Microsoft Dynamics CRM (on-premises) to Microsoft Dynamics Online Migration Guide

Steps for migrating a Microsoft Dynamics CRM (on-premises) organization to Microsoft Dynamics CRM Online

April 2014

Author(s): MinhTri Tonnu, Aravind Nair, Azharuddin Mohammed, Naveen Choppadandi, Aditya Varma, Erik Hansen, Ramanuj Brahmachary, Leah Clelland Jochim

Copyright

Contents

Microsoft Dynamics CRM (on-premises) to Microsoft Dynamics CRM Online migration background

The purpose of this document is to provide step-by-step guidance and best practices that can be leveraged by Microsoft Dynamics partners and customers to migrate an organization in a Microsoft Dynamics CRM deployment (on-premises) to Microsoft Dynamics CRM Online. The goal is to streamline the migration process to make the on-boarding experience easy and faster for the customers.

A team of Microsoft Dynamics CRM experts working closely with the Microsoft Dynamics CRM product group adopted a two-phased approach. The first phase is to migrate a Microsoft Dynamics CRM organization. The second phase involves fine-tuning and optimizing the performance, based on learnings from the first phase. This approach uses end-to-end validation of a complex Microsoft Dynamics CRM instance, which in our study included the following:

- 1.5 Terabyte of data
- ~500 GB of email attachments
- Integrating with Office Outlook and Microsoft Data Warehouse system

In the first phase, the end-to-end migration process from CRM (on-premises) to CRM Online took 12 days. During the second phase, after performance enhancements had occurred, the complete process took 3 days.

The migration process leverages Scribe desktop version 7.6.2.38821 to import data from CRM (on-premises) to CRM Online. You can expect that the CRM (on-premises) migration performance will be similar if you use the Scribe Online version.

This white paper documents the key steps that are required to migrate the CRM data as well as recommendations for scaling the migration to increase speed of the effort overall and minimize downtime.

Prerequisites

- A Microsoft Dynamics CRM 2013 or Microsoft Dynamics CRM 2011 on-premises deployment to use as the migration source.
- An instance of Microsoft Dynamics CRM Online to use as the migration target.
- A migration tool, such as Scribe.
- Administrative-level access to all products and technologies required for the migration (Active Directory, SQL Server, Microsoft Dynamics CRM, and so on).

Note Performance data included in this document are based on the organization migration of a Microsoft Dynamics CRM 2011 (on-premises) deployment to Microsoft Dynamics CRM Online. Notice that, similar results will be obtained migrating from Microsoft Dynamics CRM 2013 (on-premises).

Migration steps

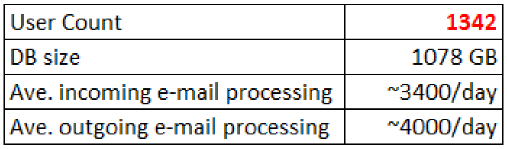
1) Prepare the database for bulk data import

a) Back up the CRM (on-premises) configuration and organization databases.
b) Ensure that the respective individuals and teams involved in the migration (that is infrastructure, operations, and consultants) are granted administrative access to the CRM (on-premises) deployment, SQL Server, and so on.

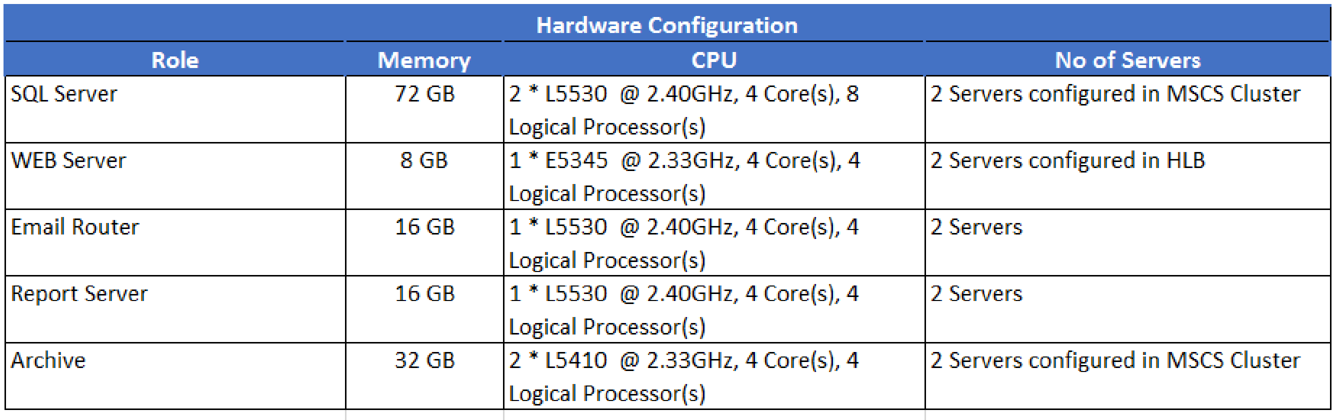
2) Microsoft Dynamics CRM (on-premises) system preparation

a) Select and review scenarios optimal for testing and data validation.

b) Ensure that the source Microsoft SQL Server has sufficient storage, RAM, and reliable high bandwidth network connectivity. For the purpose of this bulk data import, the following table reflects the configuration used in Microsoft tests and provides an example of a recommended CRM (on-premises) and SQL Server system set up.



User Count	1342
DB size	1078 GB
Ave. incoming e-mail processing	~3400/day
Ave. outgoing e-mail processing	~4000/day



Hardware Configuration			
Role	Memory	CPU	No of Servers
SQL Server	72 GB	2 * L5530 @ 2.40GHz, 4 Core(s), 8 Logical Processor(s)	2 Servers configured in MSCS Cluster
WEB Server	8 GB	1 * E5345 @ 2.33GHz, 4 Core(s), 4 Logical Processor(s)	2 Servers configured in HLB
Email Router	16 GB	1 * L5530 @ 2.40GHz, 4 Core(s), 4 Logical Processor(s)	2 Servers
Report Server	16 GB	1 * L5530 @ 2.40GHz, 4 Core(s), 4 Logical Processor(s)	2 Servers
Archive	32 GB	2 * L5410 @ 2.33GHz, 4 Core(s), 4 Logical Processor(s)	2 Servers configured in MSCS Cluster

c) Install a compatible Microsoft Dynamic CRM data migration tool, such as Scribe, with administrative access.

d) Ensure that multiple (at least two) client computers (physical or virtual) are available to run the migration tool and facilitate scale-out and performance tuning for the migration.

3) Microsoft Dynamics CRM Online environment configuration

a) Ensure that the CRM Online environment (target) uses the same Active Directory as the CRM on-premises (source).

b) Create a CRM Online instance and make sure you have administrative access to that instance.

c) To ensure the CRM (on-premises) solution customizations are applied to the new CRM Online instance, perform the customization import in the specific order of the following steps:

i) Create a solution in the CRM (on-premises) organization that you want to migrate.

(1) Add all entities used in the system to the solution. For more information, see Create your own solution.

(2) Add all the following components in this specific order:

(a) Global OptionSet(s);
(b) Dashboards;
(c) FetchXML Based Reports (if there are any SQL-based reports they must be converted to FetchXML-based prior to importing);
(d) Sitemap;
(e) Application ribbons;
(f) All the web resources available in the system;
(g) All processes;
(h) Plug-in assemblies;
(i) SDK message processing steps;
(j) Email templates and all other templates;
(k) Security roles; and
(l) Field security profiles.

