

Snapshot Best Practices: Relational Data Migration

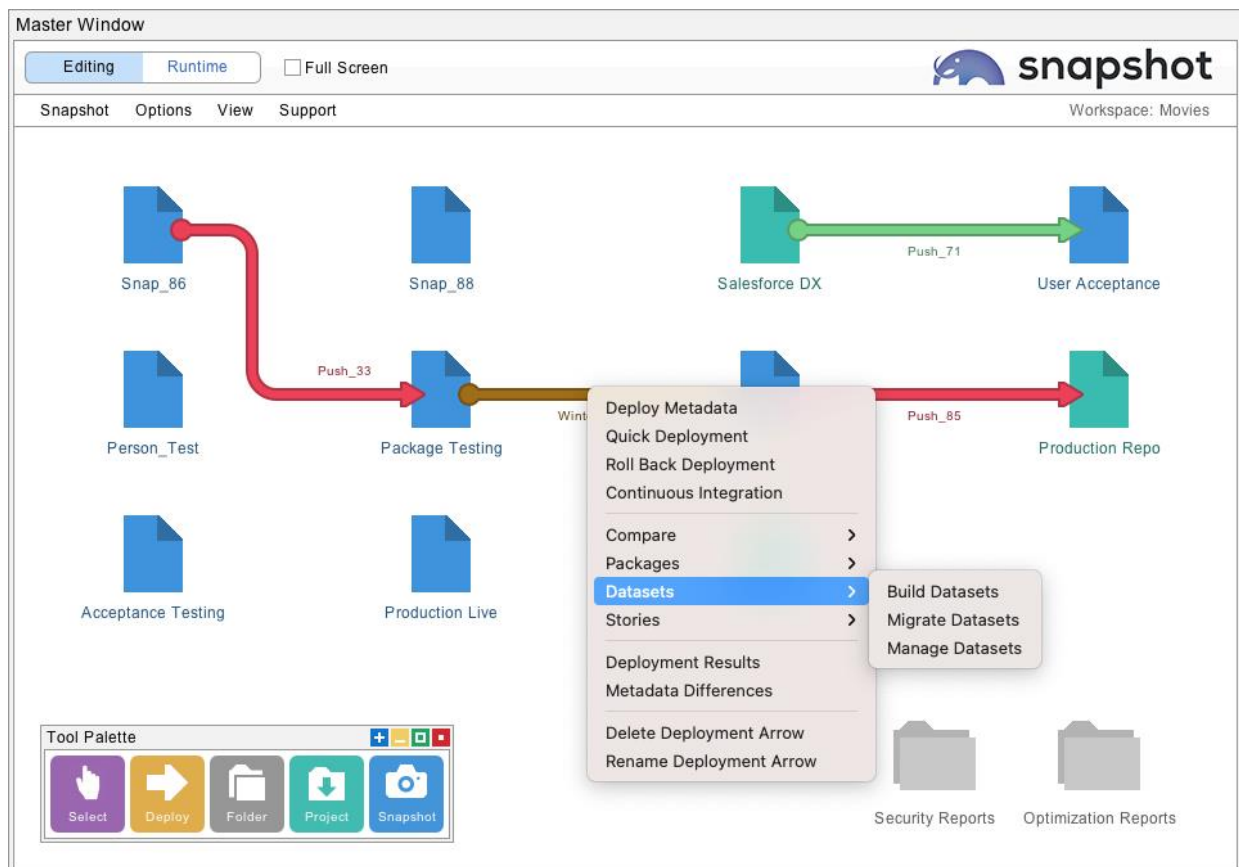


Snapshot provides powerful tools for migrating connected sets of data records between Salesforce orgs. This capability is useful for backing up data, refreshing Sandboxes, merging orgs, and populating orgs with test data for application development. The Bulk API is used for all transactions to ensure that very large datasets can be moved efficiently. We have moved millions of records while testing the data migration tools in Snapshot.

When records are migrated between orgs, all of the internal relationships are preserved. External references in the dataset are also connected to matching objects on the destination. This whitepaper presents detailed information on how to build and migrate datasets. We also document the CSV format that Snapshot uses for importing datasets from other systems or creating them with a spreadsheet editor.

Source and Destination Orgs

When you select the Deployment Arrow between any two connected Snapshots, the Options Menu will display various commands. You can also right-click the Deployment Arrow to see the options in a popup menu. The third submenu will have options to Build, Migrate, and Manage Datasets. If you do not see these options, then perhaps the Deployment Arrow is connected to a Developer Project. They do not have any actual data and cannot be used as a source to build a dataset or as a destination for migration.



The option to Build Datasets will use the source Salesforce org to download multiple records in the form of XML files to your local machine. The option to Migrate Datasets will insert and update records from the selected dataset into the destination Salesforce org. The option to Manage Datasets can be used to create a dataset from CSV files, remap field names, remap field values, create external IDs, rename the dataset, duplicate the dataset, import CSV files, and export CSV files.

Build Datasets Dialog

The first tab of the Build Dataset dialog allows you to select the dataset and template that you would like to work with. If you are moving records associated with a managed package, then check out the Prebuilt Templates menu for help moving those records. For example, Salesforce CPQ, Health Cloud, and Government Cloud all have custom templates.

Build Datasets For Winter Release (Source: person1@metazoa.com)

Select Dataset

Select Parents

Select Children

Loaded Fields

Build Dataset

Schedule Build

Create a new dataset or replace an existing one:

☐ Create New Dataset

☒ Replace Existing Dataset

test_dataset

Create a new template, use an existing template, or select a prebuilt template:

☐ Create New Template

☒ Use Existing Template

☐ Select Prebuilt Template

Import Template

Export Template

test_dataset

Account Case

Dataset: test_dataset, Source: person1@metazoa.com, Built: 12/15/2021 8:03 AM

▼ Account (Select All)

▼ Opportunity (AccountId)

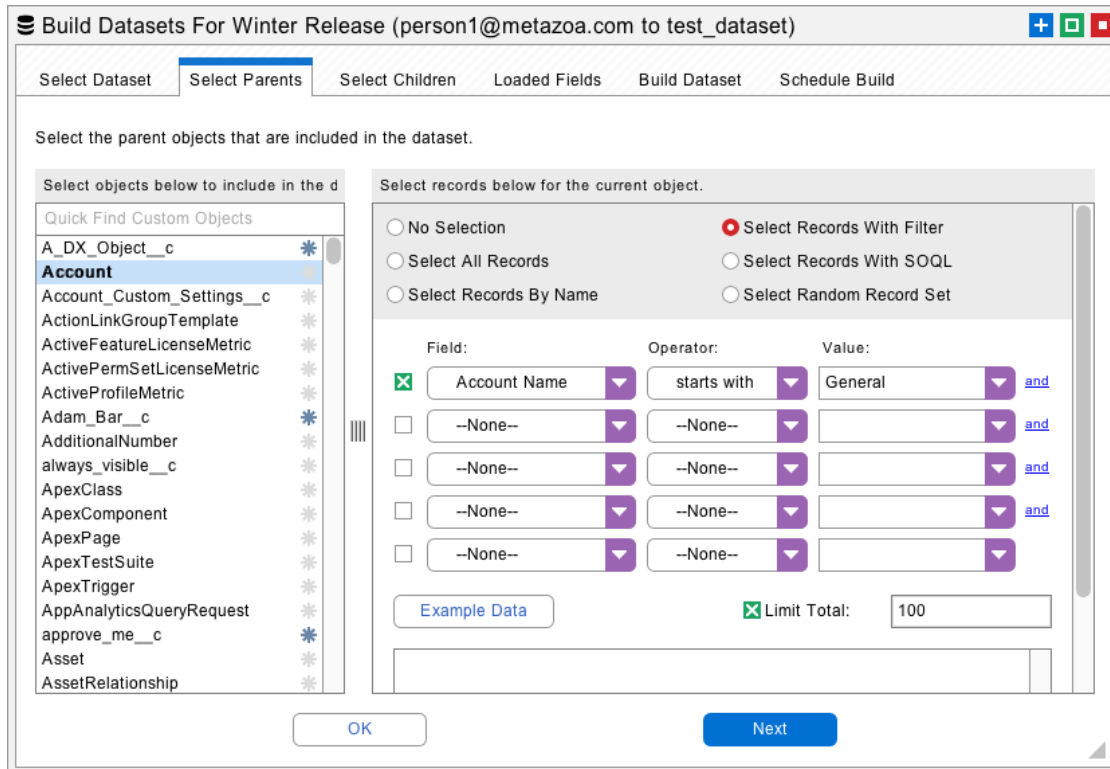
OpportunityLineItem (OpportunityId)

Cancel

Next

Selecting Parents and Children

The next tab allows you to select the parent records that you want to include in the dataset. These records are available on the source Salesforce org. You can select all records, a subset of records by name, or a subset of records using a complex filter. There is also an option to write a SOQL query by hand. The total number of downloaded records can be limited. This is useful for grabbing a random subset of records for acceptance testing or application development.



