SharePoint Online Migration Considerations for Maximum Performance

May 2017

www.metalogix.com info@metalogix.com 202.609.9100

Copyright

© Copyright Metalogix International GmbH, 2001-2017. All Rights Reserved.

All rights reserved. No part or section of the contents of this material may be reproduced or transmitted in any form or by any means without the written permission of Metalogix International GmbH.

Metalogix Content Matrix Consoles™ is a trademark of Metalogix International GmbH.

Windows SharePoint Services is either a registered trademark or a trademark of Microsoft Corporation in the United States and/or other countries. Other product and company names mentioned herein may be the trademarks of their respective owners.

Technical Support

For information about Metalogix Technical Support, please visit http://metalogix.com/support.

Technical support specialists can be reached by email at support@metalogix.com or by phone at 1.202.609.9100.

We want to hear from you. Please send any comments regarding this document to support@metalogix.com.

The level of technical support provided depends upon the support package that you have purchased. Contact us to discuss your support requirements.

Metalogix

Contents

Overview 3

Functional and API Differences between SPO and SharePoint that impact migrations 3

Different Application Programming Interfaces available for SPO 3

Performance of Migrations to SPO with the new Migration API 4

Additional Best Practices unique to Content Matrix that can improve SPO migration performance even more 8

Performance Expectations 9

Recommendations 10

Setup an Azure Blob Storage Account 10

Use Microsoft Provided Containers 14

Use Azure Based Virtual Machines 15

Use Azure Active Directory Setup 19

Using Distributed Migration 20

Configuration 21

PowerShell 22

To achieve performance levels of 10GB/hour or higher 23

Disabling the 'Following Content' Feature 23

Mapping Domain users to Claims Based User Naming Conventions 24

Overview

Migrations from SharePoint on-premises (on-prem) to SharePoint Online (referenced as SPO in this document), have a number of significant factors that make them more complex and time consuming than on-premises to on-premises SharePoint migrations. These factors include:

- Functional and Application Programming Interface (API) differences between SharePoint on-prem and SPO that result in some on-prem elements not being easy or possible to migrate to SPO.
- Performance based limitations as a result of Microsoft's protection mechanisms to ensure that no single SPO tenant/client can cause an entire multi-tenant farm to be unstable.
- Performance impact of the physical distance between a client's on-prem SharePoint environment(s) and their SPO farm location.

This document explains why these limitations exist, and helps prepare Content Matrix clients to plan a migration to SPO. This includes:

- Listing and explaining the functional and API limitations and their significance on migrations
- Explaining the migration API leveraging Azure Storage to enhance the performance of content migrated to SPO
- Ensuring that customers allocate the appropriate time and consideration required when migrating their on-premises farms to SPO.

This document is essential reading for anyone migrating from SharePoint on-prem to SPO.

Functional and API Differences between SPO and SharePoint that impact migrations

Different Application Programming Interfaces available for SPO

For on-prem SharePoint, Microsoft offers a very rich API known as the SharePoint Server Object Model. This has been the most frequently used and richest API to access SharePoint on-prem.

However, due to the multi-tenant nature of SPO, Microsoft has to ensure that no single tenant can cause other tenants' environments to break. In light of that, Microsoft cannot expose the full SharePoint Object Model in SPO. They currently expose five API's that are useful for migrations:

- The Client Side Object Model (CSOM)
- The SharePoint REST API
- SharePoint Online Remote PowerShell (subset of CSOM)
- Office 365 Remote PowerShell (subset of CSOM)
- A Migration focused API that leverages Azure Blob Storage and batching to migrate list items and documents significantly more quickly to SPO (available both from CSOM and REST)

While the CSOM API is becoming close to the SharePoint Server Object Model in terms of feature parity, it is very slow for migrations. In light of that, Microsoft launched the new Migration API in May 2015. Today, the combination of the CSOM API and the new migration API is used by Content Matrix to write content to SPO. SPO and the CSOM API has some limitations that make Content Matrix and other migration tools unable to migrate all on-prem SharePoint content. For a full set of SPO limitations, see this MS web page listing. Another important page to look at is SharePoint 2013 deprecated functionality. This applies to both SharePoint on-premises and SPO.

The bottom line is that our experience at Metalogix serving many clients who have migrated to SPO is that it is absolutely crucial to understand your SharePoint on-prem Inventory, whether any of it cannot be migrated to SPO, etc. before you start migrating to SPO. Metalogix Expert is a free tool that can help you identify some, but not all of the issues that you can expect.

Performance of Migrations to SPO with the new Migration API

There are 3 primary factors that until now have impacted performance of migrations to SPO dramatically:

1. The physical distance between your on-prem SharePoint farm and your SPO tenant, and the fact that in most cases, this Wide Area Network (WAN), is across the Internet, making it very unpredictable.

2. The protection mechanisms and throttling that Microsoft has in place to protect their multi-tenant clients from abuse by other tenants on the same farm.

3. The very chatty nature of the CSOM API that is the primary API used for migrations to SharePoint Online, compounded by the distance required for each API call to travel between on-prem environments and SPO.

In order to alleviate the 2nd and 3rd aspects that slow down migrations, Microsoft has worked closely with Metalogix and other migration vendors to create a new Migration API that was introduced in May 2015 during Microsoft's Ignite conference.

The new API uses a combination of factors to alleviate the throttling and protection mechanisms and the chattiness of the CSOM API:

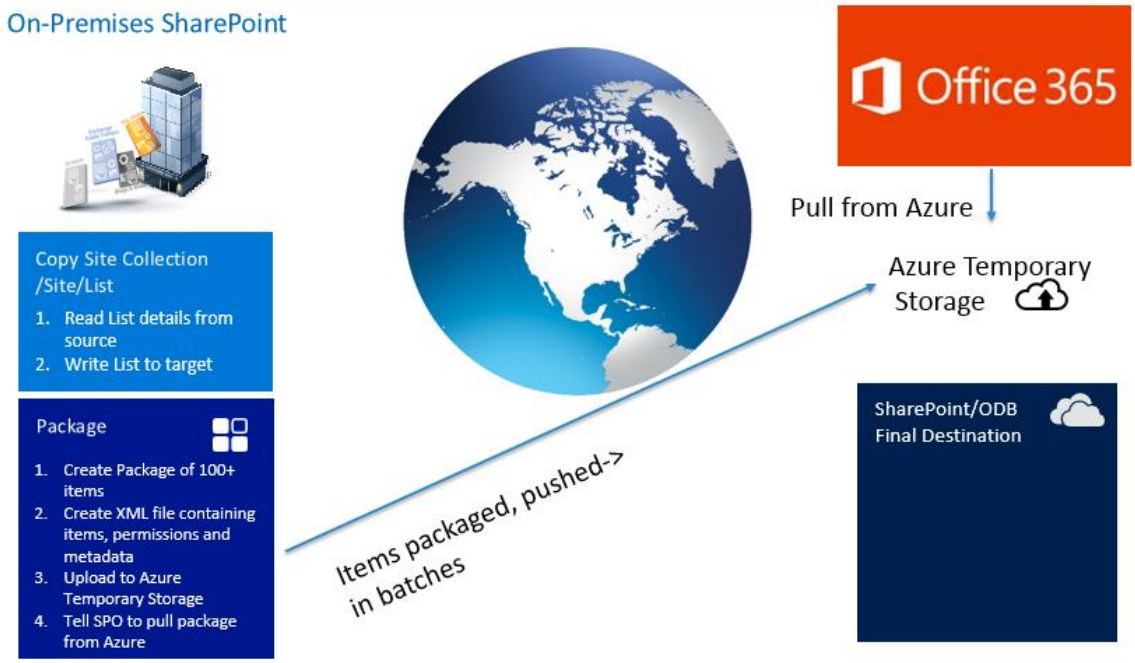
1. Site Collection structure, Site Structure, and List structure are still migrated using CSOM as they were before, and this happens prior to list items being migrated.