(3) Convert the Microsoft Dynamics CRM (on-premises) plug-ins to Sandbox mode and then add it to the solution created in step 3) c) i).

ii) Export the solution as unmanaged from CRM (on-premises) and then import the solution to the new CRM Online instance.
iii) Open each CRM Online workflow and resolve any referential issues (if any). Then, activate each workflow one by one.
iv) If you are using the Email Router, reconfigure the Email Router to connect and route to the CRM Online instance.
v) Verify that the customizations are correctly set up for incoming/outgoing email flow for the CRM Online organization.

4) Bulk data import

a) Identify the sequence of entities to be imported by analyzing the relationship among them in the solution. This is to ensure parent (or master) entity records are imported first so you can avoid errors while importing child entity records due to missing references.

b) Create one/multiple jobs and Data Translation Specification (DTS) file for each entity:

i) Connect the CRM (on-premises) organization to the new CRM Online instance using the CRM Adapter in the import tool (such as Scribe). Test to verify that the connection between the two systems is functioning.
ii) To receive the required set of records from the source entity, configure the source entities by using the object entity or custom queries.
iii) Configure the steps to be performed in the target (that is insert/update).
(1) The first migration effort will be an insert.
(2) Subsequent updates are often required for manual data changes and new data that arises.
iv) Map the required records from the CRM (on-premises) source to the new CRM Online target with the appropriate inputs.
v) Configure formulas in the migration tool, such as Scribe, for base currency and primary Business Unit (BU) to ensure the destination values are mapped to the source correctly.

c) Configure the migration tool, such as Scribe, for the internal database that is used to store the failed or rejected rows of that job. This captures any failures that may occur, which can be addressed later in a cleanup phase. If you are using the Scribe tool, there are some key steps such as the following:

i) Open Scribe Workbench as “Administrator,” to ensure there is no issue when you save the modified settings in the DTS files.
ii) Configure the option “Rejected Source Rows” to store the failed records.
iii) In the Source option (Microsoft Dynamics CRM on-premises deployment), keep only those fields that must be mapped to the target (Microsoft Dynamics CRM Online). To improve the data import rate, remove all other unmapped fields. Refer to Appendix (3) Scribe Tool – Field mapping for more information.
iv) Enable the "Use bulk-mode" operation to accelerate the data import rate.
v) Use the "Dynamics CRM bulk-API" feature to accelerate the data import rate.
vi) Refer to the Appendix in this document for best practices and tips to fine tune your DTS file creation and formulas to improve the data import rate.
vii) For more information about the steps we followed for the bulk data imports, see the Research section.

d) If any entity has related entity data:

i) Create entity records that are in an Open State. Notice that you can’t add or update entities after the entity record is in a closed or completed state.
ii) Run a separate Update job to update the status. For example, for attaching the activities to a closed case, first, the case must be created in an Open state, then import the activities associated to the case records, and then update them as Closed.
(1) Configure the job for Create.
(2) Configure the job for Update.
iii) Run one or more jobs based on the level of parallel execution needed.

5) Final Microsoft Dynamics CRM Online configuration

a) Fix and document import issues as needed. Return to the additional Scribe table that was created earlier in step 4 c).
b) Setup remaining integrations as applicable such as the following:
   i) Feedstore. Down and upstream applications that the source CRM (on-premises) system received or sent data to.
   ii) Authentication mechanisms.
   iii) New service endpoints.
c) If you use the Email router, set up and test it for use with the Microsoft Dynamics CRM Online instance.
d) Smoke test the new CRM Online environment. The basic "smoke test" should include validation of the following functionality:
   i) Business rules.
   ii) Correctness and completeness of data in the CRM Online instance.
   iii) Incoming and outgoing email flow.
   iv) Access permissions (based on Microsoft Online Services user roles and Microsoft Dynamics CRM security role configuration).
   v) Data Integration scenarios (if applicable).

e) Make adjustments to the s CRM Online instance based on the final test results.

Research behind the bulk data import from Microsoft Dynamics CRM (on-premises) to Microsoft Dynamics CRM Online

1) Research background

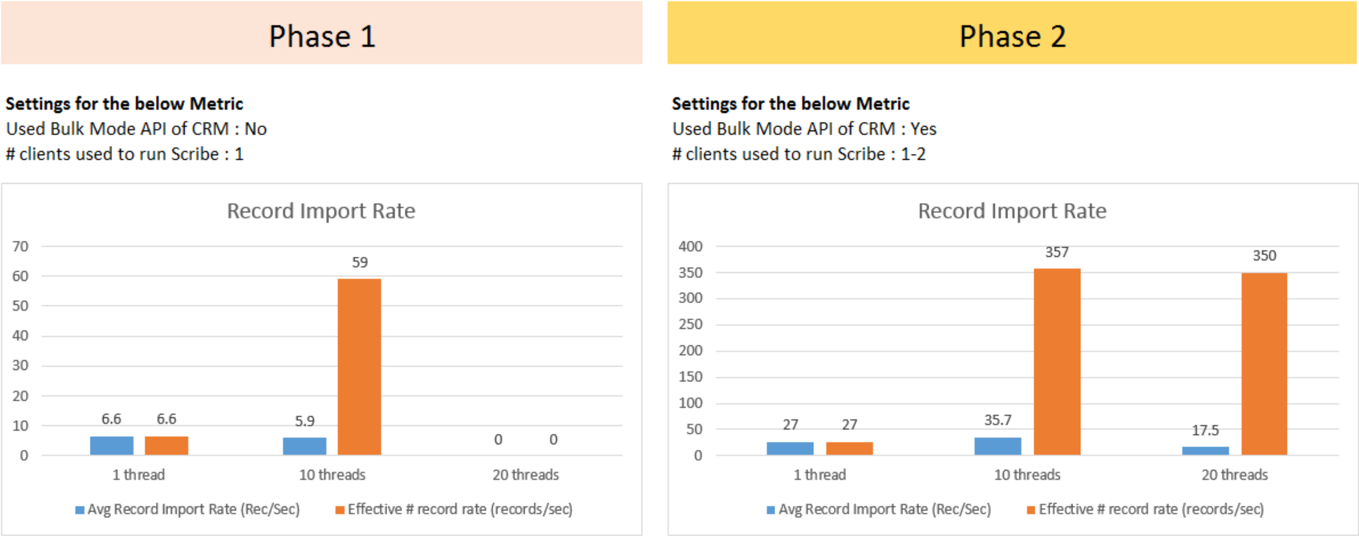
The recommend steps and performance tuning listed here was used to successfully bulk import data from Microsoft Dynamics CRM (on-premises) to Microsoft Dynamics CRM Online. This information was derived from the migration of one terabyte (TB) of CRM data. This effort was conducted two times so that the second effort could incorporate collaborative learnings and import and system optimization recommendations from industry experts and members of the Microsoft Dynamics product team.

Based on comparing the first phase of bulk data import to the second, and capturing the performance for both, a final test of CRM (on-premises) versus CRM Online response time was conducted. It demonstrated clearly that the cloud-based version was faster. Ultimately, the fastest results were obtained with changes to the SQL Server settings for the cloud.

2) Lessons learned

a) Data import time for one TB of data can be reduced by more than half from 14 days to less than three days with the following adjustments as shown in the following diagram.
   i) The use of increased parallel threads of import jobs from 1 to 20 and
   ii) Using bulk mode API.