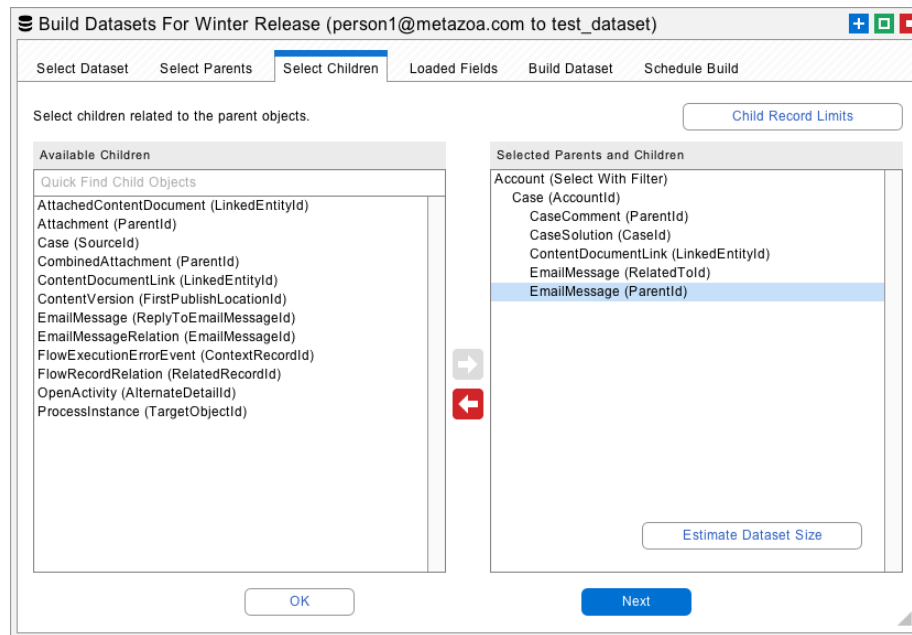
The screenshot shows a web application window titled "Build Datasets For Winter Release (person1@metazoa.com to test_dataset)". The interface has a tabbed navigation bar with the following tabs: "Select Dataset", "Select Parents" (active), "Select Children", "Loaded Fields", "Build Dataset", and "Schedule Build".

Below the tabs, the instruction "Select the parent objects that are included in the dataset." is displayed. The main area is divided into two panels:

- Left Panel:** Titled "Select objects below to include in the d", it contains a "Quick Find Custom Objects" search bar and a list of Salesforce objects. The "Account" object is selected and highlighted in blue. Other objects listed include A_DX_Object__c, Account_Custom_Settings__c, ActionLinkGroupTemplate, ActiveFeatureLicenseMetric, ActivePermSetLicenseMetric, ActiveProfileMetric, Adam_Bar__c, AdditionalNumber, always_visible__c, ApexClass, ApexComponent, ApexPage, ApexTestSuite, ApexTrigger, AppAnalyticsQueryRequest, approve_me__c, Asset, and AssetRelationship.
- Right Panel:** Titled "Select records below for the current object.", it contains several options for record selection:
 - Radio buttons for "No Selection", "Select All Records", "Select Records By Name", "Select Records With Filter" (selected), "Select Records With SOQL", and "Select Random Record Set".
 - A filter builder table with columns "Field:", "Operator:", and "Value:". The first row is populated with "Account Name", "starts with", and "General". Subsequent rows are empty with "--None--" in the first two columns.
 - A "Limit Total:" section with a checkbox (checked) and a text input field containing "100".

At the bottom of the window, there are two buttons: "OK" and "Next".

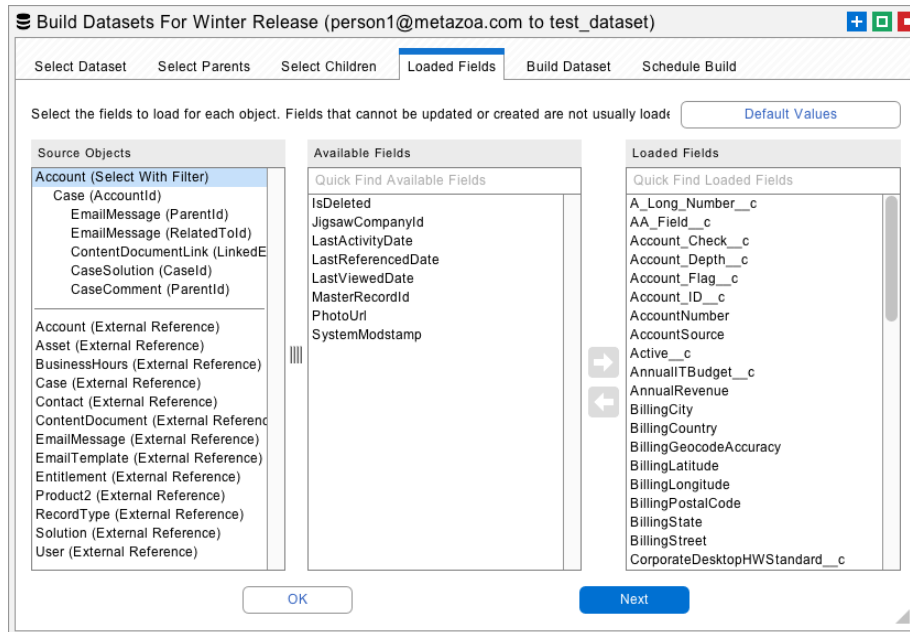
The next tab allows the selection of connected child objects for each parent object. When a dataset is created, the selected parent records are loaded first, followed by all the children connected to that parent. You can specify multiple child objects in a hierarchy. The relationship field used to associate each parent and child is shown in parenthesis. The internal relationships between parent and child are always preserved when the dataset is migrated.



Notice the Child Record Limits button at upper right. This feature allows you to add a filter or limit the number of records for the selected child object. Also see the Estimate Dataset Size button at lower right. This option will calculate the estimated number of records and total size of the dataset before it is actually created.

Loaded Fields

After that, you can select fields to load for each parent and child object. The parent and child objects that were selected will be above the dividing line at left. You can choose fields that need to be loaded by moving them to the list at right. Snapshot will automatically figure out the best fields to load, so you may not need to do anything on this tab. Fields that cannot be created or updated on the destination do not usually need to be loaded. Removing unwanted fields makes your dataset smaller in size and easier to migrate.



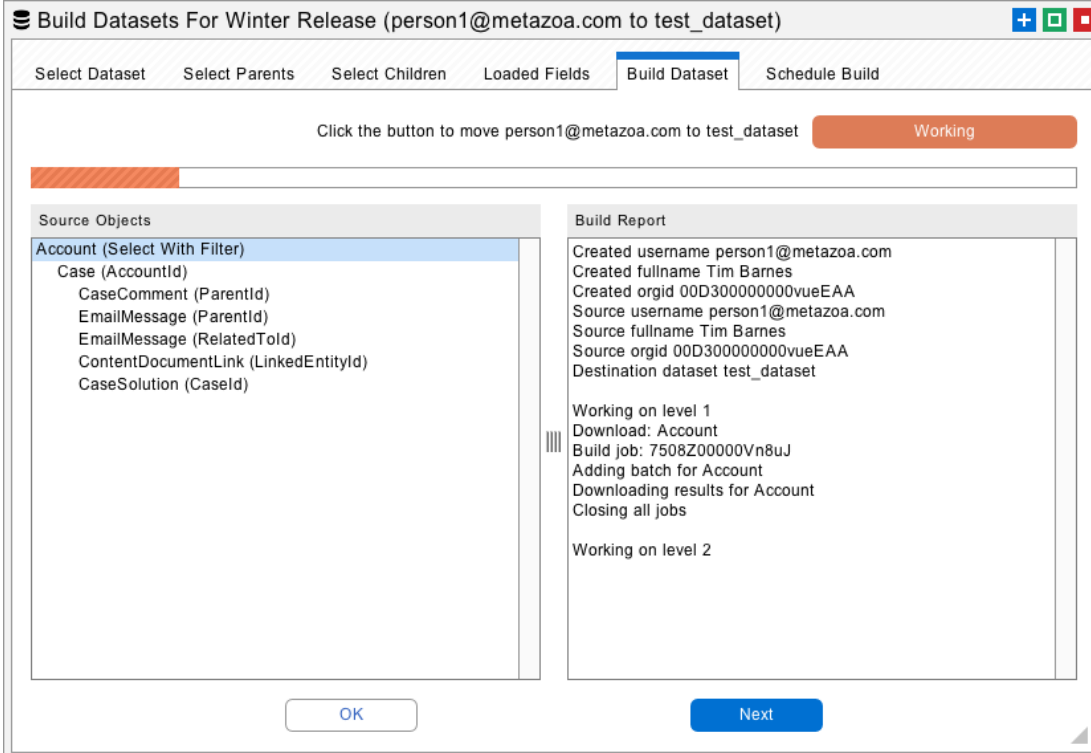
Snapshot also uses the loaded fields to identify matching records in the destination org during migration. If a source record matches a destination record, then the corresponding destination record is updated. If a source record does not match any destination record, then a new destination record is created.

Snapshot automatically selects the most common matching fields for you. However, you can also manually select matching fields. For example, if you want to match Accounts by Name and BillingCity, then be sure that both Name and BillingCity are loaded. You will be able to specify the exact list of matching fields that you want to use in the Migrate Datasets dialog.

Underneath the dividing line you will see “external references” to other objects in the destination org. For example, if you select Opportunity objects for migration then you will see an external reference to Campaigns, because the Opportunity object contains a CampaignId reference field. Snapshot will automatically connect external references to matching objects on the destination org when the dataset is migrated. You can manually specify additional fields needed for matching external references as well.

Build Datasets Button

The next tab allows you to enter the name of a new dataset and then click the Build Datasets button at right to start the download process. If you select an existing dataset name from the menu then that dataset will be replaced. All of the download results will be listed in the window pane at lower right.



The XML files with your dataset information will be saved in the “datasets” folder next to the “workspaces” folder in the Snapshot file system. Datasets are globally available for migration to any destination org. In other words, any source org can be used to build a dataset, and any dataset can be migrated to the destination org.

The last tab allows you to Build Datasets at a specific time in the future or as a recurring event. When a scheduled dataset is created, it will automatically replace the current dataset by name. Unlike Snapshots, datasets are not maintained in a time series.