2. Instead of items being migrated with one API call at a time, they are now packaged locally into a batch with a corresponding manifest XML file to store permissions, metadata, etc. before being sent to SPO.

3. Instead of items/batches being sent directly to SPO, they are instead sent to an Azure Storage location (provided by either Microsoft or the user).

4. Once a batch has been successfully uploaded to the Azure Storage location, Content Matrix uses a CSOM request to ask SPO to pull content from the Azure Storage location into the previously created list in SPO.

Please see image below:



The net result is a significant increase in performance of list items, that can be anywhere between 5 to 30 times faster than these migrations were before depending on a number of factors.

In light of this, Metalogix strongly recommends that our clients leverage the new Migration API for all SPO migrations. Users can either provide their own Azure Storage locations (as detailed later in this document), or leverage ones provided by Microsoft. While providing an account is the faster of the two options, it is also the only option with a cost.

Additional Best Practices unique to Content Matrix that can improve SPO migration performance even more

In addition to the above mentioned migration improvements, Content Matrix has a number of unique capabilities that make migrations to SPO potentially much faster than they are with the new Migration API, and we encourage clients who are doing large migrations (1TB or more) to take advantage of these capabilities:

1. Content Matrix has the unique ability to connect directly to SharePoint Content Databases for 99% of the migration from SharePoint on-prem to SPO.
2. This allows our clients to:
    a. Backup their Content DBs on-premises
    b. Copy their Content DB .bak files to an Azure VM running SQL Server in the same geographical region as their SPO tenancy (if the content DBs involved are very large, you can also choose to ship physical disks to the Azure team, and that team will load them into your Azure VM).
    c. Restore the Content DBs to SQL Server in Azure VM.
    d. Use Content Matrix' "Connect to SharePoint Database" functionality to migrate directly from the Content DBs in the Azure VM to SPO:

3. As per the above diagram, by moving the Content DBs to an Azure VM in the same geographical region as your SPO tenancy, you have now moved the bulk of the objects

involved in the migration to what is effectively a LAN in the SPO geographical region. The Azure VMs are frequently in the same datacenter as the SPO infrastructure.

4. In addition to migrating directly from the Content DBs, Content Matrix allows you to:

    a. Migrate many items in parallel without a steep reduction in performance. Content Matrix has significant parallelism and multi-threading capabilities that allow you for example to migrate 3 or 4 Site Collections at the same time from a single Azure VM, and then spin up additional Azure VMs in which you can migrate additional Site Collections in parallel.

    b. Leverage the unique Distributed Migration capabilities of Content Matrix to distribute the load of migration across a multitude of different servers thereby increasing the level of parallelism.

    c. Leverage significant automation capabilities with the ability to use PowerShell to control any operation available in the Content Matrix UI programmatically. This opens up many possibilities including the ability to create a list or CSV file that contains source Site Collections, Target Site Collections, Site Collection Job Configuration files, and Status, and then have the migration machine start the next available migration in this list. This is a recommended configuration in tandem with Distributed Migration to automate migration actions across a multitude of different servers. Please reference the document "PowerShell Distributed Migration.pdf" for more information.

        i. An example of a list like this could be:

Title	Source Site Collection	Target Site Collection	Job Configuration File	Status
Finance NEW	http://tamirsp2010:1818/	https://metalogixsoftware.sharepoint.com/finance	Standard	Not Started
Marketing NEW	http://tamirsp2010:200/sites/MetalogixIntranet	http://metalogixsoftware.sharepoint/com/sites/MetalogixIntranet	Incremental	In Progress
Home NEW	http://tamirsp2010:300/sites/home	https://metalogixsoftware.sharepoint.com/sites/home	Standard with 3 year item filter	Not Started

Performance Expectations

There are a significant number of factors that will impact the performance of migrations using the new API. Many of those are beyond any migration vendor's control. For example, the SPO tenancy has a service that runs to pull data from Azure Blob Storage accounts. If that service is being called on by many clients at the same time, it could significantly slow down migrations.

Another example is the workflow. While the new API is significantly faster for list items/documents, it is not faster for complex site collection/sub site structures with very complex and large Managed Metadata Trees, workflows, content types, etc. With that in mind, we've seen migration speeds vary between 2GB/hour and 40GB/hour depending on workload when using reasonable hardware for the migration machines involved.

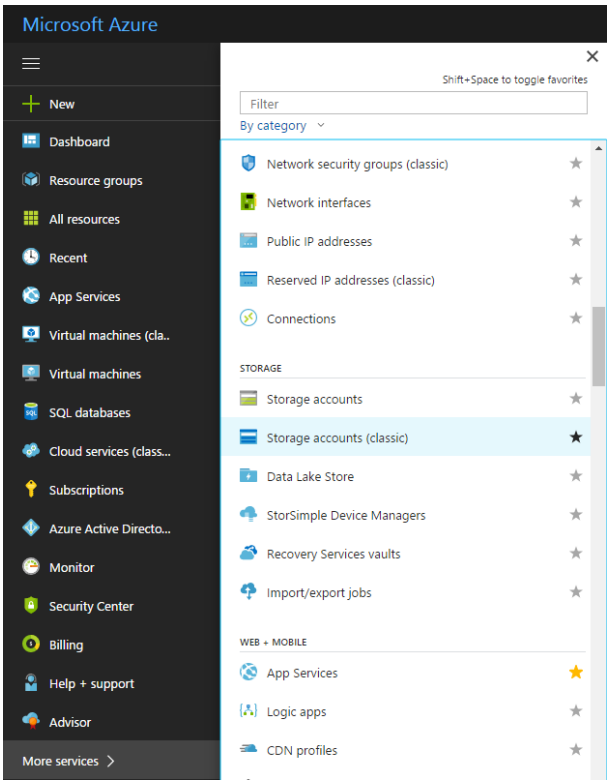
As compared to speeds of between 200MB and 550MB/hour on a single on-premises machine, and 400MB/hour to 10GB/hour on Azure VMs, this is a very significant increase. When trying to determine how long a project will take, we still recommend a conservative approach. Test several workload types using the new API, and preferable using Azure VMs, and see what you get. Run multiple jobs in parallel, and if you see no difference, and can afford the additional RAM/CPU, increase that, and then see what you get. Once you have those speeds, factor in that you will not maintain those speeds 24 hours/day due to variability in SPO. You will also have more remediation work to do post migration due to the differences between SP on-prem and SPO.

