Migrate data from

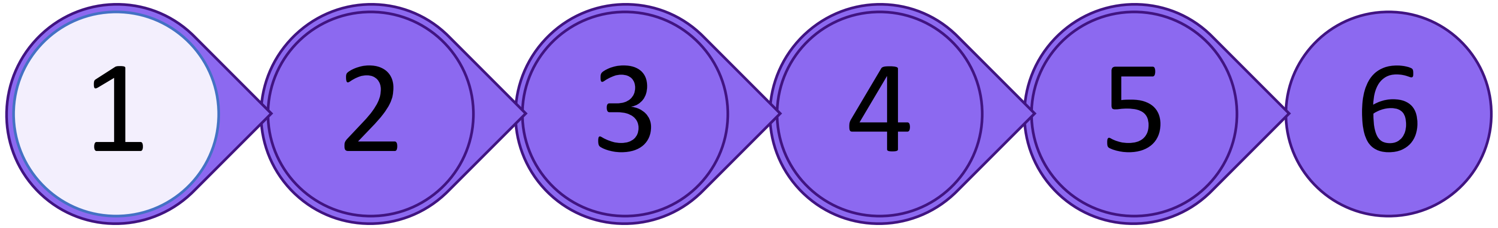
SharePoint list to Dataverse

November, 2022

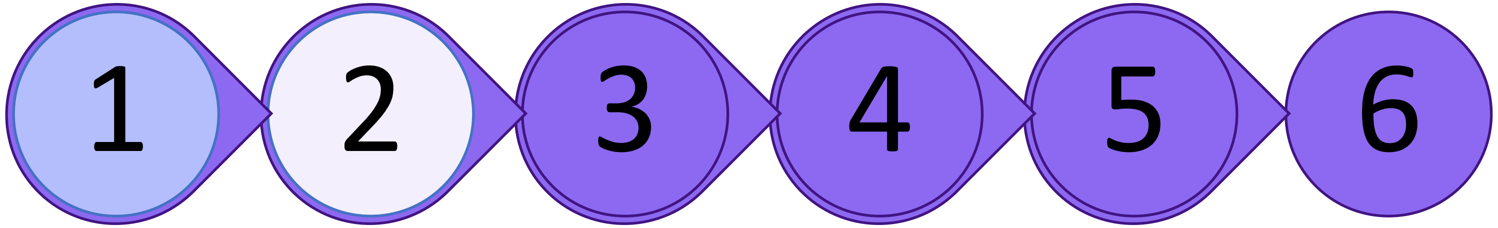
Steps

1. Analyze the SharePoint list columns
2. Create a Dataflow and connect to SharePoint site
3. Apply the data transformation
4. Create target Dataverse table with complex columns
5. Map the columns and publish the Dataflows
6. Develop a Power Automate Flows to update complex columns in each row

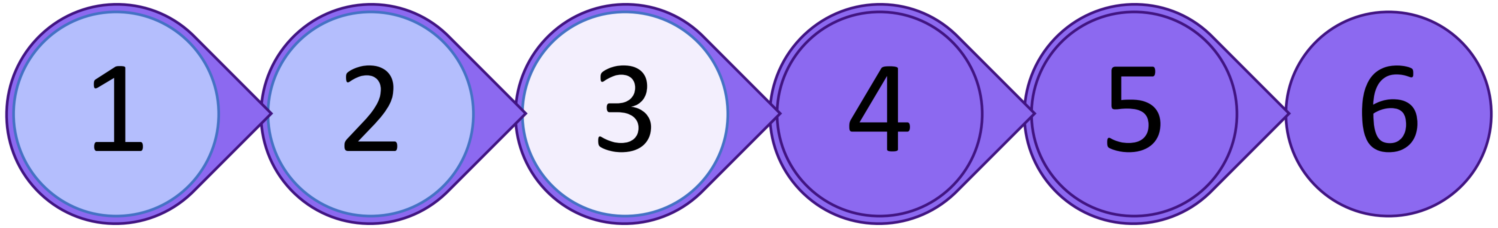
1. **Analyze the SharePoint list columns**
2. Create a Dataflow and connect to SharePoint site
3. Apply the data transformation
4. Create target Dataverse table with complex columns
5. Map the columns and publish the Dataflows
6. Develop a Power Automate Flows to update complex columns in each row