Manage Datasets Dialog

The Manage Datasets dialog has some very helpful utilities for creating and transforming your datasets. From this interface you can create a dataset from CSV files, remap field names, remap field values, create external IDs, rename the dataset, duplicate the dataset, import CSV files, and export CSV files. The next section discusses the CSV file format that is used by the create, import, and export options. If you don't need that information right now, you might want to skip the next section. After that, we discuss all the options in the Manage Datasets interface.

Manage Datasets For Winter Release

Manage Datasets

Compare Datasets

Display Report

Schedule Report

Available Datasets

Quick Find Available Datasets

account_data

data_archive

developer_test_data

production_backup

sandbox_seed_data

test_data

Dataset Objects

Account (Select All)

Opportunity (AccountId)

OpportunityLineItem (OpportunityId)

Management Options

Create

Remap

External

Rename

Duplicate

Delete

Import

Export

Dataset: account_data

Arrow Name: Winter Release

Username: person1@metazoa.com

Source Org ID: 00D300000000vueEAA

Source Org Name: Genericcorp

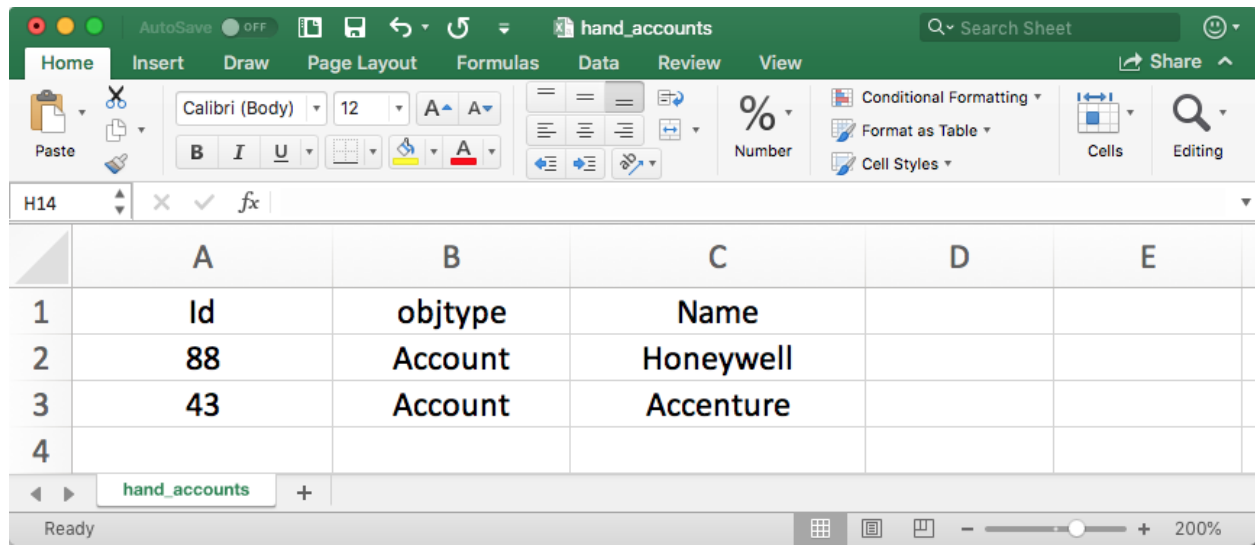
Built: 12/15/2021 8:03 AM

OK

Next

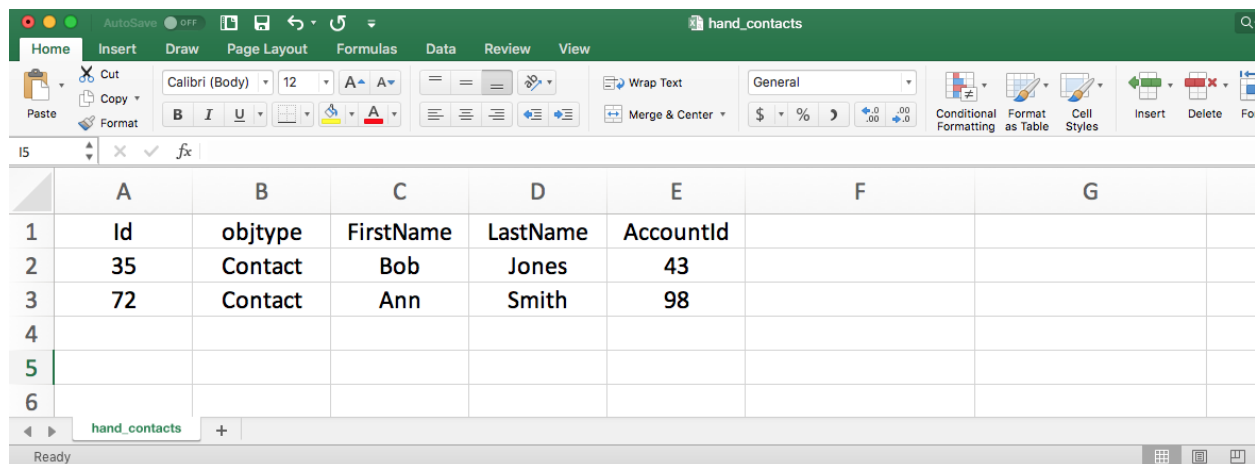
CSV File Format

The CSV file format encodes all the field values as well as the internal and external relationships in the dataset. This format expects the first row of CSV data to be field names, followed by additional rows for each record. The columns must include the field name “ID” to specify a unique record ID, and the field name “objtype” to specify the object type. Here is an example of the CSV file format with two Account records:



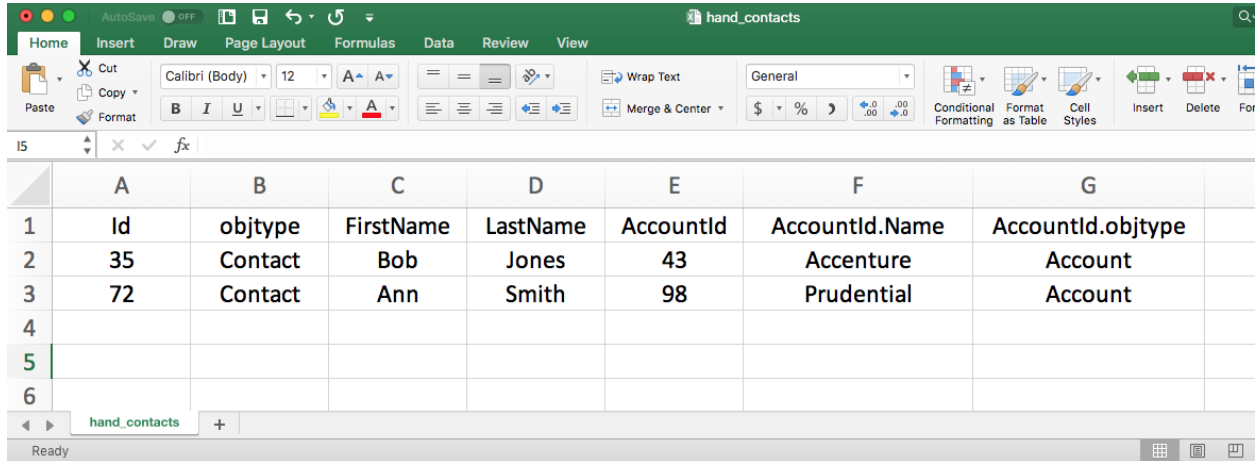
	A	B	C	D	E
1	Id	objtype	Name		
2	88	Account	Honeywell		
3	43	Account	Accenture		
4					

The rest of the columns are used for other fields like the Account Name. In this example, the ID field is from some other system, because these are obviously not 18-character Salesforce IDs. The IDs can be in any format, but they must be self-consistent for all the internal relationships to be maintained. Now let’s look at another imported CSV file with Contact data that refers to the Account records above:



	A	B	C	D	E	F	G
1	Id	objtype	FirstName	LastName	AccountId		
2	35	Contact	Bob	Jones	43		
3	72	Contact	Ann	Smith	98		
4							
5							
6							

The Contact data contains a reference to a parent Account record. Snapshot will connect each Contact to any Account with that ID in the destination org. In our example, the Contact record for Bob Jones will be properly connected to the Account record for Accenture because they both have the same Id = 43.



	A	B	C	D	E	F	G
1	Id	objtype	FirstName	LastName	AccountId	AccountId.Name	AccountId.objtype
2	35	Contact	Bob	Jones	43	Accenture	Account
3	72	Contact	Ann	Smith	98	Prudential	Account
4							
5							
6							

Here is another version of the Contact CSV file that includes better matching information for the parent Account records. If you include the AccountId.objtype and AccountId.Name fields, then these fields will be available in the Migrate Datasets dialog to match destination objects with the same object type and name. In our example, one Contact will connect to Accenture by ID, and the other Contact will connect to Prudential by Name, if such a record exists in the destination org.