Diagram 1a: Data Import Rates



Phase 1

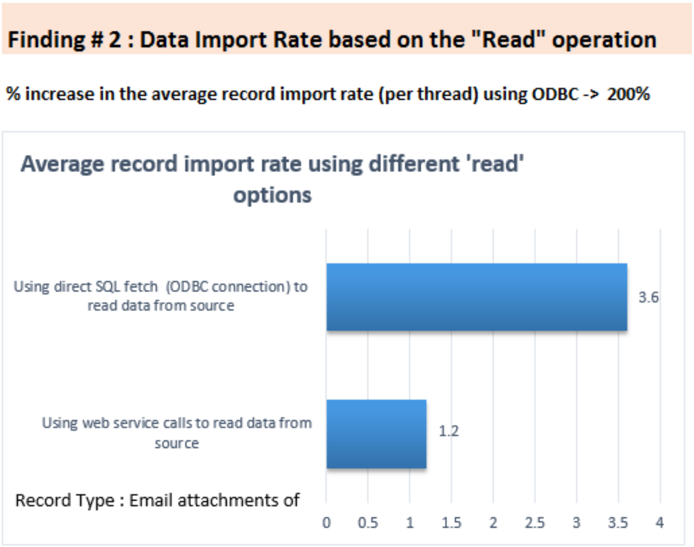
Settings for the below Metric
Used Bulk Mode API of CRM : No
# clients used to run Scribe : 1

Phase 2

Settings for the below Metric
Used Bulk Mode API of CRM : Yes
# clients used to run Scribe : 1-2

b) Data import rate for the “Read” operation for the CRM (on-premises) data can be increased 200% by switching from web service calls to SQL Fetch (ODBC connection) as shown in Diagram 2.

Diagram 2: Data import rate “Read” operations using web service calls vs. SQL Fetch (ODBC)

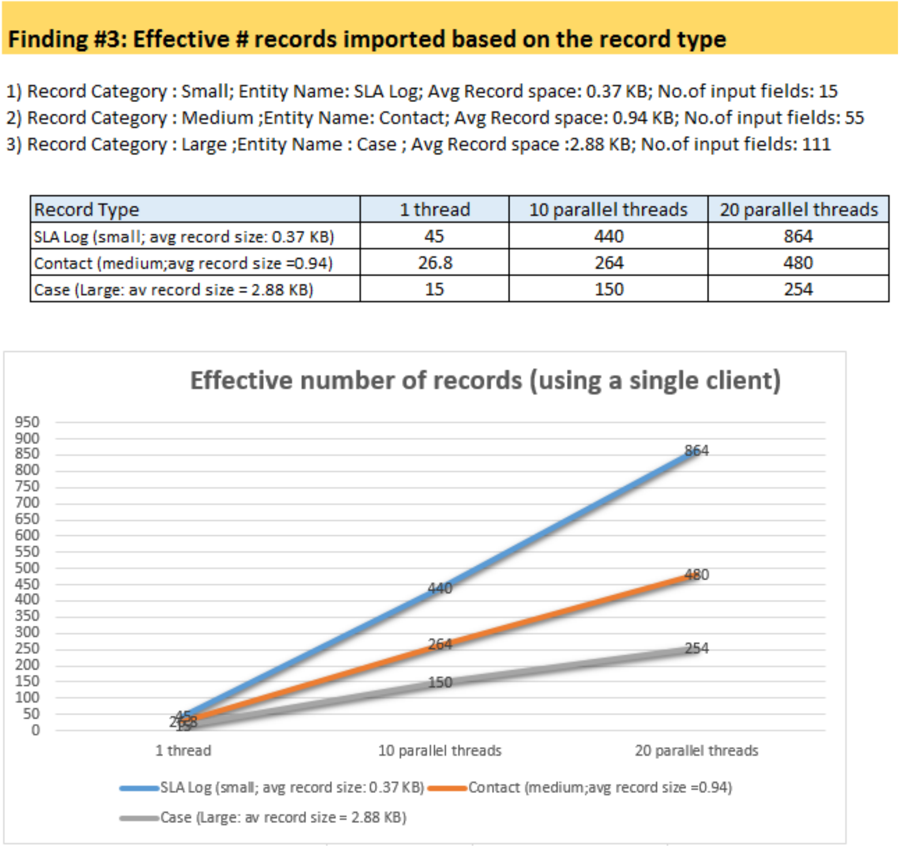


Finding # 2 : Data Import Rate based on the "Read" operation

% increase in the average record import rate (per thread) using ODBC -> 200%

c) Increasing parallel threads during data import using the Scribe tool from 1 to 10 to 20 improves the import rate of records of all sizes as shown in Diagram 3.

Diagram 3: Data import rates for records of various sizes using increased parallel threads



Finding #3: Effective # records imported based on the record type

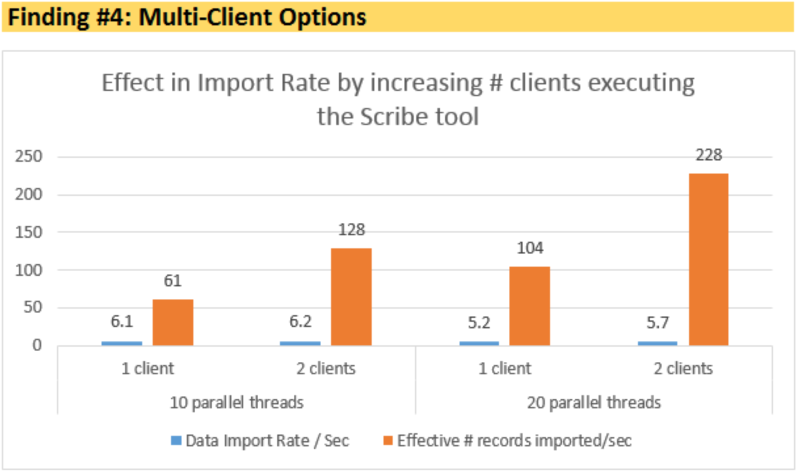
1) Record Category : Small; Entity Name: SLA Log; Avg Record space: 0.37 KB; No.of input fields: 15
2) Record Category : Medium ;Entity Name: Contact; Avg Record space: 0.94 KB; No.of input fields: 55
3) Record Category : Large ;Entity Name : Case ; Avg Record space :2.88 KB; No.of input fields: 111

Record Type	1 thread	10 parallel threads	20 parallel threads
SLA Log (small; avg record size: 0.37 KB)	45	440	864
Contact (medium;avg record size =0.94)	26.8	264	480
Case (Large: av record size = 2.88 KB)	15	150	254

Effective number of records (using a single client)
950
900
850
800
750
700
650
600
550
500
450
400
350
300
250
200
150
100
50
0
45
26.8
15
440
264
150
864
480
254
1 thread
10 parallel threads
20 parallel threads
SLA Log (small; avg record size: 0.37 KB)
Contact (medium;avg record size =0.94)
Case (Large: av record size = 2.88 KB)

