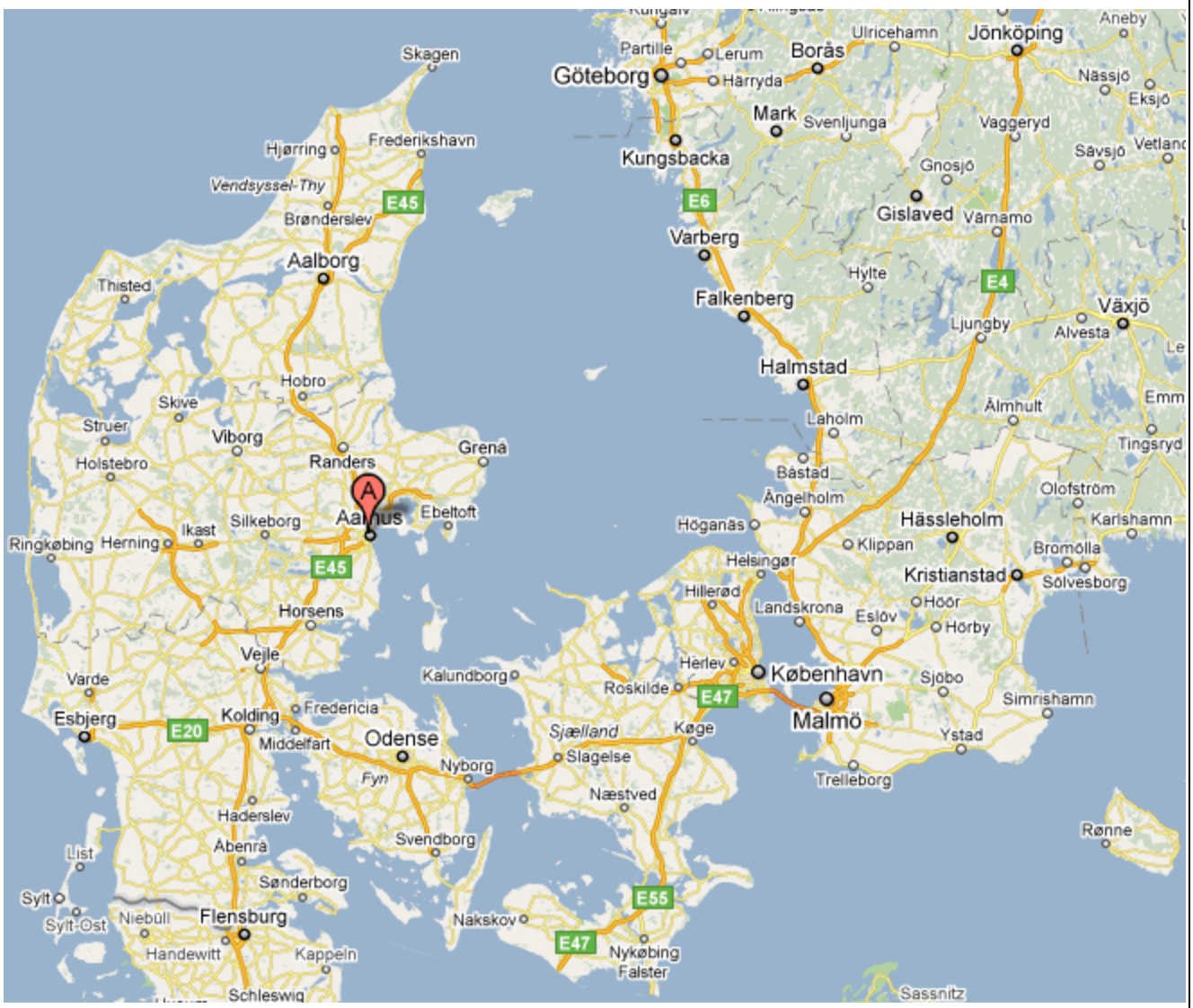
8. If the error looks like a bug in CopyMove then we also need a copy of the CopyMove diagnostics log file. See the section **Diagnostic Logging** to learn how to produce a diagnostics file using the `GetLogFile.cmd` script included in the product download.

© 2008 - 2016 SharePoint Products

SharePoint Products

Web: www.sharepointproducts.com

Address: Aabogade 15, DK-8200 Aarhus N, Denmark

Phone: +45 8734 5539

Sales: sales@sharepointproducts.com
Support: support@sharepointproducts.com
General: info@sharepointproducts.com