In this manner, Snapshot will use the imported CSV data to match both internal and external references. The required reference information is usually easy to include in the CSV file. For example, you could use the Salesforce SOQL Query below to capture all the information needed to create our example CSV data for Accounts and Contacts:

```
SELECT Id,Name FROM Account
```

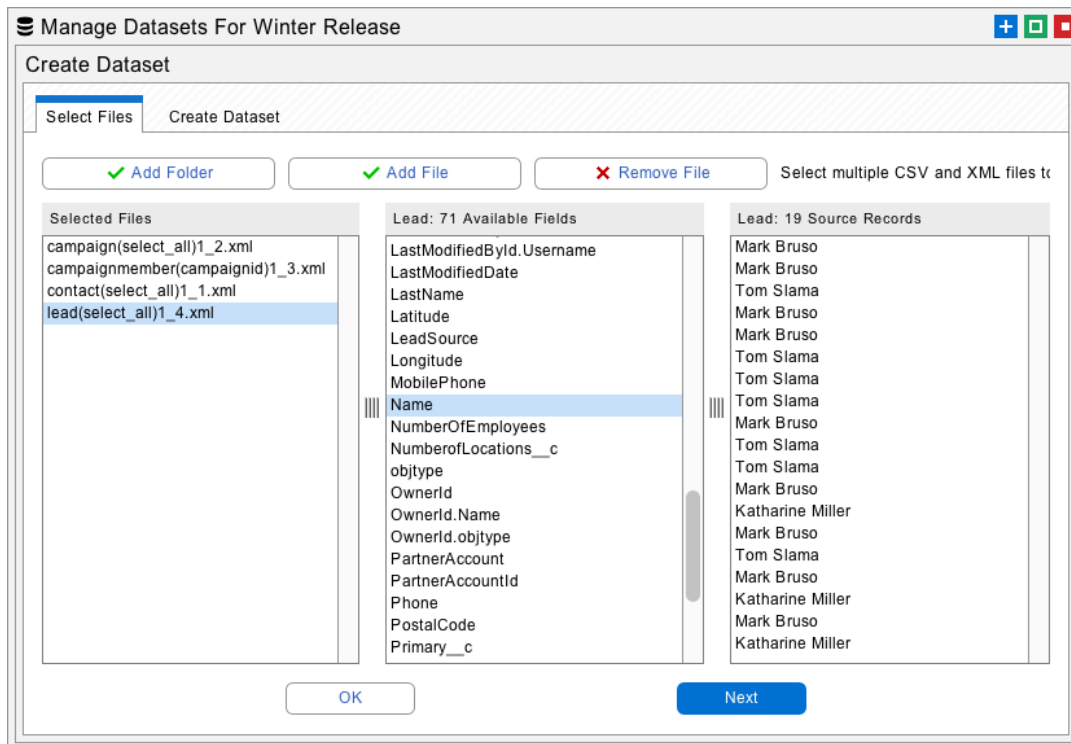
```
SELECT Id,FirstName,LastName,AccountId,Account.Name FROM Contact
```

Create Datasets

Click the Create button on the Manage Datasets dialog to open an interface that allows you to import files and folders. This information will be used to create a new dataset. The files should be in the CSV file format discussed above. You can also import XML files. Simply go to the dataset folder and select any group of XML files from an existing dataset. This is useful if you want to migrate an existing dataset in sections.

The imported files are added to the list at left. You can right-click the items here and adjust their order. This is useful for making sure that parent objects are processed before child objects. In the previous example, you would want to make sure that Accounts are listed before Contacts. The imported fields and source records are visible in the lists at right.

To create a new dataset, go to the next tab, give your new dataset a name, and click the Create Dataset button. This will add the new dataset to the list on the first tab of the Manage Datasets dialog.



Manage Datasets For Winter Release

Create Dataset

Select Files | Create Dataset

✓ Add Folder | ✓ Add File | ✗ Remove File | Select multiple CSV and XML files to

Selected Files	Lead: 71 Available Fields	Lead: 19 Source Records
campaign(select_all)1_2.xml	LastModifiedById.Username	Mark Bruso
campaignmember(campaignid)1_3.xml	LastModifiedDate	Mark Bruso
contact(select_all)1_1.xml	LastName	Tom Slama
lead(select_all)1_4.xml	Latitude	Mark Bruso
	LeadSource	Mark Bruso
	Longitude	Tom Slama
	MobilePhone	Tom Slama
	Name	Tom Slama
	NumberOfEmployees	Mark Bruso
	NumberOfLocations__c	Tom Slama
	objtype	Tom Slama
	OwnerId	Mark Bruso
	OwnerId.Name	Katharine Miller
	OwnerId.objtype	Mark Bruso
	PartnerAccount	Tom Slama
	PartnerAccountId	Mark Bruso
	Phone	Katharine Miller
	PostalCode	Mark Bruso
	Primary__c	Katharine Miller

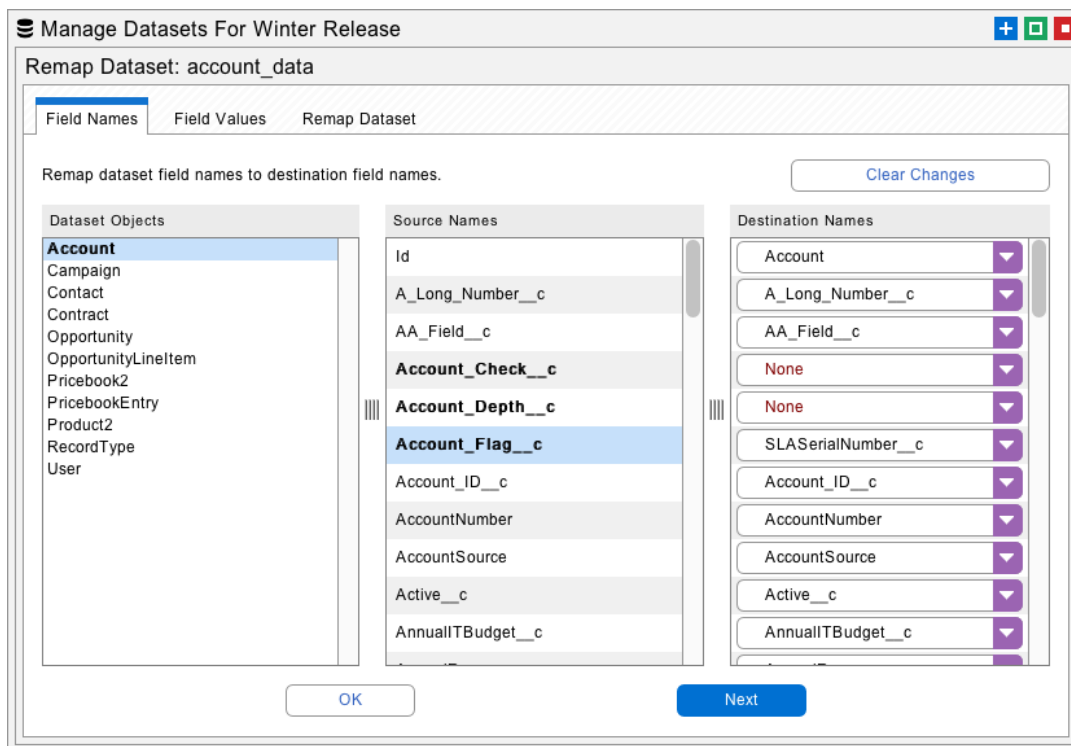
OK | Next

Remapping Field Names

Click the Remap button on the Manage Datasets dialog to open an interface that will remap field names and values. Remapping field names is useful when you want to migrate the data between fields with different names. For example, you might need to merge two different custom objects.

To use this feature, select the object to be remapped at left. Select the field name to remap in the center and select the destination field at right. The fields at right will be defined by the available objects and fields in the destination org. You might have to select an arrow that has the desired destination org for this to work as expected.

Proceed to the Remap Dataset tab and click the Remap Dataset button. This will transform all the field names in the selected dataset. Now the field names in this dataset will match the destination org.



Manage Datasets For Winter Release

Remap Dataset: account_data

Field Names | Field Values | Remap Dataset

Remap dataset field names to destination field names. [Clear Changes](#)

Dataset Objects	Source Names	Destination Names
Account	Id	Account
Campaign	A_Long_Number__c	A_Long_Number__c
Contact	AA_Field__c	AA_Field__c
Contract	Account_Check__c	None
Opportunity	Account_Depth__c	None
OpportunityLineItem	Account_Flag__c	SLASerialNumber__c
Pricebook2	Account_ID__c	Account_ID__c
PricebookEntry	AccountNumber	AccountNumber
Product2	AccountSource	AccountSource
RecordType	Active__c	Active__c
User	AnnualTBudget__c	AnnualTBudget__c