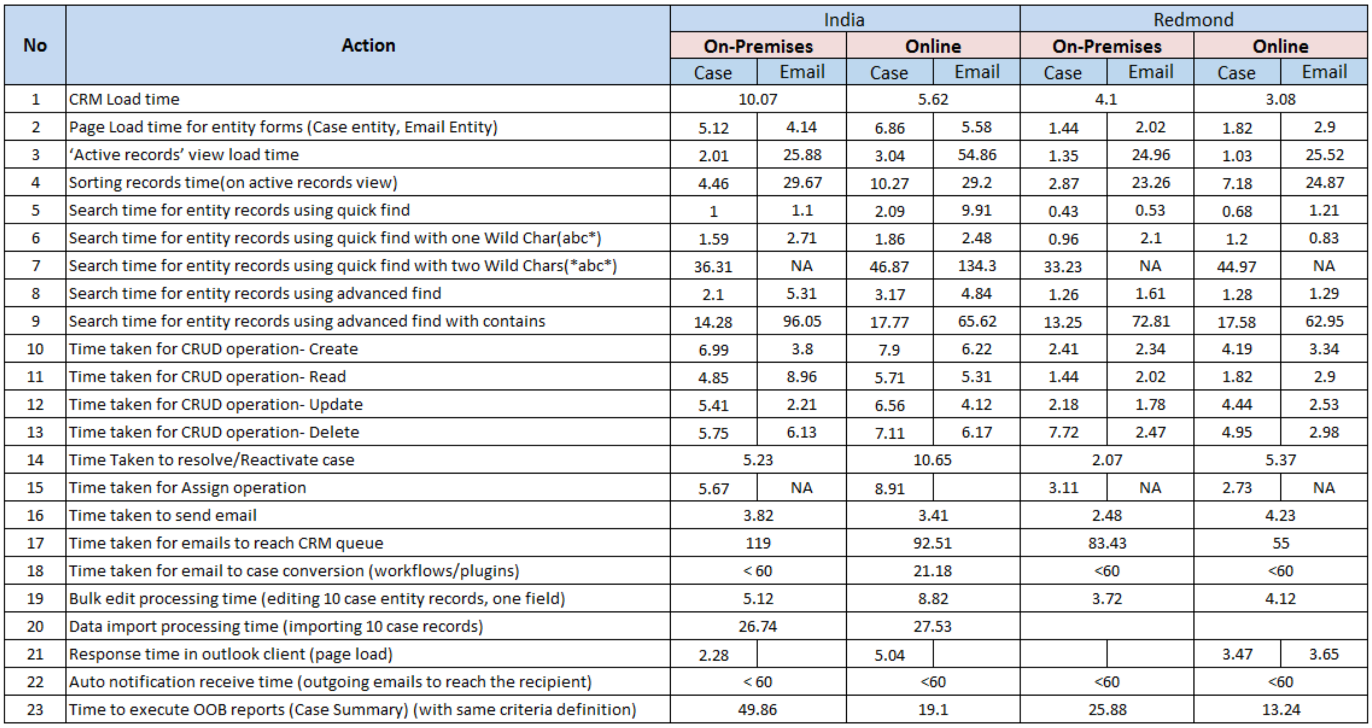
d) Increasing the number of clients that are running the migration tool (Scribe) as well as parallel threading exponentially increases the data import speed as shown in in the following diagram.

Diagram 4: Increased import rate by increasing the number of clients running the migration (Scribe) tool


Finding #4: Multi-Client Options

e) Successful bulk data import combined with performance tuning outlined here will provide a cloud-based solution comparable or faster than CRM (on-premises) versions.

Diagram 5: Response rates in two locations for Microsoft Dynamics CRM (on-premises) and Microsoft Dynamics CRM Online



No	Action	India				Redmond			
		On-Premises		Online		On-Premises		Online	
		Case	Email	Case	Email	Case	Email	Case	Email
1	CRM Load time	10.07		5.62		4.1		3.08	
2	Page Load time for entity forms (Case entity, Email Entity)	5.12	4.14	6.86	5.58	1.44	2.02	1.82	2.9
3	'Active records' view load time	2.01	25.88	3.04	54.86	1.35	24.96	1.03	25.52
4	Sorting records time(on active records view)	4.46	29.67	10.27	29.2	2.87	23.26	7.18	24.87
5	Search time for entity records using quick find	1	1.1	2.09	9.91	0.43	0.53	0.68	1.21
6	Search time for entity records using quick find with one Wild Char(abc*)	1.59	2.71	1.86	2.48	0.96	2.1	1.2	0.83
7	Search time for entity records using quick find with two Wild Chars(*abc*)	36.31	NA	46.87	134.3	33.23	NA	44.97	NA
8	Search time for entity records using advanced find	2.1	5.31	3.17	4.84	1.26	1.61	1.28	1.29
9	Search time for entity records using advanced find with contains	14.28	96.05	17.77	65.62	13.25	72.81	17.58	62.95
10	Time taken for CRUD operation- Create	6.99	3.8	7.9	6.22	2.41	2.34	4.19	3.34
11	Time taken for CRUD operation- Read	4.85	8.96	5.71	5.31	1.44	2.02	1.82	2.9
12	Time taken for CRUD operation- Update	5.41	2.21	6.56	4.12	2.18	1.78	4.44	2.53
13	Time taken for CRUD operation- Delete	5.75	6.13	7.11	6.17	7.72	2.47	4.95	2.98
14	Time Taken to resolve/Reactivate case	5.23		10.65		2.07		5.37	
15	Time taken for Assign operation	5.67	NA	8.91		3.11	NA	2.73	NA
16	Time taken to send email	3.82		3.41		2.48		4.23	
17	Time taken for emails to reach CRM queue	119		92.51		83.43		55	
18	Time taken for email to case conversion (workflows/plugins)	< 60		21.18		<60		<60	
19	Bulk edit processing time (editing 10 case entity records, one field)	5.12		8.82		3.72		4.12	
20	Data import processing time (importing 10 case records)	26.74		27.53					
21	Response time in outlook client (page load)	2.28		5.04				3.47	3.65
22	Auto notification receive time (outgoing emails to reach the recipient)	< 60		<60		<60		<60	
23	Time to execute OOB reports (Case Summary) (with same criteria definition)	49.86		19.1		25.88		13.24	

Note The metrics provided here are specific to the implementation we used for our CRM (on-premises) to CRM Online migration. Your actual results will depend on factors specific to your implementation of CRM (on-premises), which include customizations, amount and complexity of CRM data, infrastructure, and network bandwidth.

3) Recommendations

a) Ensure the source (on-premises) SQL Server, web servers, and the server where the import tool runs have performant hardware, sufficient available RAM and hard disk space.

b) Any client computer running the Scribe tool for uploading the data should have at least 20% available memory.

c) The last CRM (on-premises) Deletion Service Job should have completed successfully. This can be viewed in CRM in **Settings** > **System Jobs**.

d) The fix for PrincipalObjectAccess table growth should be applied to the CRM (on-premises) organization database. For more information, see http://support.microsoft.com/kb/2664150.

e) Asynchronous (Async) Cleanup Jobs should have completed successfully for the on-premises application database. For more information see: http://support.microsoft.com/kb/968520.

f) Use multiple clients to run the import tool to accelerate the data import rate (effective records per second).

g) Use the CRM Bulk API feature of Microsoft Dynamics CRM to import the data in batches. This can improve the data import rate.

Note We don't recommend migrating an organization database that has a lot of records with more than 2 MB in each record, such as email records with attachments. This can cause the import to fail due to the heavy load on the server.

h) Run multiple jobs as parallel threads using the import tool to increase the data import rate. 10-20 parallel jobs is ideal.

i) If the CRM (on-premises) organization has plug-ins that use on create, convert them as asynchronous before data import to ensure a good import rate.

j) If the CRM (on-premises) organization has plug-ins that aren't required at the time of data import we recommend that you turn them off. Allow 6-7 hours for the backlogs to complete after the data import and then turn on the plug-ins again. This is to ensure a good data import rate.

k) Ensure the services listed here are in a Stop state on the source Microsoft Dynamics CRM (on-premises) server(s) while you do the data import.

i) Asynchronous Processing services.

ii) Asynchronous Processing (maintenance) services.

iii) Deletion services.

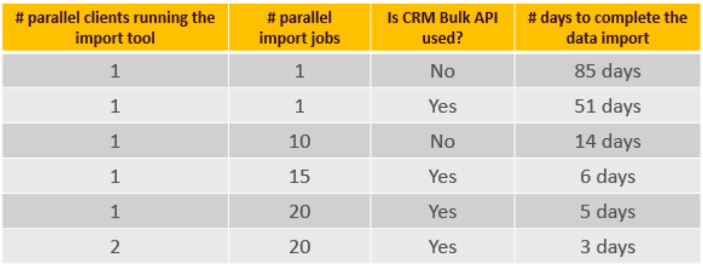
iv) Index creation services.

l) For entities with a large average record size, such as email records with attachments, use direct SQL Fetch from the source CRM (on-premises) to accelerate the data import rate.

m) We recommend that you have the server where the import tool runs (Scribe) and the source CRM (on-premises) server(s) located in the same data center. This is so that the SDK calls at the source don't deteriorate performance and become a bottleneck.

n) The range of options that are used will ultimately have a direct effect on the number of days a given bulk data import will take as demonstrated in the following diagram.