Recommendations

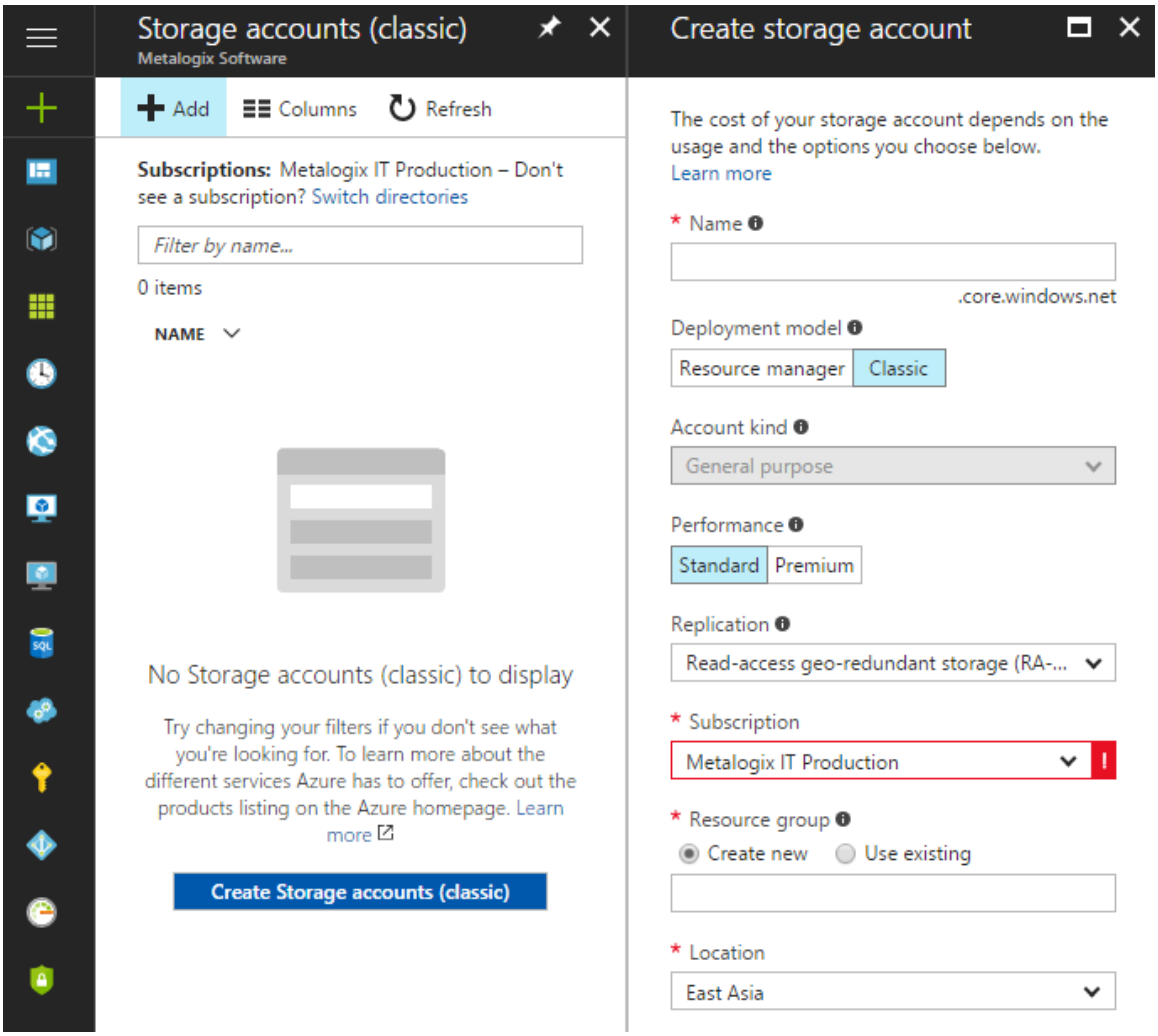
Setup an Azure Blob Storage Account

In order to use the new API, you can either provide your own Azure Storage location (for faster performance) or use one provided by Microsoft. This will detail how to generate your own once you have an Azure Login:

1. Log into the Azure portal by going to https://manage.windowsazure.com.
2. In the Azure Portal, choose the Storage accounts (classic) option:



3. Select “+ ADD”:

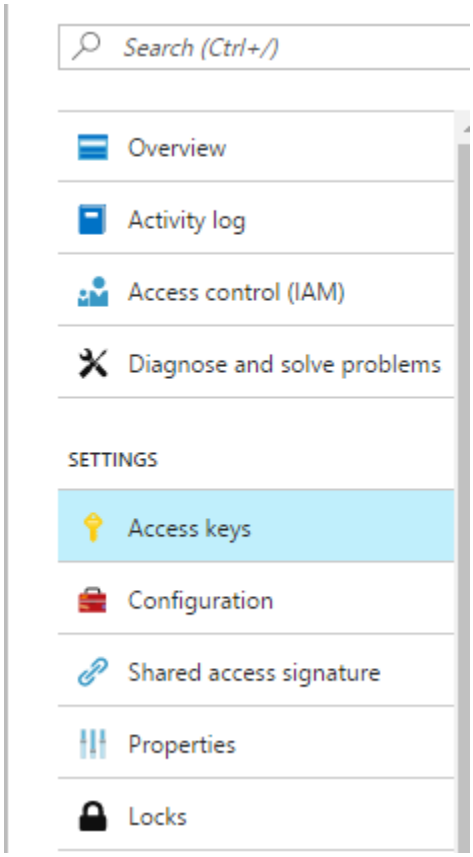


4. In the URL field, enter any name that can be verified.

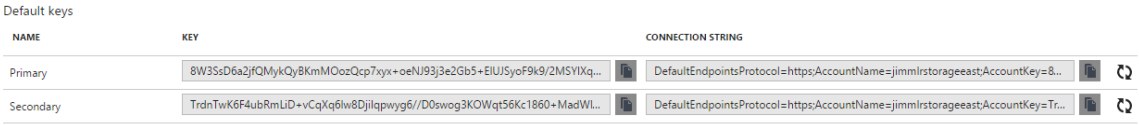
5. In the Location, make sure you put the location of your SPO tenancy. This is **critical** to achieve maximum performance. If you do not know the region your SPO tenancy is in, please contact your MS Technical Account Manager (TAM) if your company has one, or contact MS Support.

6. For Replication, select “Locally Redundant” and then select “Create”.

7. Once the storage account has completed creation, select “Access Keys” from the settings section of the screen:



8. Copy the Primary Connection String and store this for use later during configuration of Content Matrix:



9. On the machine(s) where you have installed Content Matrix SharePoint Edition, open the file c:\ProgramData\Metalogix\EnvironmentSettings.xml, and scroll to the bottom of this file.

10. Find the XML parameter below, and populate the connection string from the "CONNECTION STRING" field in step 8:

```
<XmlableEntry>

        <Key>UploadManagerAzureStorageConnectionString</Key>

        <Value>DefaultEndpointsProtocol=https;AccountName=acme123;AccountKey=Tk9zp+o4NUnxC7
        JCcdYzIUN4qYq0BA6nvGs2YGFBTZARvtOrIO/RkzapBR7mMdlz9w2tmP/er5JfXGpLWUL0JA==</V
        alue>
```

```
</XmlableEntry>
```

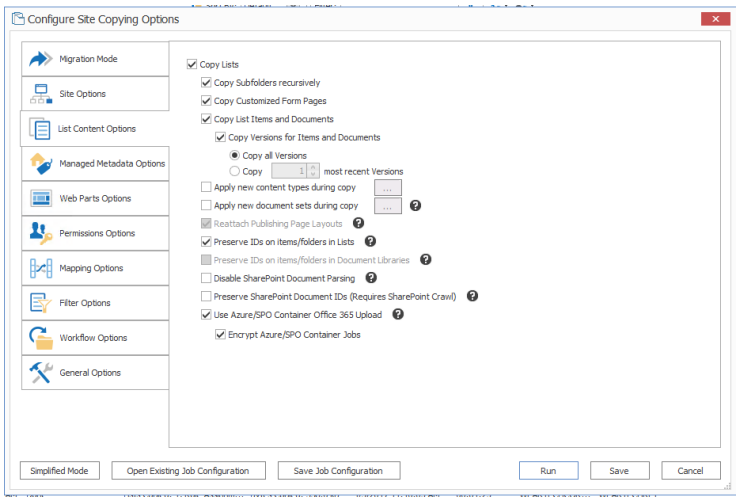
11. Note some of the other parameters related to the use of Azure Blob Storage. You can set the batch size in number of items, or MB, and determine how many items or how many MB should be in each batch. For large lists with smaller items, we recommend starting with batch sizes of between 100 to 200 items. For smaller or larger lists with larger items, we recommend setting the parameter “UploadManagerDetermineBatchSizesByMB” to True, and then setting the parameter “UploadManagerBatchSizeThresholdInMB” to between 300MB and 500MB. We will update this guidance as we learn more about what works best.