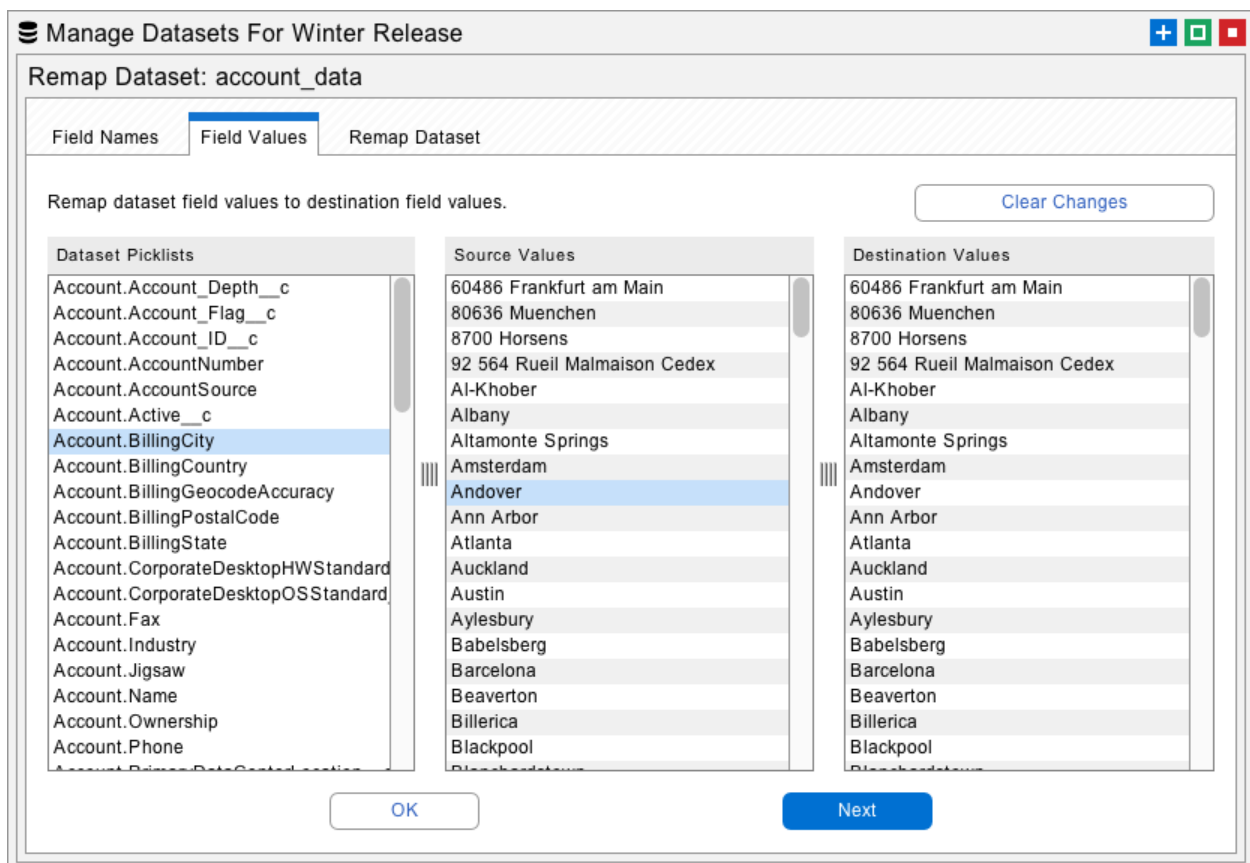
OK Next

Remapping Field Values

Click the Remap button on the Manage Datasets dialog to open an interface that will remap field names and values. Remapping field values is useful when there are username transformations, picklist differences, and email suffixes that need to be matched against the destination org.

To use this feature, select the object and field to be remapped at left. Select the source value in the center and type in a destination value at right. You can import and export all the destination values as text files. Right-click the destination for editing options.

Proceed to the Remap Dataset tab and click the Remap Dataset button. This will transform all the field values in the selected dataset. Now this dataset can be migrated to the destination and the new field values will match.



Manage Datasets For Winter Release

Remap Dataset: account_data

Field Names | **Field Values** | Remap Dataset

Remap dataset field values to destination field values. [Clear Changes](#)

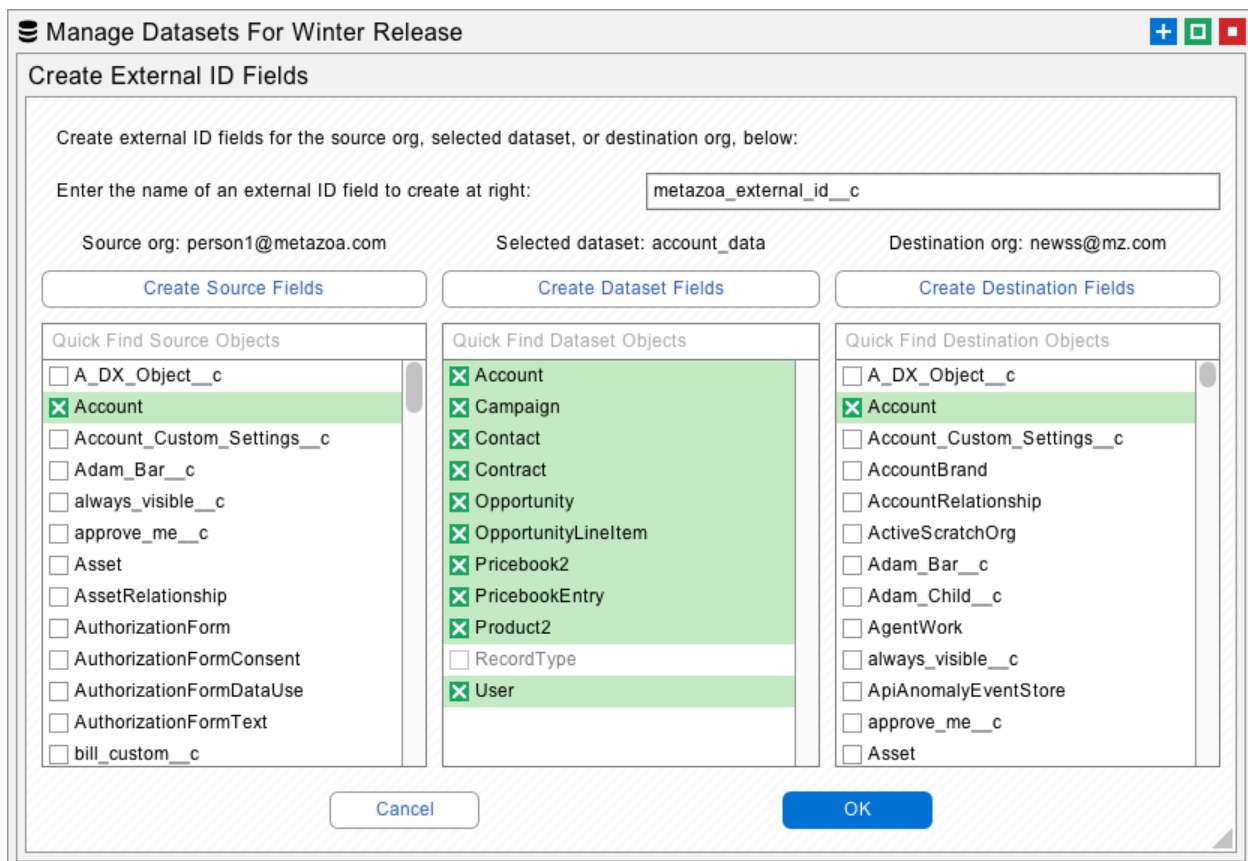
Dataset Picklists	Source Values	Destination Values
Account.Account_Depth__c	60486 Frankfurt am Main	60486 Frankfurt am Main
Account.Account_Flag__c	80636 Muenchen	80636 Muenchen
Account.Account_ID__c	8700 Horsens	8700 Horsens
Account.AccountNumber	92 564 Rueil Malmaison Cedex	92 564 Rueil Malmaison Cedex
Account.AccountSource	Al-Khober	Al-Khober
Account.Active__c	Albany	Albany
Account.BillingCity	Altamonte Springs	Altamonte Springs
Account.BillingCountry	Amsterdam	Amsterdam
Account.BillingGeocodeAccuracy	Andover	Andover
Account.BillingPostalCode	Ann Arbor	Ann Arbor
Account.BillingState	Atlanta	Atlanta
Account.CorporateDesktopHWStandard	Auckland	Auckland
Account.CorporateDesktopOSStandard	Austin	Austin
Account.Fax	Aylesbury	Aylesbury
Account.Industry	Babelsberg	Babelsberg
Account.Jigsaw	Barcelona	Barcelona
Account.Name	Beaverton	Beaverton
Account.Ownership	Billerica	Billerica
Account.Phone	Blackpool	Blackpool

[OK](#) [Next](#)

Create External IDs

Click the External button on the Manage Datasets dialog to open an interface that will create external IDs on the source org, the dataset itself, or the destination org. When you create an external ID on the source org, Snapshot creates a custom **formula** field that sets the field value equal to the object ID. When you create an external ID on the destination org, then Snapshot creates a custom **text** field that receives the source ID value. You can give the new custom field any name that you like. Field Level Security is automatically set for the System Administrator Profile and the field is hidden from everyone else.

Sometimes administrators don't want to create a bunch of external IDs in the source org. That might be disruptive if the org is in active use. In this case, Snapshot offers the capability to create an external ID on the dataset itself, as if the external ID was actually from the source org. This works exactly the same way when the destination records are tagged or matched with the source ID. The beauty of this is that you can fully exploit the power of migrating data with external IDs without actually having any external ID fields in the source org.



Manage Datasets For Winter Release

Create External ID Fields

Create external ID fields for the source org, selected dataset, or destination org, below:

Enter the name of an external ID field to create at right:

Source org: person1@metazoa.com Selected dataset: account_data Destination org: newss@mz.com

[Create Source Fields](#) [Create Dataset Fields](#) [Create Destination Fields](#)

Quick Find Source Objects

- ☐ A_DX_Object__c
- ☒ Account
- ☐ Account_Custom_Settings__c
- ☐ Adam_Bar__c
- ☐ always_visible__c
- ☐ approve_me__c
- ☐ Asset
- ☐ AssetRelationship
- ☐ AuthorizationForm
- ☐ AuthorizationFormConsent
- ☐ AuthorizationFormDataUse
- ☐ AuthorizationFormText
- ☐ bill_custom__c

Quick Find Dataset Objects

- ☒ Account
- ☒ Campaign
- ☒ Contact
- ☒ Contract
- ☒ Opportunity
- ☒ OpportunityLineItem
- ☒ Pricebook2
- ☒ PricebookEntry
- ☒ Product2
- ☐ RecordType
- ☒ User