Diagram 6: Data import time for 1 TB database



# parallel clients running the import tool	# parallel import jobs	Is CRM Bulk API used?	# days to complete the data import
1	1	No	85 days
1	1	Yes	51 days
1	10	No	14 days
1	15	Yes	6 days
1	20	Yes	5 days
2	20	Yes	3 days

4) Known issues

a) With recommended workarounds.

i) Bulk data imports.

Issue Summary	Description	Work Around
During data import from Microsoft Dynamics CRM (on-premises) to Microsoft Dynamics CRM Online, the createdon time stamp uses the current system date of import.	Defaulting to system date can affect the migration if the application has SLA logic based on this field value.	Map the "createdon" field by using the "overriddencreatedon" field, correct record created date will be reflected in CRM Online.
If the activity party in the source is deleted for any activity, importing activity parties into Microsoft Dynamics CRM Online will result in an error.	Because the reference is broken for the activity parties, an error occurs while importing them. In the CRM (on-premises) organization, some emails contains invalid (non-existing) activity parties. When creating these activity parties in CRM Online, the import fails.	Review the feasibility of having a dummy contact for all such references.
Can't import quick campaign records.	Quick campaigns can't be imported by using tools.	Create manually using the Microsoft Dynamics CRM SDK.
Error while importing records where the owner permission was removed or the user was deactivated.	If an owner no longer has privileges for records in a CRM (on-premises) organization, creating them in CRM Online will cause the error message "Principal User is missing prvReadNew_Reseller privilege."	Add security roles to the default teams. This will make sure users who belong to the team will get those permissions and allow the successful import of these records.
Importing email attachments for quick campaign related emails takes a long time.	When users send email messages from a quick campaign to multiple accounts, contacts, the system creates the email records and corresponding email attachments. Only one record for the attached file will be created in the "Attachments" table and that record will be referred in all records in the ActivityMimeAttachments table. When you import email attachments to Microsoft Dynamics CRM Online, the Scribe tool creates new attachment records for each ActivityMimeAttachment record instead of using a single attachment record. As a result, import time takes longer and the attachments table will significantly increase in size. For example, In Microsoft Dynamics CRM, if a user sends email messages to 1000 contacts using a Quick Campaign by attaching a 3 MB file, then 1000 email messages and 1000 email attachments (Activity Mime Attachment) records are created. However, only one record will be created in the "Attachments Table". When importing those records to CRM Online, instead of single 3MB file, a new attachment record with 3MB data will be created for each Activity Mime Attachment. So there are 1000 records, each with 3 MB of data. Since the database size will increase by 3000 MB it will take longer to create each of the 3 MB files.	1. Import one single email attachment of Quick Campaign email record to Microsoft Dynamics CRM Online. 2. Then, by using the Microsoft Dynamics CRM SDK or Scribe tool, retrieve the "Attachment Id" of that email attachment from CRM Online, because it will be different from CRM (on-premises). 3. Then import all other email attachments related to the same quick campaign by passing the "Attachment Id" captured from the previous step. Remove the "body" field from Source query, because you are mapping to the existing attachment and it is not required. 4. Repeat the previous steps for other different attachment files and quick campaigns. Because you are using the same attachment ID in all related email attachments, the database size will not grow. Because you are not mapping the body field for rest of the email attachments, the import will be faster.

ii) FetchXML

Issue Summary	Description	Work Around
Microsoft Dynamics CRM (on-premises) SQL-based custom reports will throw an error while importing them to Microsoft Dynamics CRM Online.	CRM (on-premises) SQL-based custom reports throw an error while importing them to CRM Online because SQL-based reports aren't supported with CRM Online.	Convert your direct SQL report to FetchXML in the CRM (on-premises) solution and then add the solution to CRM Online.

b) Pending a workaround

i) Bulk data imports

Issue Summary	Description
During data import, modified by, modified on, and created by timestamps use the current system dates and the properties of the user who ran the import.	Modified By, Created By value is set to the user who is running the import (system). Modified On value is set as the time of the data import.
Can't import audit records.	Audit records can't be imported to CRM Online because CRM doesn't support a create operation for Audit records.
Can't import queue items.	Queue items can't be created as part of the data migration.
During import of marketing lists, the "Last Used On" field does not reflect the actual value. Instead, it is populated with the import date.	This behavior can affect the auditing and reporting of the migrated data.
Organization service does not have any create option for asynchronous operation records.	In progress workflow instances will be affected and the history of the asynchronous jobs will become delayed.

ii) FetchXML

Issue Summary	Description
FetchXML - No equivalent for Common Table Expression (CTE)	SQL-based custom report conversion to FetchXml causes a delay in the migration.
FetchXML - No equivalent for Conditional Processing (CASE WHEN)	
FetchXML - No equivalent for Filtering by aggregate-based criteria (HAVING)	
FetchXML - No equivalent for Aggregate Operation "MAX, MIN" on Date field	
FetchXML - No equivalent for composite conditions on joins	
FetchXML - No equivalent for Set Operators (UNION, INTERSECTION, MINUS and so on)	
FetchXML - No equivalent for Nested Subqueries / Operating on "derived tables"	
FetchXML - No equivalent for string functions (like Left, Right, Substring...)	
FetchXML - No equivalent for SQL User-Defined Functions (UDFs)	
FetchXML - No equivalent for Temporary Tables	
FetchXML - No equivalent for Subqueries with Aggregate functions (it allows join with only 2 entities)	
FetchXML - No equivalent for Conditional statement within the Query (like IF, Else, While)	