12. Find or add the parameter UploadManagerLocalTemporaryStorageLocation and set it to a location on your hard drive where processing of items and files can take place. It is important to have at least 40GB of free disk space for this processing. In the example below, I’m using C:\AzureAPI as the location for processing:

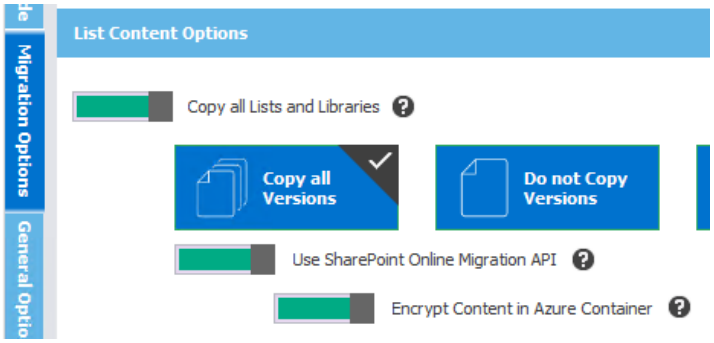
```
<XmlableEntry>

  <Key>UploadManagerLocalTemporaryStorageLocation</Key>

  <Value>C:\AzureAPI</Value>

</XmlableEntry>
```

13. Start Content Matrix as you normally would.

14. For any migration actions, in the List Content Options tab in Advanced Mode, select the “Use Azure/SPO Container Office 365 Upload” option. You may optionally choose to select “Encrypt Azure/SPO Container Jobs” for extra security at the cost of performance:



In Simplified Mode, under Migration Options and List Content Options, toggle the switch for “Use SharePoint Online Migration API”. You may optionally choose to select “Encrypt Content in Azure Container” for extra security at the cost of performance.



Use Microsoft Provided Containers

The Microsoft Provided Containers are a set of Azure Storage containers that are associated with a user's tenant. This container is provided to a user at no cost from Microsoft and is associated with the infrastructure the user's tenancy exists within. Content can only be uploaded to this location when it is also encrypted, making encryption a requirement for this action.

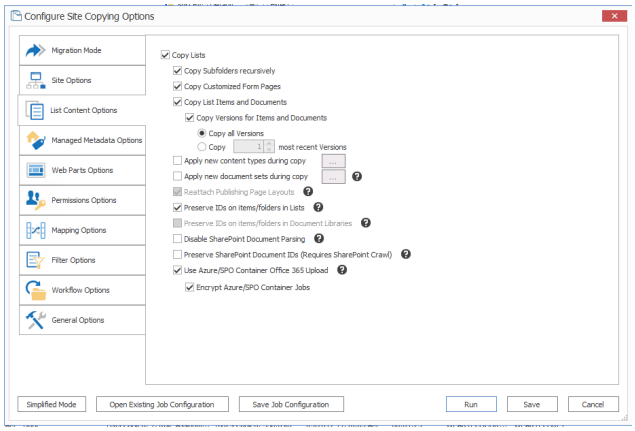
This action requires no work from the user outside of the selection of the appropriate checkboxes/toggles within Content Matrix.

1. On the machine(s) where you have Content Matrix SharePoint Edition installed, open the file c:\ProgramData\Metalogix\EnvironmentSettings.xml, and scroll to the bottom of this file.

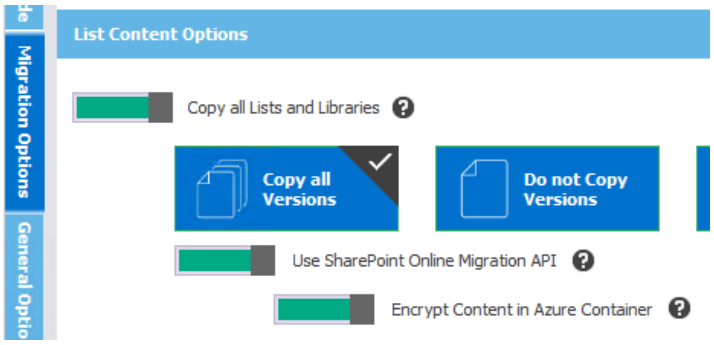
2. Find the XML parameter below and ensure the value of it is empty:

```
<XmlableEntry>
        <Key>UploadManagerAzureStorageConnectionString</Key>
        <Value />
</XmlableEntry>
```

3. For any migration actions, in the List Content Options tab in Advanced Mode, select both the "Use Azure/SPO Container Office 365 Upload" option and the "Encrypt Azure/SPO Container Jobs" option.



In Simplified Mode, under Migration Options and List Content Options, toggle both the switch for "Use SharePoint Online Migration API" and the switch for "Encrypt Content in Azure Container".



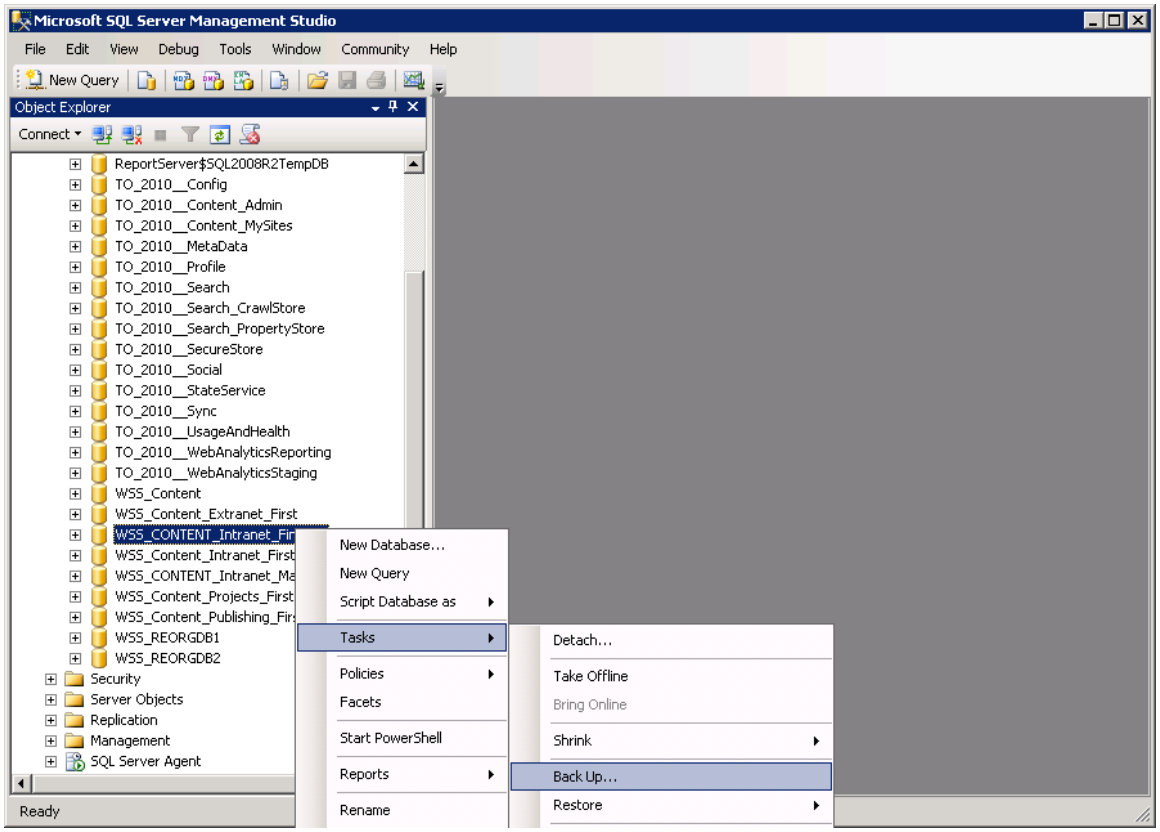
Use Azure Based Virtual Machines

As mentioned above, we have found that migrations from Azure Virtual Machines (VM) in the same region as the SPO instance being migrated to, increase performance of the migration. The process that we recommend for any migration to SPO is:

1. If you do not know the region your SPO tenancy is in, please contact your MS Technical Account Manager (TAM) if your company has one, or contact MS Support. If MS Support does not give you this information, please contact Metalogix, and we will get you the information.