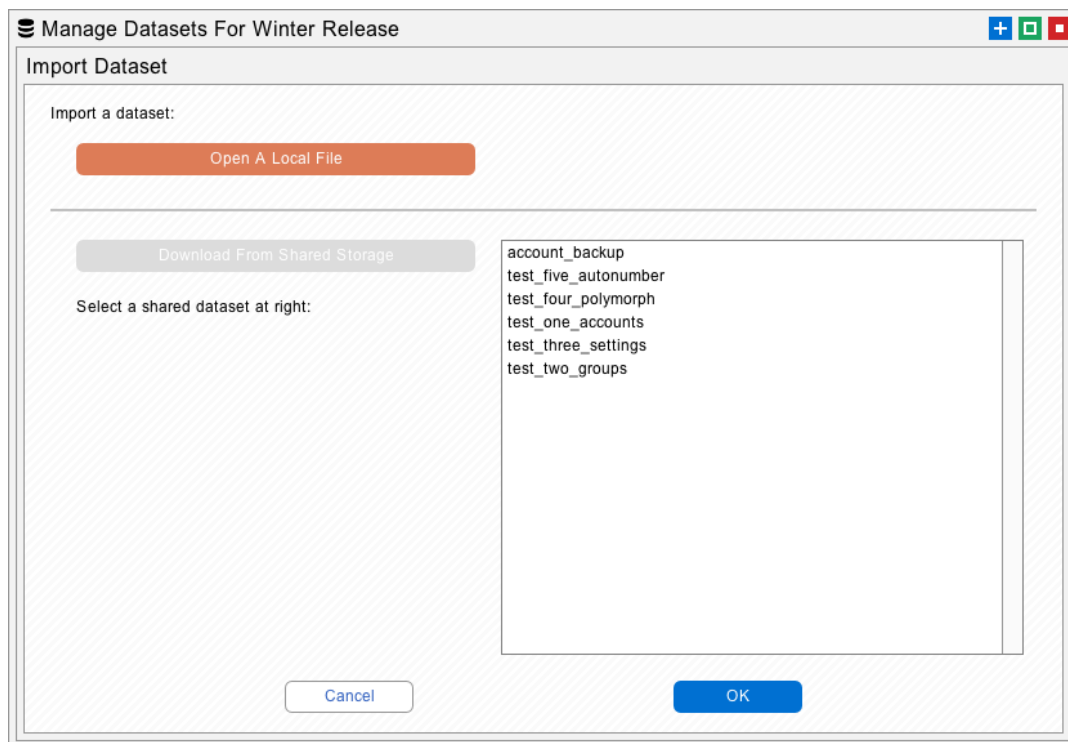
Quick Find Destination Objects

- ☐ A_DX_Object__c
- ☒ Account
- ☐ Account_Custom_Settings__c
- ☐ AccountBrand
- ☐ AccountRelationship
- ☐ ActiveScratchOrg
- ☐ Adam_Bar__c
- ☐ Adam_Child__c
- ☐ AgentWork
- ☐ always_visible__c
- ☐ ApiAnomalyEventStore
- ☐ approve_me__c
- ☐ Asset

[Cancel](#) [OK](#)

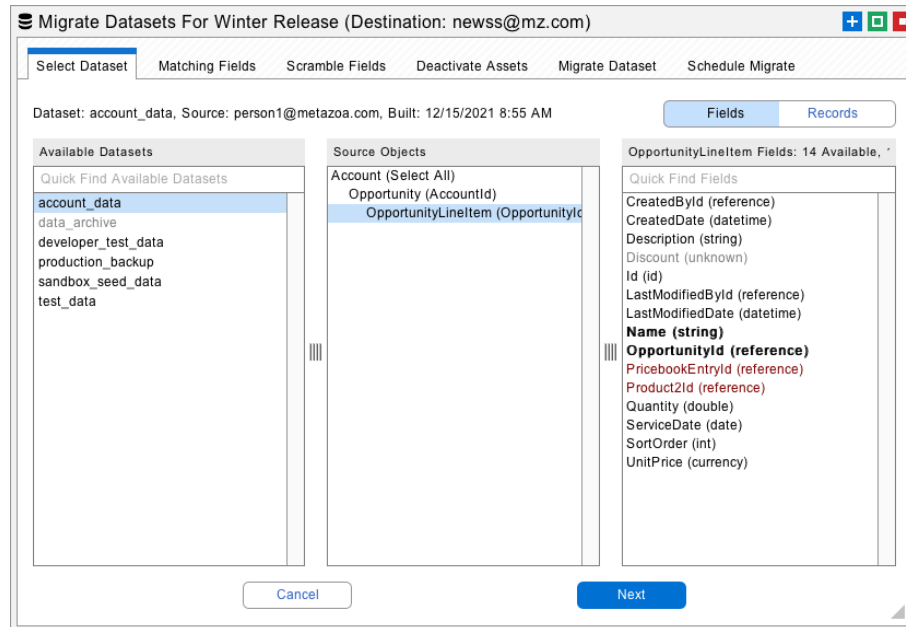
Import and Export Datasets

Click the Import button on the Manage Datasets dialog to open an interface that will import a dataset as a local file or from shared storage. Click the Export button to open an interface that will export a dataset as a local file or to shared storage. The local files will be saved as a zipped dataset folder. The shared storage uses Salesforce Content folders and documents that are saved in the Snapshot License Org. This is an easy way to archive datasets or share them with your administrative team.



Migrate Datasets Dialog

After a dataset has been created, you are ready to migrate these records to a destination Salesforce org. Right-click a Deployment Arrow that is connected to the correct destination org and select the Migrate Datasets option to get started. The Migrate Datasets dialog allows you to select any of the global datasets from the list at left and see the objects and fields that are available in the list at right. The next four tabs provide options for matching fields, scrambling fields, deactivating assets, and finally migrating the selected dataset.



Here is a power user tip. You can right-click any of the objects in the middle list and export the data as an XML or CSV file. These files will be in the correct format for the Import Datasets dialog. For example, you could export a CSV file, edit the file as a spreadsheet, and then import your changes.

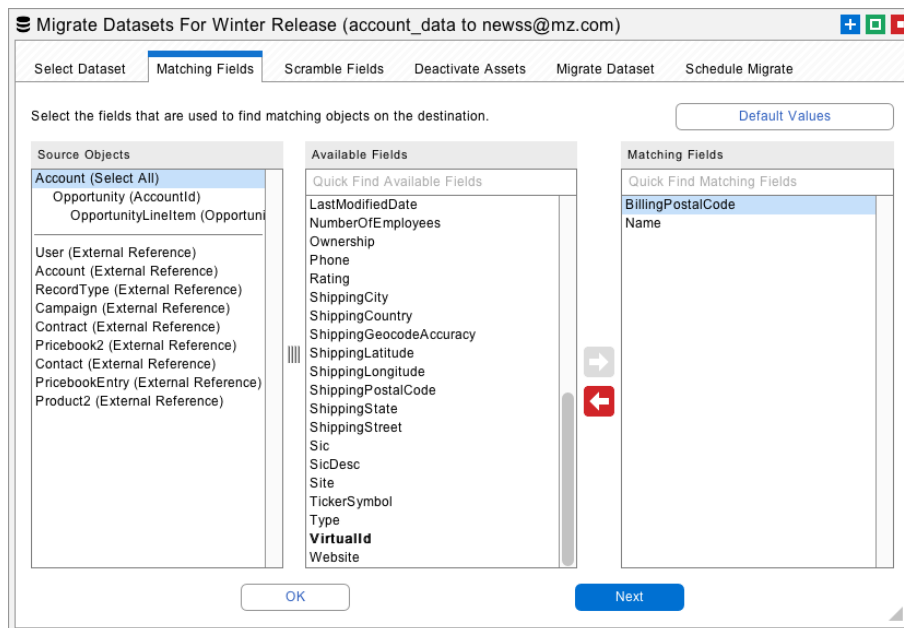
Matching Fields

Snapshot uses the loaded fields to identify matching records in the destination org during migration. If a source record matches a destination record, then the corresponding destination record is updated. If a source record does not match any destination record, then a new destination record is created.

Snapshot automatically selects common matching fields for you. However, you can also manually select matching fields. For example, if you want to match Accounts by Name and BillingCity, then be sure that both Name and BillingCity are loaded. The selected fields create a logical AND filter for matching destination records.

One powerful way to match destination objects is with External Ids. Other common matching fields include object names, email addresses, and usernames. These fields will automatically be available to select for matching. Some Salesforce sandboxes have the same Ids as production orgs. In that case you can simply use the Id field for matching destination objects.

Another way to match destination objects is with Virtual IDs. Use the Virtual Id to match hard to find objects that don't have a name. When these objects are created, Snapshot will remember the source ID and destination ID and always match them during future migrations.



Migrate Datasets For Winter Release (account_data to newss@mz.com)

Select Dataset Matching Fields Scramble Fields Deactivate Assets Migrate Dataset Schedule Migrate

Select the fields that are used to find matching objects on the destination. [Default Values](#)

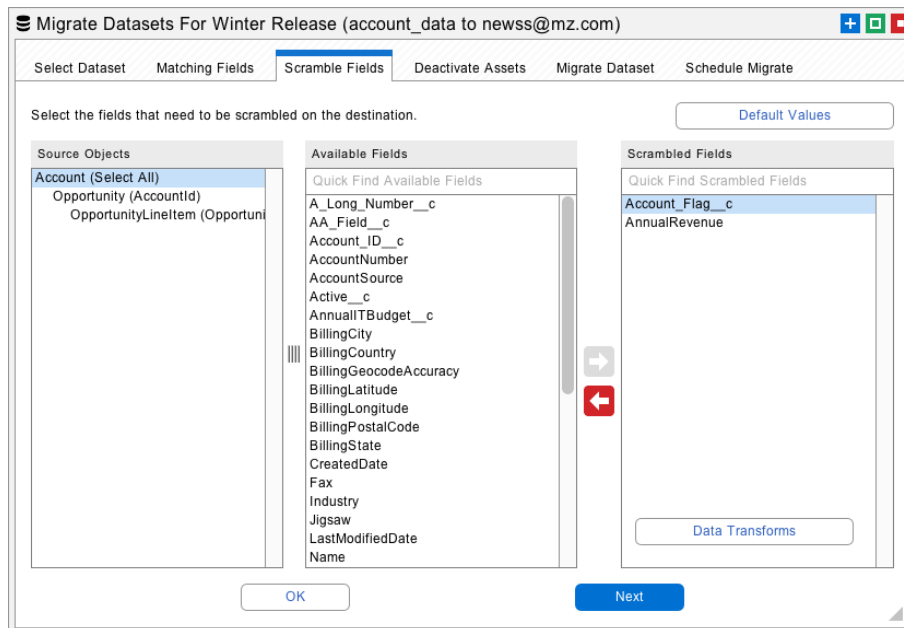
Source Objects	Available Fields	Matching Fields
Account (Select All)	Quick Find Available Fields	Quick Find Matching Fields
Opportunity (AccountId)	LastModifiedDate	BillingPostalCode
OpportunityLineItem (OpportunityLineItemId)	NumberOfEmployees	Name
User (External Reference)	Ownership	
Account (External Reference)	Phone	
RecordType (External Reference)	Rating	
Campaign (External Reference)	ShippingCity	
Contract (External Reference)	ShippingCountry	
Pricebook2 (External Reference)	ShippingGeocodeAccuracy	
Contact (External Reference)	ShippingLatitude	
PricebookEntry (External Reference)	ShippingLongitude	
Product2 (External Reference)	ShippingPostalCode	
	ShippingState	
	ShippingStreet	
	Sic	
	SicDesc	
	Site	
	TickerSymbol	
	Type	
	VirtualId	
	Website	

OK Next

Underneath the dividing line you will see “external references” to other objects in the destination org. Snapshot will automatically connect external references to matching objects on the destination org when the dataset is migrated. You can manually specify additional fields needed for matching external references as well.