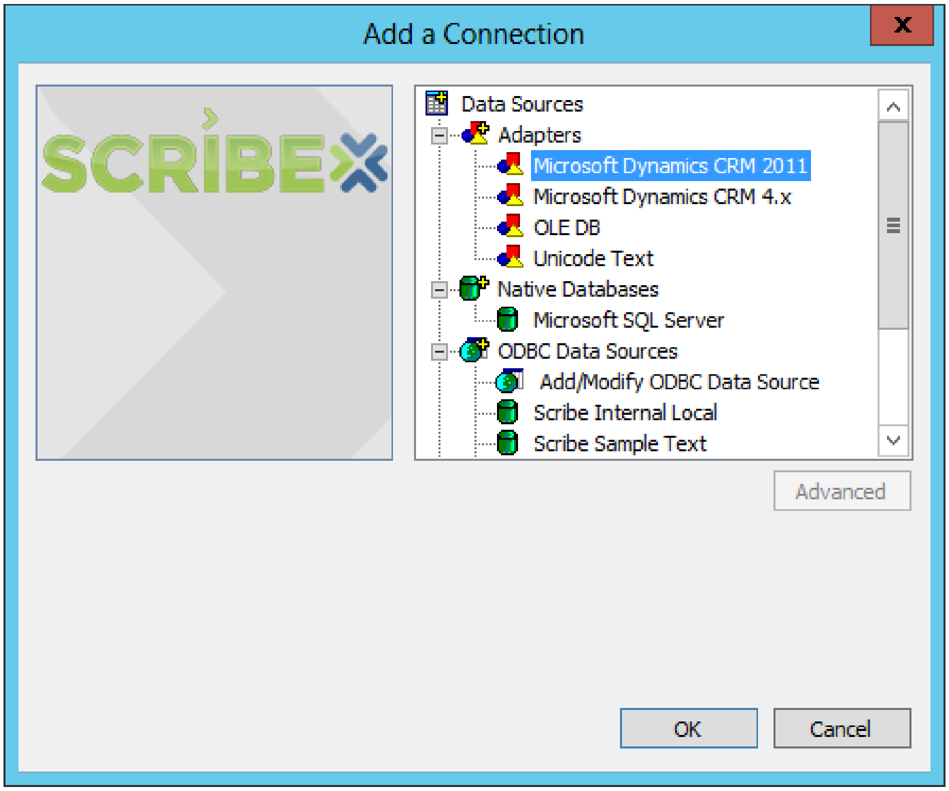
Appendix

1) Scribe Workbench best practices

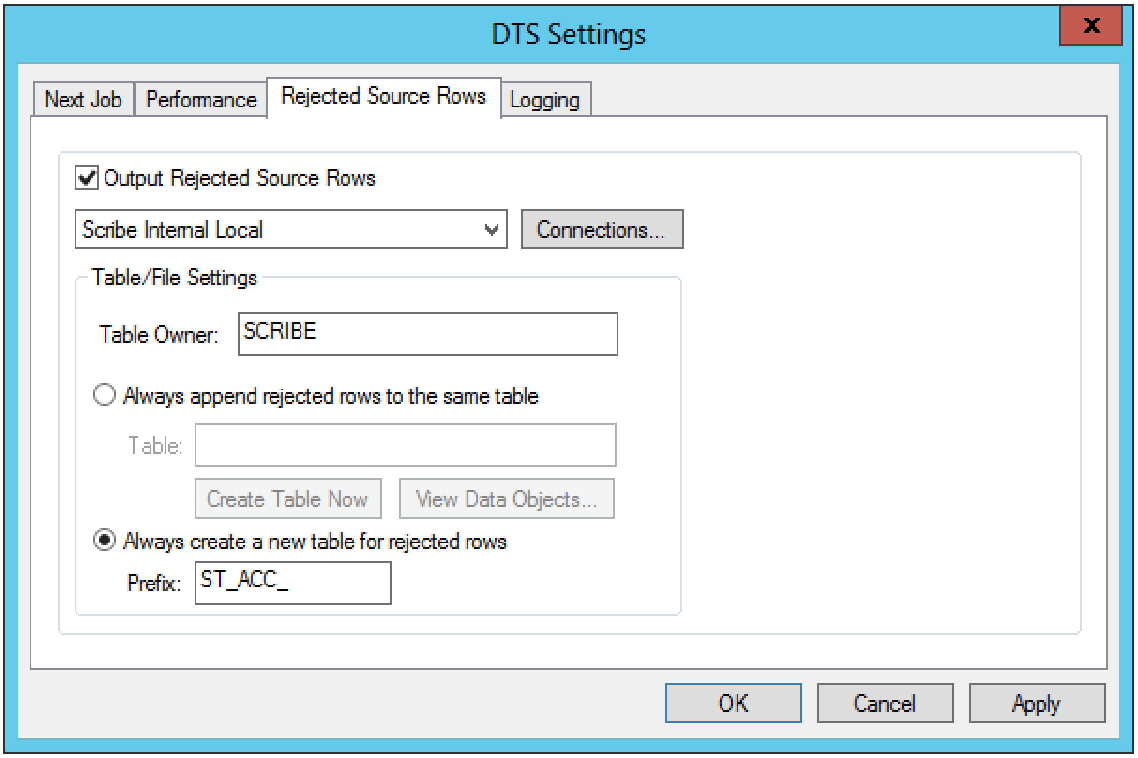
1. Make sure to open the Scribe Workbench as ‘Administrator.’ This ensures that any settings modified in the DTS file can be saved without any issues.

2. Naming Convention. To easily identify and track migration files, name the DTS files in the format “Orgname_EntityName_[Operation].dts”.
   Example: For import of new accounts -- “STRATA_Account.dts”
   To update existing accounts -- “STRATA_Account_Update.dts”

3. Based on the Microsoft Dynamic CRM versions of source and destination instances, use the respective version of the CRM adapter for Scribe for the connection. For example, if the source and destination instances are in Microsoft Dynamics CRM 2013, use the CRM 2013 adapter for Scribe.
   Note Use an ODBC data source instead of a CRM adapter if you need complex filtering and there is no “created on” field mapping in the **Email Attachments** and **Activity Parties** entities.



4. Configure Rejected Source Rows. To do this, click **Run** > **Edit Settings** > **Rejected Source Rows** tab. Give a unique Prefix (8 characters), which identifies the entity of your organization. All rejected rows of this job will be inserted into this table.



☑ Output Rejected Source Rows

Scribe Internal Local

Table/File Settings

Table Owner: SCRIBE

○ Always append rejected rows to the same table

Table:

● Always create a new table for rejected rows

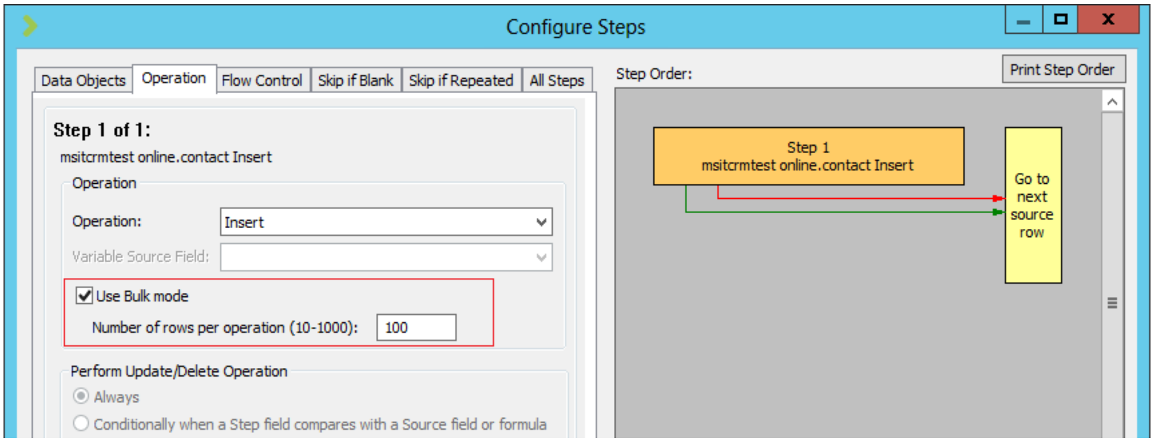
Prefix: ST_ACC_

5. In the source, keep only the fields that must be mapped to the target. Remove all other unmapped fields. Doing this can improve import performance.

6. If an entity has large number of records, split the DTS file into multiple DTS files by adding filters and running them in parallel to improve the performance. For example, if a contact has 50,000 records, split them into 10 jobs with 5,000 records per job. Another example is to add a filter on entity id and also add sorting on the entity id.

7. Normally, the Scribe tool will process 1 record per 1 transaction. But you can increase this by using the Scribe Bulk Mode and CRM Bulk API.

   If the number of records is more than 100, use Scribe Bulk mode by selecting the option in the Configure Steps area of the tool. This processes 100 records in a transaction.