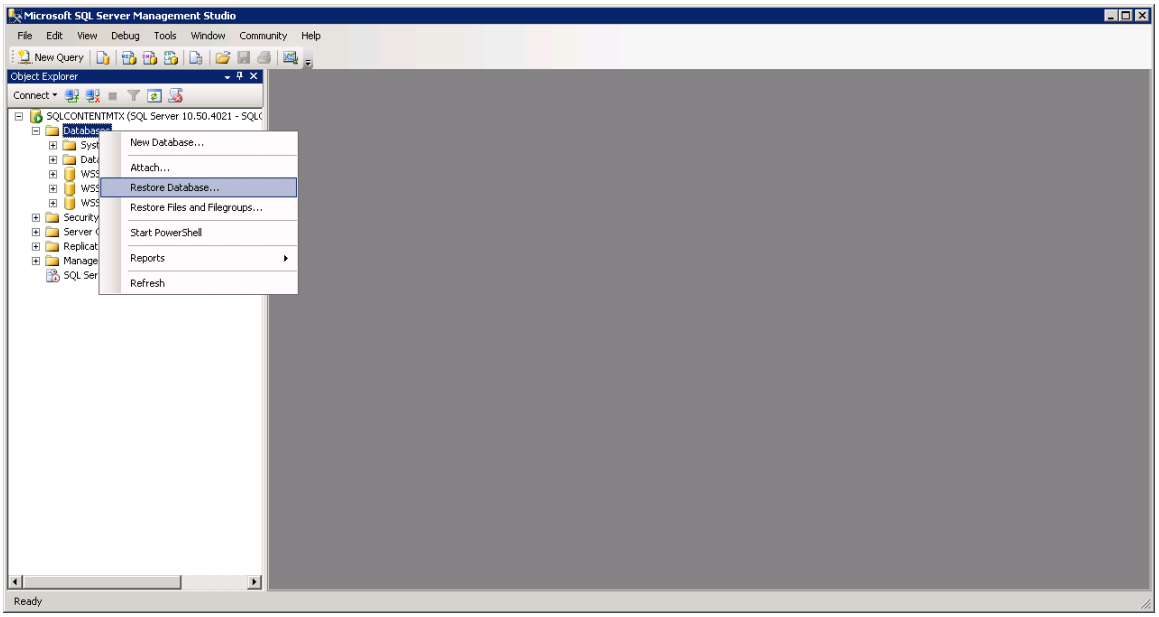
2. Create an Azure VM in the same region as your O365 tenancy with Windows Server 2008 or 2012, and SQL Server 2008 or 2012 (please note that if you have the bits and rights to use SQL Server 2008 or 2012, it is cheaper to create a Windows only VM, and then install SQL Server on it).

3. Using SQL Server Management Studio Back up your source Content DBs to .bak files:



4. Use ftp, or other mechanism to copy the .bak file to your Azure VM. You can ship a physical hard disk to the Azure team if needed. For more information how to ship a physical hard disk, please see Microsoft's guidance here.

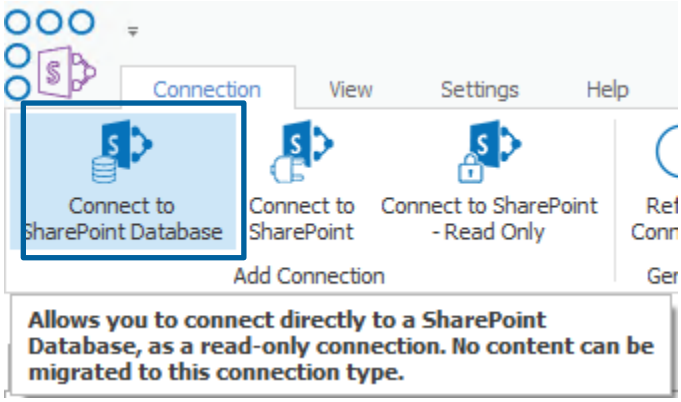
5. In your Azure VM, restore the .bak files to a new databases:



6. Once these databases have been restored, give your Azure VM admin dbo access to each of these databases

7. Install Content Matrix SharePoint Edition Console on the Azure VM (you can install the console on multiple other Azure VMs that do not have SQL Server on them, but must be in the same internal network in order to enable parallel, multi username actions).

8. In Content Matrix, use the Connect to SharePoint Database option:



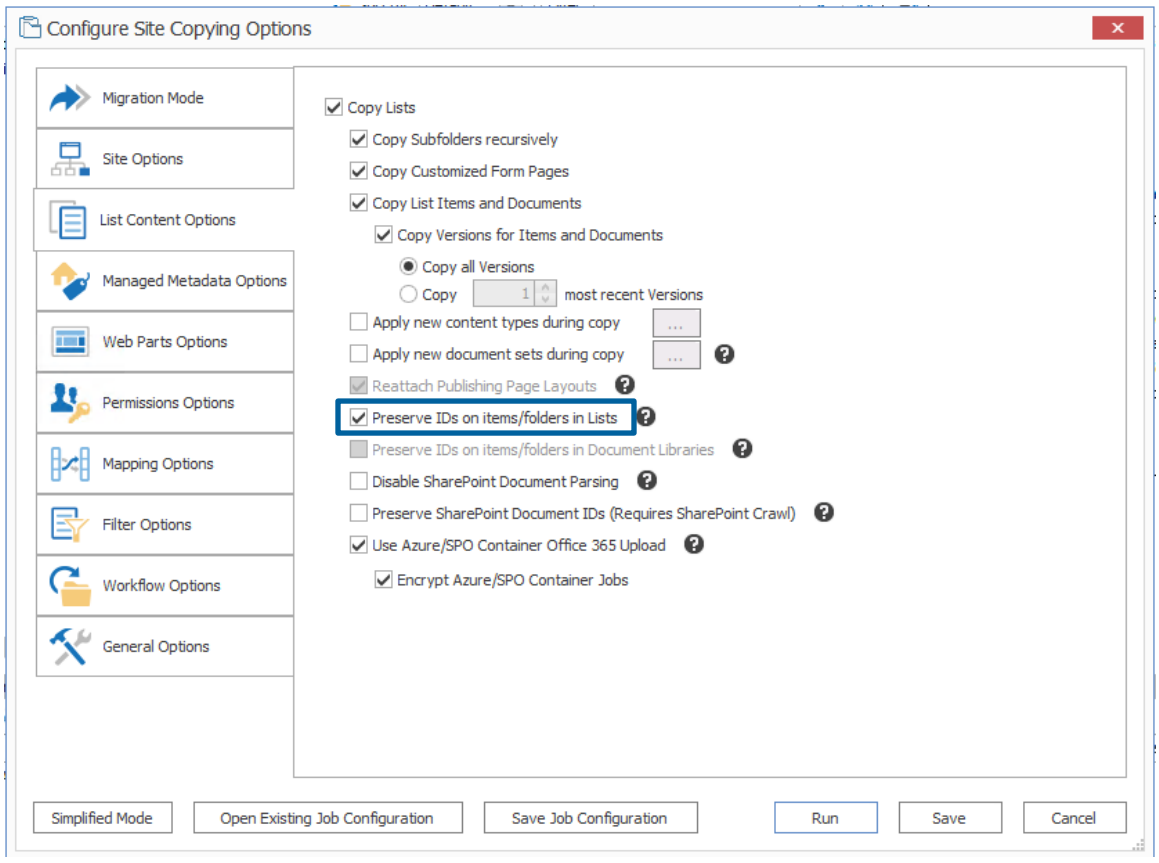
9. Enter the name of the database server (at this point most likely (local))

10. Choose the Content DB you just restored, and the Site Collection within that Content DB that you would like to connect to.