Scrambled Fields

Datasets are often used to move records into a Salesforce Sandbox or Developer Edition for testing or application development. In these situations, you may want to scramble data records that contain sensitive information. These fields might contain financial information, such as credit cards or bank accounts, or personal information, such as email addresses or Social Security numbers. The Scramble Fields tab provides an easy way to obscure fields on the destination org. Move the fields that you want to scramble over to the list at right.



Migrate Datasets For Winter Release (account_data to newss@mz.com)

Select Dataset Matching Fields **Scramble Fields** Deactivate Assets Migrate Dataset Schedule Migrate

Select the fields that need to be scrambled on the destination. [Default Values](#)

Source Objects	Available Fields	Scrambled Fields
Account (Select All) Opportunity (AccountId) OpportunityLineItem (OpportunityId)	Quick Find Available Fields A_Long_Number__c AA_Field__c Account_ID__c AccountNumber AccountSource Active__c AnnualTBudget__c BillingCity BillingCountry BillingGeocodeAccuracy BillingLatitude BillingLongitude BillingPostalCode BillingState CreatedDate Fax Industry Jigsaw LastModifiedDate Name	Quick Find Scrambled Fields Account_Flag__c AnnualRevenue Data Transforms

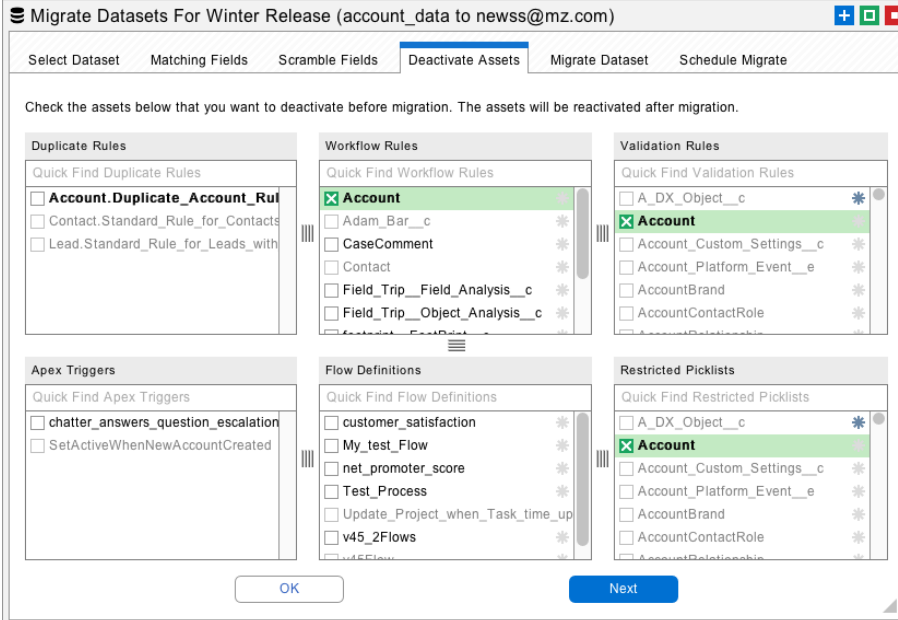
OK Next

Also notice the Data Transforms button at the lower right. Use this feature to transform field values on the fly as they are being written to the destination org. You can transform field values by prefix, suffix, or substring. You can also search and replace by multiple values. This is useful for remapping picklists.

Deactivate Assets

When a Salesforce record is inserted or updated, various Apex Triggers, Workflow Rules, and Validation Rules might be invoked. Apex Triggers perform custom actions before or after records are changed. Workflow Rules can also be invoked when records change, at which point they will perform automated actions. Validation Rules verify that the record data meets some kind of criteria before being inserted, updated, or deleted.

All of these automated behaviors can cause potentially undesirable effects during data migration. For example, thousands of emails might be sent out, or some records might not be updated. The Deactivate Assets tab provides an easy way to deactivate Apex Triggers, Workflow Rules, and Validation Rules in the destination org before data migration is attempted. After migration, the deactivated triggers and rules will be turned back on.



Migrate Datasets For Winter Release (account_data to newss@mz.com)

Select Dataset Matching Fields Scramble Fields **Deactivate Assets** Migrate Dataset Schedule Migrate

Check the assets below that you want to deactivate before migration. The assets will be reactivated after migration.

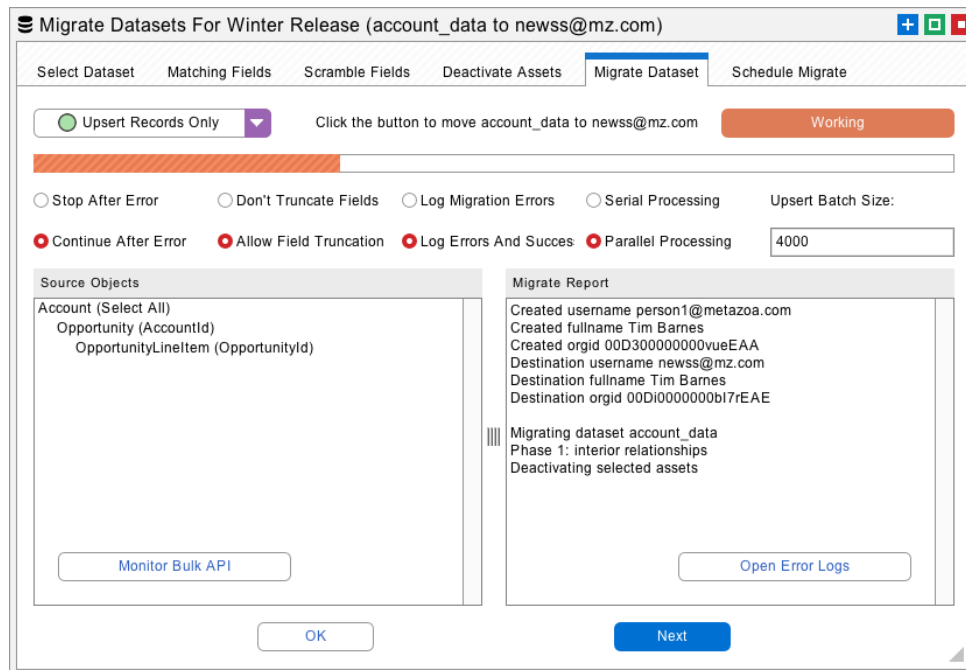
Duplicate Rules	Workflow Rules	Validation Rules
Quick Find Duplicate Rules	Quick Find Workflow Rules	Quick Find Validation Rules
<input type="checkbox"/> Account.Duplicate_Account_Rul	<input checked="" type="checkbox"/> Account	<input type="checkbox"/> A_DX_Object__c
<input type="checkbox"/> Contact.Standard_Rule_for_Contacts	<input type="checkbox"/> Adam_Bar__c	<input type="checkbox"/> Account_Custom_Settings__c
<input type="checkbox"/> Lead.Standard_Rule_for_Leads_with	<input type="checkbox"/> CaseComment	<input type="checkbox"/> Account_Platform_Event__e
	<input type="checkbox"/> Contact	<input type="checkbox"/> AccountBrand
	<input type="checkbox"/> Field_Trip__Field_Analysis__c	<input type="checkbox"/> AccountContactRole
	<input type="checkbox"/> Field_Trip__Object_Analysis__c	<input type="checkbox"/> AccountRelationship

Apex Triggers	Flow Definitions	Restricted Picklists
Quick Find Apex Triggers	Quick Find Flow Definitions	Quick Find Restricted Picklists
<input type="checkbox"/> chatter_answers_question_escalation	<input type="checkbox"/> customer_satisfaction	<input type="checkbox"/> A_DX_Object__c
<input type="checkbox"/> SetActiveWhenNewAccountCreated	<input type="checkbox"/> My_test_Flow	<input checked="" type="checkbox"/> Account
	<input type="checkbox"/> net_promoter_score	<input type="checkbox"/> Account_Custom_Settings__c
	<input type="checkbox"/> Test_Process	<input type="checkbox"/> Account_Platform_Event__e
	<input type="checkbox"/> Update_Project_when_Task_time_up	<input type="checkbox"/> AccountBrand
	<input type="checkbox"/> v45_2Flows	<input type="checkbox"/> AccountContactRole
		<input type="checkbox"/> AccountRelationship

OK Next

Migrate Datasets Button

The next tab has the main interface for migrating datasets to the destination org. First, make sure that the migration options are set correctly, these are discussed in more detail below. Then click the Migrate Datasets button to get started. All of the details of the migration will be written to the report at lower right. The last tab allows you to Migrate Datasets at a specific time in the future or as a recurring event.