Step 1 of 1:

msitcrmtest online.contact Insert

Operation

Operation: Insert

Variable Source Field:

☑ Use Bulk mode

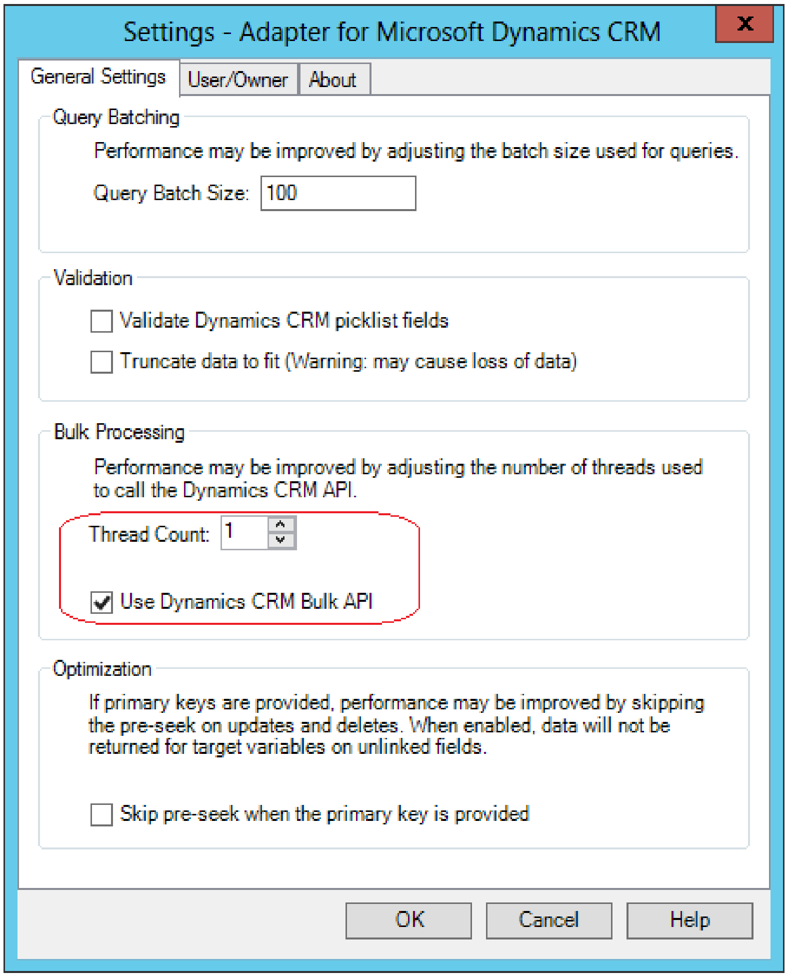
Number of rows per operation (10-1000): 100

Perform Update/Delete Operation

● Always

○ Conditionally when a Step field compares with a Source field or formula

8. If the number of records is more than 1,000, then enable CRM Bulk API, by opening the Target adapter settings and use the settings described here in addition to the Scribe Bulk Mode setting explained previously.



Query Batching

Performance may be improved by adjusting the batch size used for queries.

Query Batch Size: 100

Validation

□ Validate Dynamics CRM picklist fields

□ Truncate data to fit (Warning: may cause loss of data)

Bulk Processing

Performance may be improved by adjusting the number of threads used to call the Dynamics CRM API.

Thread Count: 1

☑ Use Dynamics CRM Bulk API

Optimization

If primary keys are provided, performance may be improved by skipping the pre-seek on updates and deletes. When enabled, data will not be returned for target variables on unlinked fields.

□ Skip pre-seek when the primary key is provided

If an entity has large number of records, split them into multiple jobs as mentioned in step 6 and enable bulk mode and CRM Bulk API settings. Then process all jobs in parallel.

You can process around 20 parallel jobs for each server and even more by running jobs simultaneously from multiple servers.

Note Select the “Skip pre-seek when the primary key is provided” option for Update/Delete jobs.

For more information, see the following web pages.

- http://www.scribesoft.com/helplibrary/prod/en/Subsystems/Insight/configtarget/bulk_operations.htm
- http://www.scribesoft.com/helplibrary/prod/en/Subsystems/AdapterForDynamicsCRM/adapterbehavior/bulk_processing.htm

2) A few tips about how to use the Scribe Workbench

1. **Boolean field migration**
   Scribe tool gives the Boolean value as “-1” instead of “1” at target side. Use this formula to resolve this issue:
   if(S1=-1,"1",S1)
2. **System User**
   a. Users who have an empty Primary Email field cannot be imported. Use this formula to fill a dummy email Id in the empty Primary Email field.
   IF (ISERROR(S1), "DummyEmail@Test.com",S1)
   ISERROR will return true if the S1 field is null otherwise return false.
   S1 -> Email field reference number at source.
   b. For creating Users Synchronized with Office 365, you must set “*IsSyncwithDirectory”* field value to “1”.

3. **System User Roles**
   Role IDs are different between the source and destination. Therefore, you must ensure this is adjusted by using this formula when you migrate the “System User Roles” into Microsoft Dynamics CRM Online.

   DBLOOKUP2(DBLOOKUP(S1, "OP Connection Name", "role", "roleid", "name"),S5,"OL Connection Name","role", "name", "businessunitidname","roleid")

4. Map the “createdon” field from Source to the “overriddencreatedon” field in Target to ensure that the Created On date can be preserved during import of records.

5. **Manage lookups for entities having default records.**
   In Microsoft Dynamics CRM Online, some entities such as currencies, subjects, and business units have default records. Here are the steps to use while mapping these fields in the DTS file:
   a. Migrate the records into Microsoft Dynamics CRM Online, except default records.
   b. Use this formula in the DTS file:
   if(S130="{On-PremiseDefaultRecordGUIDValue}","{ OnlineDefaultRecordGUIDValue }", S130)

6. **ObjectTypeCode formula for annotations, activities**
   “ObjectTypeCodes” are different for custom entities in both CRM (on-premises) and CRM Online. Therefore, you must manage these typecodes with an IF formula as follows:

   if(S56=10001,"10006",if(S56=10011,"10010",if(S56=10013,"10004",if(S56=10014,"10005",if(S56=10003,"10009",if(S56=10012,"10012",if(S56=10002,"10007",if(S56=10010,"10008",S56))))))))

7. **Owner mapping**
   Check whether all record owners (users) exist in CRM Online. If not, map them to a default user. If the total number of missing owners is few, then use an IF expression to map to the default user. Otherwise, use the formula here.

   DBLOOKUPDEFAULT(S156 , "OL Connection Name", "systemuser", "systemuserid", "systemuserid", "{DF574253-4E68-E311-940D-002DD80F0157}")

8. **Remove the mapping for system calculated fields**
   Some Microsoft Dynamics CRM entities have system calculated fields that can’t be updated using Scribe DTS files. Therefore, you must remove the mapping for these fields as part of the migration.
   For example: Opportunity entity has system calculated fields such as total amount.

9. **Remove the mapping for state and status code fields in activities, opportunity, Invoice, quotes, order.**
   You can’t create the activity parties and attachments for closed activities. Also you cannot add line items for closed opportunities, invoices, orders and quotes. Therefore, you must remove the mapping for these fields.

10. **Activity Parties**

➔ Always use "ODBC connection" for Source and write a SQL query to filter out owner and regarding fields as they will be created by-default with Activity record creation.

*SELECT * FROM ActivityParty WHERE ParticipationTypeMask<>8 AND ParticipationTypeMask<>9 AND ActivityId IN (SELECT ActivityId FROM Email)*

➔ Activity Parties have two extra fields in the target as "ActivtyTypeCode" and "ActivityFieldName."

a. ActivityTypeCode:
You must migrate the activity parties of each activity separately so that you can hardcode this value in the Scribe DTS file. For example, Email – 4202.

b. Activity Field Name:
The Source will have "Field display names" (participationtypemaskname), so you must use this formula to change them to a logical name.

if(S5=2,"to", if(S5=3, "cc", if(S5=4, "bcc", if(S5=1,"from", if(S5=5,"requiredattendees", if(S5=6,"optionalattendees", if(S5=7,"organizer", if(S5=12,"partners", S5), if(S5=11,"customer", S5))))))))

For more information about the Activity Party entity, see [http://msdn.microsoft.com/en-us/library/gg328549(v=crm.6).aspx.](http://msdn.microsoft.com/en-us/library/gg328549(v=crm.6).aspx)

11. **Email attachments:** Steps to migrate the "Email Attachments."
    a. Configure the DTS file with the Source as "ODBC Connection" and query from "ActivityAttachment" object and Target with the Microsoft Dynamics CRM 2011 or Microsoft Dynamics CRM 2013 Adapter connecting to the "activitymimeattachment" entity.
    b. Auto map all the fields and accept all mappings.
    c. Remove "AttachmentID" field mappings.