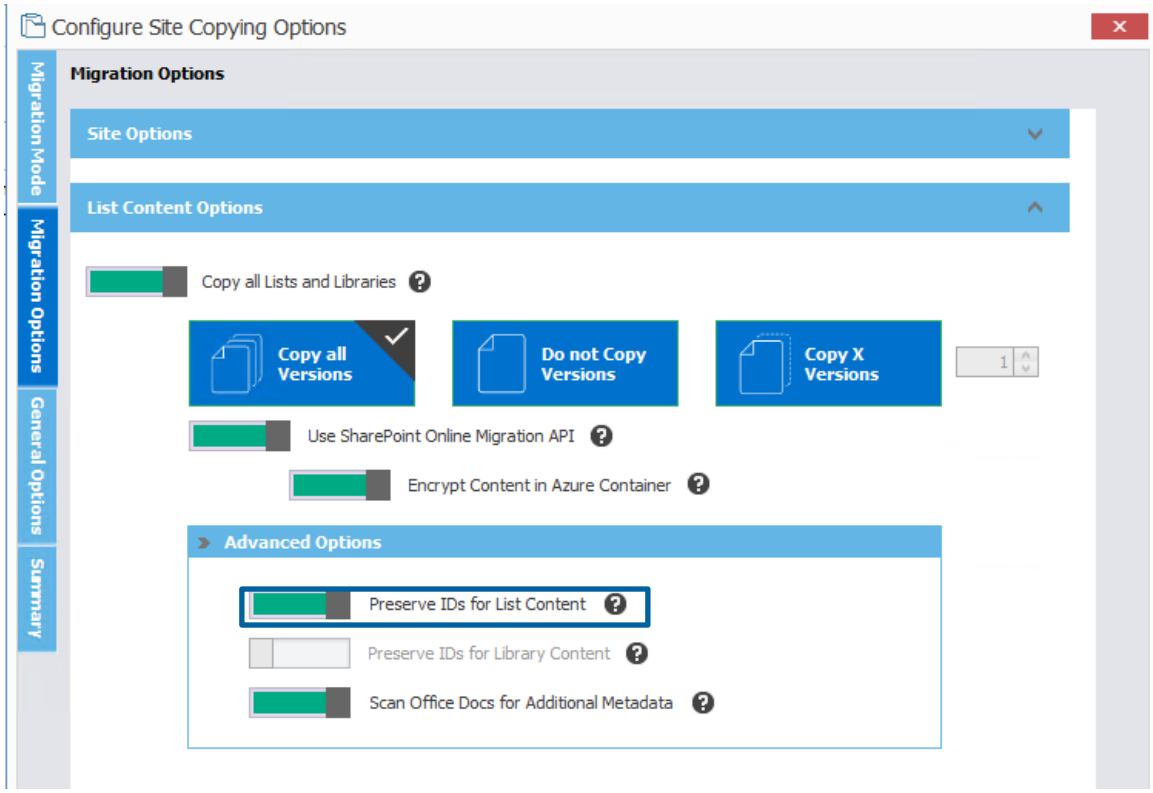
11. Migrate content as you normally would with Content Matrix

12. For any operations that require farm level connections (such as copying the Managed Metadata Services Term Store), migrate that from the source farm instead of Azure.

13. For the final Incremental Copy, there is no need to re-backup your Content DBs and restore them in Azure. You can perform an incremental copy from the actual source as long as you selected 'Preserve Item ID' in Advanced Mode for all List copy or Site copy operations that include lists:



If using Simplified Mode toggle "Preserve IDs for List Content" in the Advanced Options section of the List Content Options tab:



Use Azure Active Directory Setup

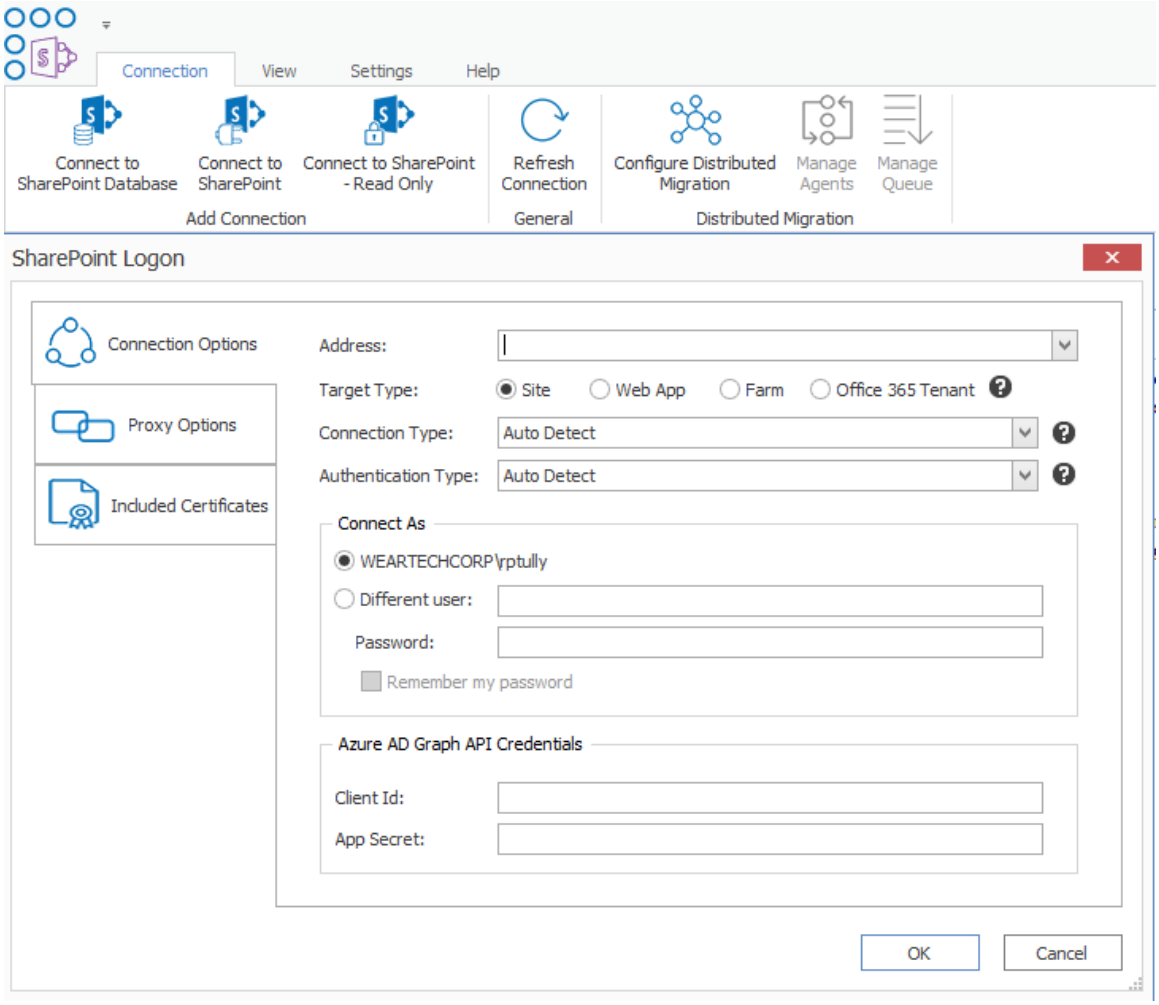
For a very small performance gain (potentially negligible), Content Matrix SharePoint Edition can communicate using a Client ID and Application Secret when connecting to SPO. This requires the following steps:

1. Register your application with Azure AD. Instructions on how to do so can be found here.
2. On the machine(s) where you have Content Matrix SharePoint Edition installed, open the file c:\ProgramData\Metalogix\EnvironmentSettings.xml, and scroll to the bottom of this file.
3. Find the XML parameter below set the value to "Graph":

   ```
   <XmlableEntry>
   <Key>ResolvePrincipalsMethod</Key>
   <Value>Graph</Value>
    </XmlableEntry>
   ```

   Note: the only options of this parameter are "Graph" and "People". "People" is the default behavior.

4. Connect to SharePoint using the “Connect to SharePoint” dialog. The screen will have additional fields. Enter the predefined Client Id and App Secret from Step One into these fields and connect as normal:

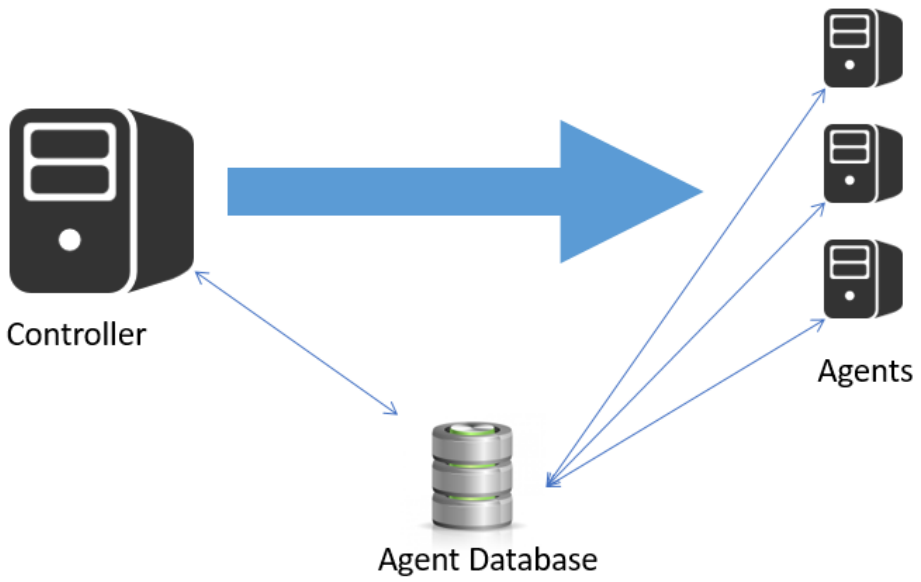


Using Distributed Migration

Content Matrix is able to leverage a unique feature called Distributed Migration in order to shift the load of multiple jobs to multiple machines using an Agent-Controller model for content migration distribution.

In the Agent-Controller model, one installation functions as the primary source for the distribution of migration jobs across multiple machines. The Controller does not (by default) participate in the migration process and only functions as a conduit for generating, configuring, and distributing jobs across the multitude of Agents. This does not prevent the Controller from executing jobs locally and independently.

On the Controller a Configuration Wizard will guide through the appropriate steps to generate an architecture similar to below:



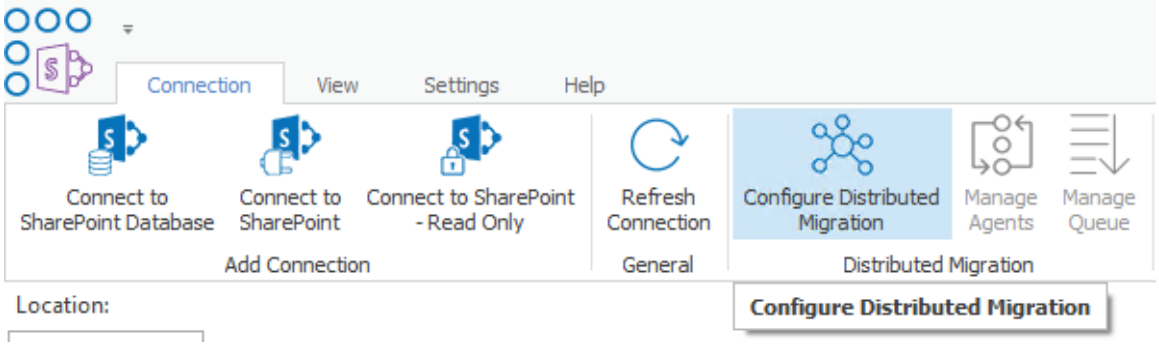
In this architecture, there is a shared database across all the listed machines. This database is known as the "Agent Database" and contains a shared list of all jobs configured for migration in this model, as well as a listing of all Agents and (if configured) a shared grouping of settings for all consoles.

In the previously picture scenario, the Controller generates all jobs and writes these into the Agent Database. The Controller then queues up the listed jobs, identifies any Agents that are not currently performing a migration, and pushes that job to that Agent. Should there be no free Agents, jobs will remain queued until an Agent becomes available. As an Agent migrates content it will write job information back to the Agent Database. This information can then be viewed from the context of the Controller.