The screenshot shows a web application window titled "Migrate Datasets For Winter Release (account_data to newss@mz.com)". The window has several tabs: "Select Dataset", "Matching Fields", "Scramble Fields", "Deactivate Assets", "Migrate Dataset" (which is active), and "Schedule Migrate".

Below the tabs, there is a dropdown menu set to "Upsert Records Only" and a button labeled "Click the button to move account_data to newss@mz.com". To the right of this button is a status indicator that says "Working" with a red progress bar.

Below the progress bar, there are several radio button options for migration settings:

- ☐ Stop After Error
- ☐ Don't Truncate Fields
- ☐ Log Migration Errors
- ☐ Serial Processing
- ☒ Continue After Error
- ☒ Allow Field Truncation
- ☒ Log Errors And Successes
- ☒ Parallel Processing

There is also a field for "Upsert Batch Size:" with the value "4000".

The window is divided into two main sections:

- Source Objects:** A list containing "Account (Select All)", "Opportunity (AccountId)", and "OpportunityLineItem (OpportunityId)". Below this list is a button labeled "Monitor Bulk API".
- Migrate Report:** A text area containing the following information:
 - Created username person1@metazoa.com
 - Created fullname Tim Barnes
 - Created orgid 00D300000000vueEAA
 - Destination username newss@mz.com
 - Destination fullname Tim Barnes
 - Destination orgid 00Di00000000bi7rEAEBelow this, it says "Migrating dataset account_data", "Phase 1: interior relationships", and "Deactivating selected assets". At the bottom of this section is a button labeled "Open Error Logs".

At the bottom of the window, there are two buttons: "OK" and "Next".

Migration Options

There are various options for Data Migration:

- Upsert Records Only
- Delete Then Upsert
- Delete Records Only

The menu in the upper left gives you the option to delete the current dataset before migration, delete and then upsert, and upsert only. For production orgs we check twice that the administrator wants to delete the data. Be careful with this option.

- Stop After Error
- Continue After Error

If an error occurs, then Snapshot will either stop processing additional files with the Bulk API or continue. All errors are written to the log files. Common errors include too many duplicate records, email addresses in the wrong format, field data in the wrong format, etc.

- Don't Truncate Fields
- Allow Field Truncation

One common problem when moving data between different types of orgs is text strings that are too long for the destination field. Select this option to automatically truncate fields to the correct length or otherwise report an error.

- Log Migration Errors
- Log Errors and Success

All errors are written to the Log Files located in the “datasets” folder in the Snapshot file system. Optionally you can also log successful migrations. The log file contains the source and destination Ids as well as any error message.

- Serial Processing
- Parallel Processing

Parallel processing is faster but may cause database lock errors. Switch to serial processing if there are lots of lock errors reported.

The Upsert Batch Size field is used for specifying smaller batch sizes. There is a limit of 12 duplicate records per batch, so a smaller batch size may be helpful in avoiding this error.

Special Cases

Our Monarch product has been rewritten quite a few times over the last 15 years. Along the way we have added some special cases to handle difficult migration issues. Ironically, Custom Objects work great, and we handle every use case. The problems start when migrating Standard Objects that were developed back in the olden days. They often have special server-side rules and constraints that require special handling.

Content Documents

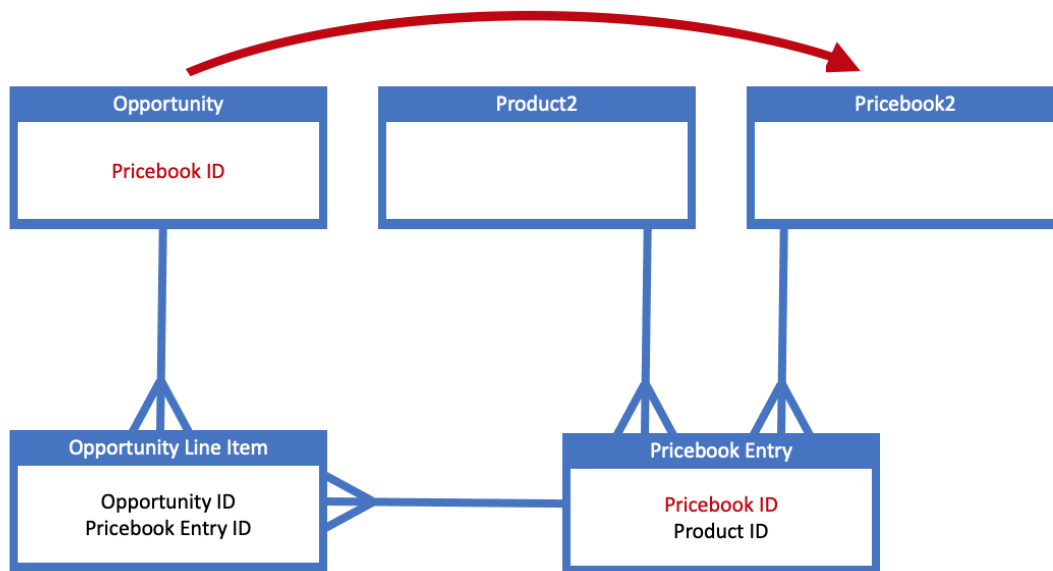
Salesforce Content (also called Salesforce Files) is the new way to store documents and attachments in an org. Normally a Content Document will be the parent for any number of Content Version child records. The Content Version record holds the actual data. There are lots of limitations working with Content Documents, but Snapshot takes care of most of these problems. Simply migrate Content Documents with Content Versions as children. But there are two issues to be aware of.

First, only the latest Content Version can be updated. Be sure that your source Dataset only includes the latest Content Version for each Content Document. You can do this by setting a child filter on the Content Version object. The filter should be “IsLatest=true” and this can be set with the Child Record Limits option.

Secondly, Content Versions that are marked as a major version cannot be updated. This is controlled by the “IsMajorVersion” field and normally you will want that value to be false. This can be enforced by remapping the field value in the Manage Datasets interface or on the fly with the Scramble Fields and Transforms tab in Migrate Datasets.

Opportunity Pricebooks

Every Opportunity has some Opportunity Line Items, and each line item is associated with a Pricebook Entry. The Pricebook Entry is automatically generated to match each Product and Pricebook pair. However, the Opportunity also has a direct reference to a Pricebook. That Pricebook must be the same one that is indirectly referenced by the Opportunity Line Item via the Pricebook Entry.



You can only create Opportunity Line Items that reference the same Pricebook that the Opportunity uses. This is tricky because the reference is indirect on both sides, so the correct Pricebook Entry must be selected. Monarch will do this automatically for you. Simply migrate these items in the hierarchy below:

```
Product2 (Select All)
Pricebook2 (Select All)
    PricebookEntry (Pricebook2Id)
```

Group Members

Here is another special case. Group Members are the junction object that connects a Group to another User or Group. They are not updateable, so Monarch will delete them and then create them as needed during migration. Be sure that the parent Group is migrated at the same time, the hierarchy should look like this:

```
Group (Select All)
  GroupMember (GroupId)
```

Custom Settings

There are two types of Custom Settings: List and Hierarchical. List custom settings must have a unique name, and the Setup Owner Id field must be equal to the Organization Id. Hierarchical settings must have unique Setup Owner Id that is equal to the Organization ID, a User ID, or a Profile ID. All of these constraints make Custom Settings hard to migrate. Monarch will handle the migration automatically, although there are some possible errors. For example, you cannot migrate a list Custom Setting to a hierarchical one, and vice versa.

Polymorphic Relationships

Some lookup relationship fields are polymorphic. This means that a single relationship field can reference many different types of objects. For example, the Event object has a polymorphic relationship field called WhatID. This field can refer to dozens of different object types including Accounts, Contracts, and Opportunities.

Since many different types of objects can be referenced, there is no easy way to specify complex matching criteria for polymorphic relationship fields. By default, polymorphic relationship fields only reference the destination object by name. But some objects (like Accounts) can have duplicate names. Monarch handles this situation by first matching by name and then looking for External ID fields that match the source object ID.

Bulk API Limitations

The Bulk API won't move everything. A related problem is that some objects have Base64 fields that can be gigantic. For these special object Monarch switches to the SOAP Data API and handles the migration in a serial mode. This ensures that every type of object can be migrated. Here are the objects with Base64 fields:

- Attachment
- ContentNote
- ContentVersion
- Document
- EmailCapture
- EventLogFile
- MailmergeTemplate
- MobileApplicationDetail
- QuoteDocument
- Scontrol
- StaticResource

And here are the objects the Bulk API does not support:

- AcceptedEventRelation
- CaseStatus
- ContractStatus
- KnowledgeArticle
- KnowledgeArticleVersion
- KnowledgeArticleVersionHistory
- KnowledgeArticleViewStat
- KnowledgeArticleVoteStat
- LeadStatus
- OpportunityStage
- PartnerRole
- RecentlyViewed
- SolutionStatus
- TaskPriority
- UserRecordAccess
- ContentFolderItem
- DeclinedEventRelation
- EventWhoRelation
- TaskStatus
- TaskWhoRelation
- UndecidedEventRelation
- AccountUserTerritory2View
- DevelopmentPackageVersion
- InstalledPackageVersion

Conclusion

This whitepaper has discussed the best practices for data migration using Snapshot on the Salesforce platform. The Snapshot product from Metazoa provides a best-of-breed solution for continuous integration with a highly flexible toolset.

support@metazoa.com

1-833-METAZOA

1-833-638-2962

<https://www.metazoa.com>

Twitter: @metazoa4sf

Facebook: <https://www.facebook.com/metazoa4sf>

LinkedIn: <https://www.linkedin.com/company/18493594/>