       Microsoft Dynamics CRM automatically creates an "Attachmentid" on the creation of the "activitymimeattachment" record and updates the created "Attachmentid" in the "ActivityMimeAttachment" table.

       Therefore, if you map this field, you will have different "AttachmentIDs" in both the table's attachment and activitymimeattachment.

       If you don't remove the mapping, users won't see the corresponding attachment in the email activity.
    d. Either remove the mapping for "ActivityId" or Objectid and ObjectTypecode.

       Both mappings represent the same value.
12. Exclude the statecode and Statuscode fields in the Insert DTS file if you have Update operations on the entity.

13. **Managing failed records in the Scribe Internal database for further review and action:** To configure the internal database for storing the failed records during import so that they can be reviewed, corrected as needed, and reimported again, follow these steps.
    a. Click the setting icon of the DTS file
    b. Locate the "Rejected Source rows" tab
    c. Select the option "output rejected source rows"
    d. Provide a prefix for the table. The Scribe tool will create a table with the name as "Prefix + currentdatetime"
    e. The table will be created by using the current source entity columns together with error message.
    f. Perform the data correction based on the error messages.
    g. Run the DTS file by using the table as the "Source" that reduces the execution time because it will execute only the failed records.

We recommend that you provide a clear and unambiguous name for the rejected source rows table to avoid any confusion processing these records. For example, use the naming convention *InstanceName_EntityName_Operation*.

3) Scribe Tool – Field mapping

1. Make sure to open the Scribe Workbench as “Administrator,” to ensure that any settings modified in the DTS file can be saved without any issues.

2. Add a Data Link from “createdon” field (source) to “overriddencreatedon” field (target).

3. Click Links, select the Auto Link option, and then select Link fields that have exact name matches.

4. Next, click the Accept All Auto Links option.

5. If any entity reference data is not available at the time of import, remove the State Code and Status Code mappings. Run an update on account after Contact import to update this field and status fields.

6. Remove the mappings for system calculated fields, such as “total amount in opportunity.”

7. Add formulas for required target fields, such as the following:

    a. Boolean fields

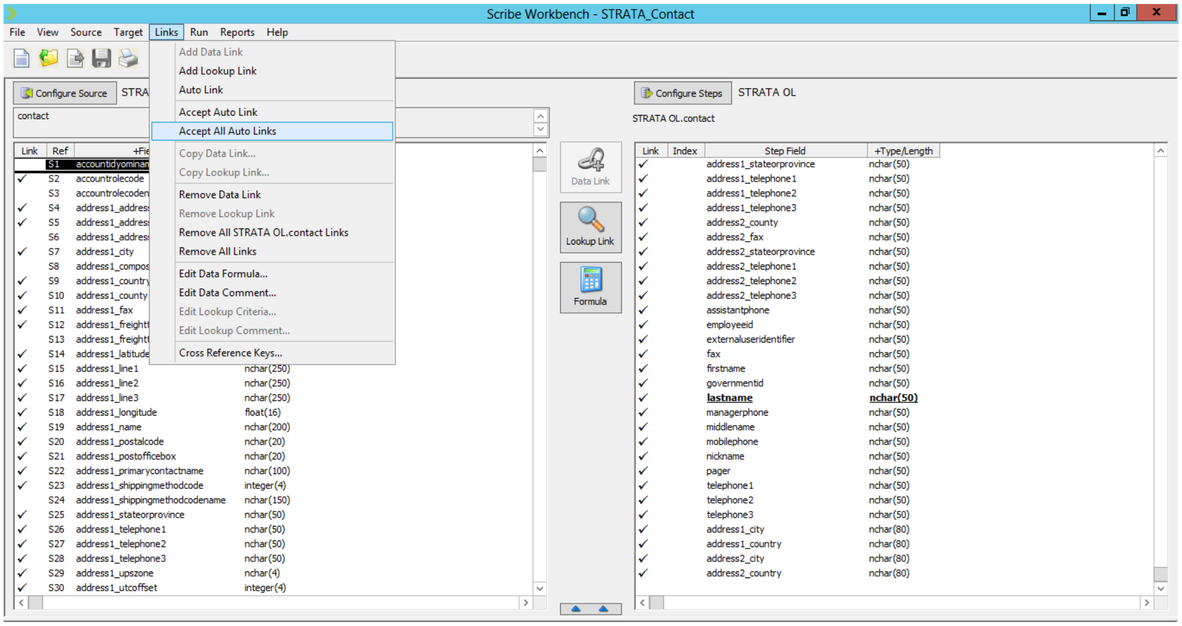
    b. Currency

    c. Business Units

8. Review each system job to verify the actual data mapping, and then run the job.

9. We recommend that you always use the “Auto link” option to map the source and destination fields. This helps map all the source fields to target fields together (in a single step). This saves the time that is required to map the fields one-by-one.

    Here is a screen capture of the Auto Link option in the Scribe tool.



contact

Link	Ref	+Fie	
	S1	accountidyominan	
✓	S2	accountrolecode	
	S3	accountrolecoden	
✓	S4	address1_addres	
✓	S5	address1_addres	
	S6	address1_addres	
✓	S7	address1_city	
	S8	address1_compos	
✓	S9	address1_country	
✓	S10	address1_county	
✓	S11	address1_fax	
✓	S12	address1_freightt	
	S13	address1_freightt	
✓	S14	address1_latitude	
✓	S15	address1_line1	nchar(250)
✓	S16	address1_line2	nchar(250)
✓	S17	address1_line3	nchar(250)
✓	S18	address1_longitude	float(16)
✓	S19	address1_name	nchar(200)
✓	S20	address1_postalcode	nchar(20)
✓	S21	address1_postofficebox	nchar(20)
✓	S22	address1_primarycontactname	nchar(100)
✓	S23	address1_shippingmethodcode	integer(4)
	S24	address1_shippingmethodcodename	nchar(150)
✓	S25	address1_stateorprovince	nchar(50)
✓	S26	address1_telephone1	nchar(50)
✓	S27	address1_telephone2	nchar(50)
✓	S28	address1_telephone3	nchar(50)
✓	S29	address1_upszone	nchar(4)
✓	S30	address1_utcoffset	integer(4)

STRATA OL.contact

Link	Index	Step Field	+Type/Length
✓		address1_stateorprovince	nchar(50)
✓		address1_telephone1	nchar(50)
✓		address1_telephone2	nchar(50)
✓		address1_telephone3	nchar(50)
✓		address2_county	nchar(50)
✓		address2_fax	nchar(50)
✓		address2_stateorprovince	nchar(50)
✓		address2_telephone1	nchar(50)
✓		address2_telephone2	nchar(50)
✓		address2_telephone3	nchar(50)
✓		assistantphone	nchar(50)
✓		employeeid	nchar(50)
✓		externaluseridentifier	nchar(50)
✓		fax	nchar(50)
✓		firstname	nchar(50)
✓		governmentid	nchar(50)
✓		**lastname**	**nchar(50)**
✓		managerphone	nchar(50)
✓		middlename	nchar(50)
✓		mobilephone	nchar(50)
✓		nickname	nchar(50)
✓		pager	nchar(50)
✓		telephone1	nchar(50)
✓		telephone2	nchar(50)
✓		telephone3	nchar(50)
✓		address1_city	nchar(80)
✓		address1_country	nchar(80)
✓		address2_city	nchar(80)
✓		address2_country	nchar(80)