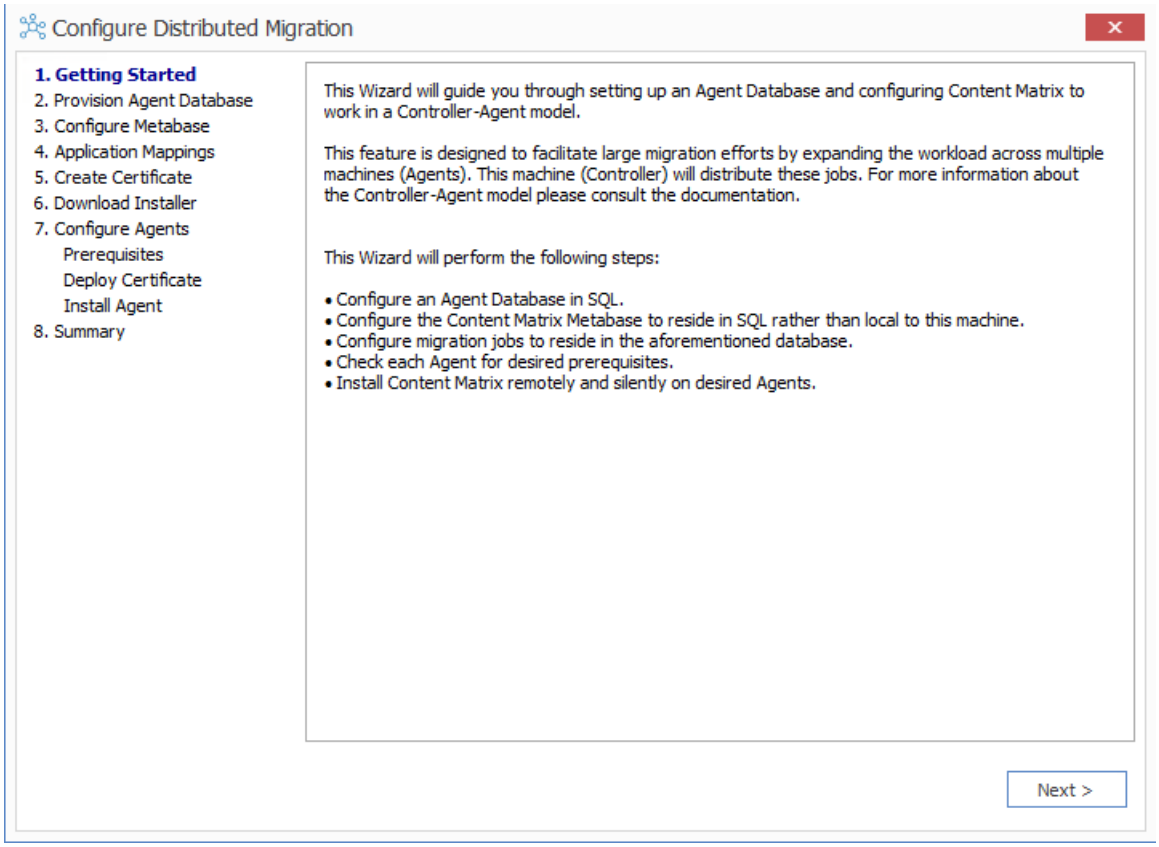
Configuration

The following steps should be followed for leveraging Distributed Migration appropriately:

- Launch Content Matrix SharePoint Edition.
- Within the top ribbon of the console is a button labeled "Configure Distributed Migration":



- Upon clicking that button, a Wizard will pop-up walking the user through the configuration. The Wizard will provide information for all actions being performed. Please consult the documentation should there be concerns with any particular screen or configuration.



PowerShell

As PowerShell extensibility is one of the major tenents of Content Matrix, that extensibility has been brought to the concept of Distributed Migration. Please reference the document "PowerShell Distributed Migration" for further details of how to leverage PowerShell for an Agent-Controller model.

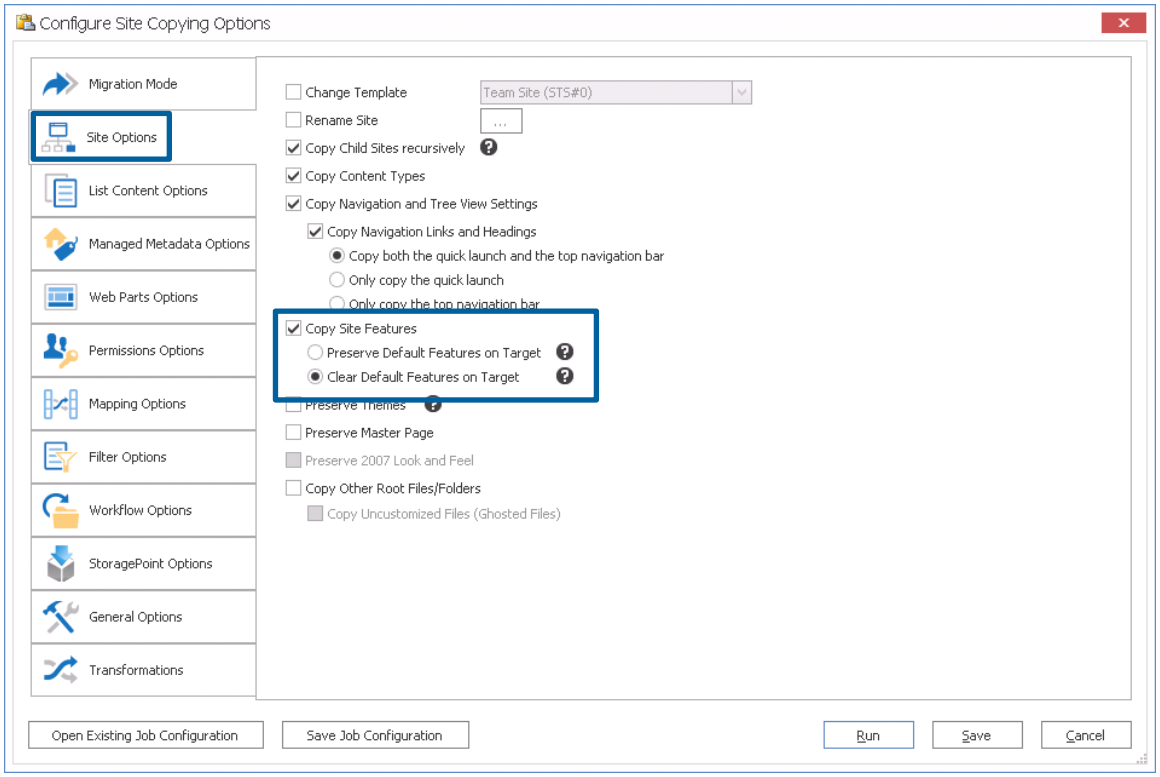
To achieve performance levels of 10GB/hour or higher

1. Create multiple migration users in your SPO tenancy, each with Global Admin permissions (MS recently rolled out SPO Admin roles. If those are available in your tenancy, you should be able to use those instead of Global Admin roles).
2. Spin up multiple Azure VMs in the same domain as the SQL Server mentioned above.
3. Ensure that the SQL Server VM has sufficient RAM to deal with multiple simultaneous jobs.
4. Plan your migration so that each VM is migrating several jobs at a time, with larger Site Collections or Sites being migrated on separate machines (i.e.. Balance the workload across multiple VMs).
5. To achieve optimal performance, first migrate the structure of your SharePoint farm, and then the content.
6. Use Distributed Migration to shift the load across multiple VMs.
7. Use PowerShell to control migration jobs. This allows you to programmatically control migrations, check against an Excel file, DB, or SharePoint list for a list of jobs that are still required, and perform migrations in the order you would like without manning the consoles frequently.
8. PowerShell also uses far fewer UI resources, and therefore results in faster migrations. PowerShell is also able to take advantage of the Distributed Migration features.

Disabling the 'Following Content' Feature

SPO sites (but not site collections), have a feature called: 'Following Content'

- This feature is enabled by default when a new site is created. When it is disabled, performance of Site copies improves by approximately 20% to 25% for document heavy workloads.
- Since Content Matrix does not currently give you the option of turning off only a particular feature on the target site, the only way to achieve this currently is to specify in the Site Options tab of the Paste Configuration screen to 'Clear Default Features on Target':



Mapping Domain users to Claims Based User Naming Conventions

Content Matrix automatically maps domain users from source to target domains if domain mapping is in place. For example, if the source domain is metalogix1, and the target domain is metalogix2, setting up a Domain Mapping in the 'Edit Global Mappings' screen accessible from the Settings menu (see our online help on this), allows users to setup a domain mapping between metalogix1 and metalogix2.

Once that has been setup, Content Matrix will automatically map metalogix1\jdoe to metalogix2\jdoe, and so on. In addition, for migrations to Office 365 or other claims based authentication providers, if the source domain is the same name as the email address format of usernames in Office 365 scenarios using Active Directory Federation Services (ADFS), a typical source domain name such as metalogix1\jdoe would be automatically mapped to i:0#.f|membership|jdoe@metalogix.com.

However, if you have multiple domains on your source farm, or domains that are not named the same as the @<companyname>.com email address that the user would use to login to Office 365, you can still create the appropriate mapping.

There are two ways to address this. These are described in detail including a sample csv file and PowerShell scripts that can be used to make the process easy in a document titled “Content Matrix User Mapping from AD to Claims including O365”, which is included in the installer zip file for Content Matrix. This document is also available in the Metalogix Support Knowledge Base located within the Metalogix Customer Portal.