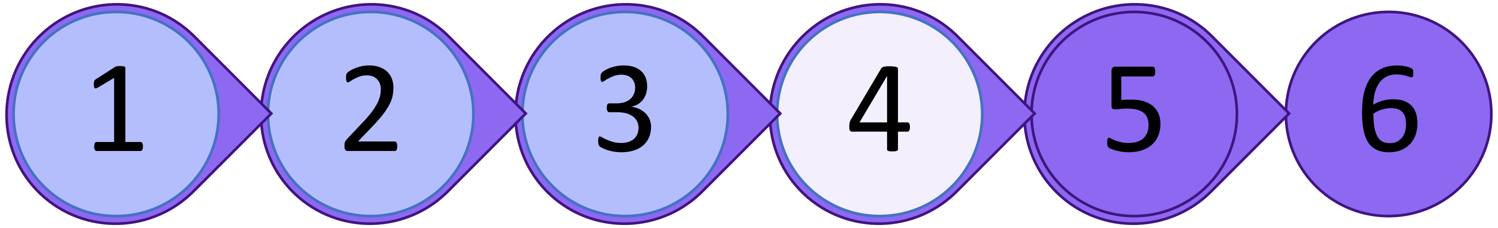
1. Analyze the SharePoint list columns
2. **Create a Dataflow and connect to SharePoint site**
3. Apply the data transformation
4. Create target Dataverse table with complex columns
5. Map the columns and publish the Dataflows
6. Develop a Power Automate Flows to update complex columns in each row



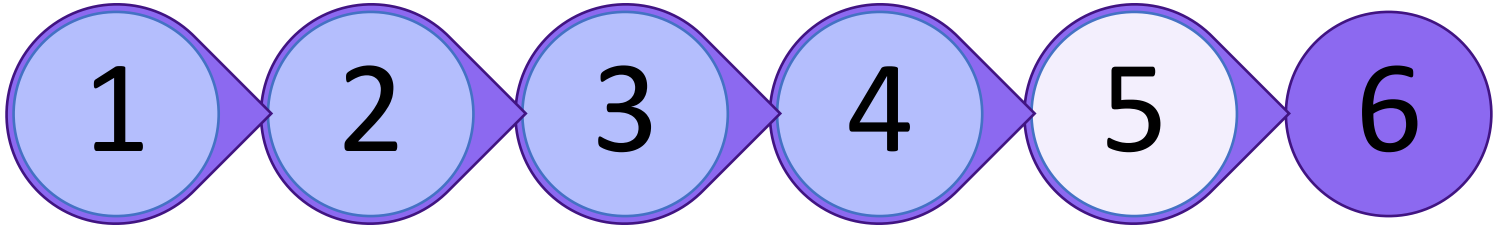
1. Analyze the SharePoint list columns
2. Create a Dataflow and connect to SharePoint site
3. **Apply the data transformation**
4. Create target Dataverse table with complex columns
5. Map the columns and publish the Dataflows
6. Develop a Power Automate Flows to update complex columns in each row



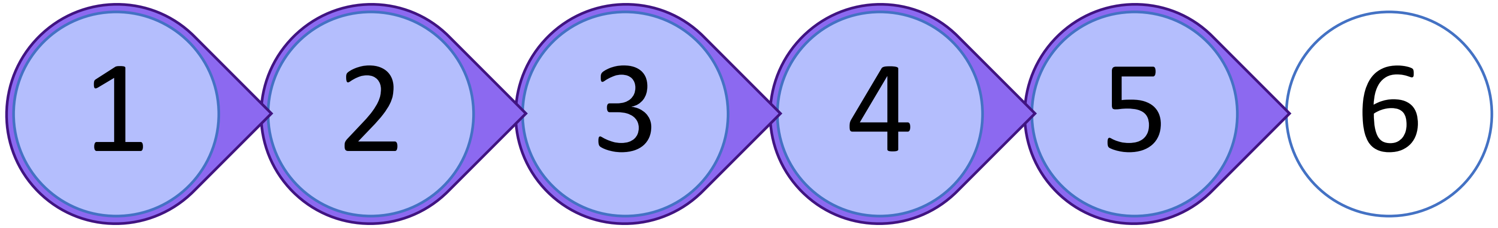
1. Analyze the SharePoint list columns
2. Create a Dataflow and connect to SharePoint site
3. Apply the data transformation
4. **Create target Dataverse table with complex columns**
5. Map the columns and publish the Dataflows
6. Develop a Power Automate Flows to update complex columns in each row



1. Analyze the SharePoint list columns
2. Create a Dataflow and connect to SharePoint site
3. Apply the data transformation
4. Create target Dataverse table with complex columns
5. **Map the columns and publish the Dataflows**
6. Develop a Power Automate Flows to update complex columns in each row



1. Analyze the SharePoint list columns
2. Create a Dataflow and connect to SharePoint site
3. Apply the data transformation
4. Create target Dataverse table with complex columns
5. Map the columns and publish the Dataflows
6. **Develop a Power Automate Flows to update complex columns in each row**



Dataflows and Power Automate

Context and Tech stack

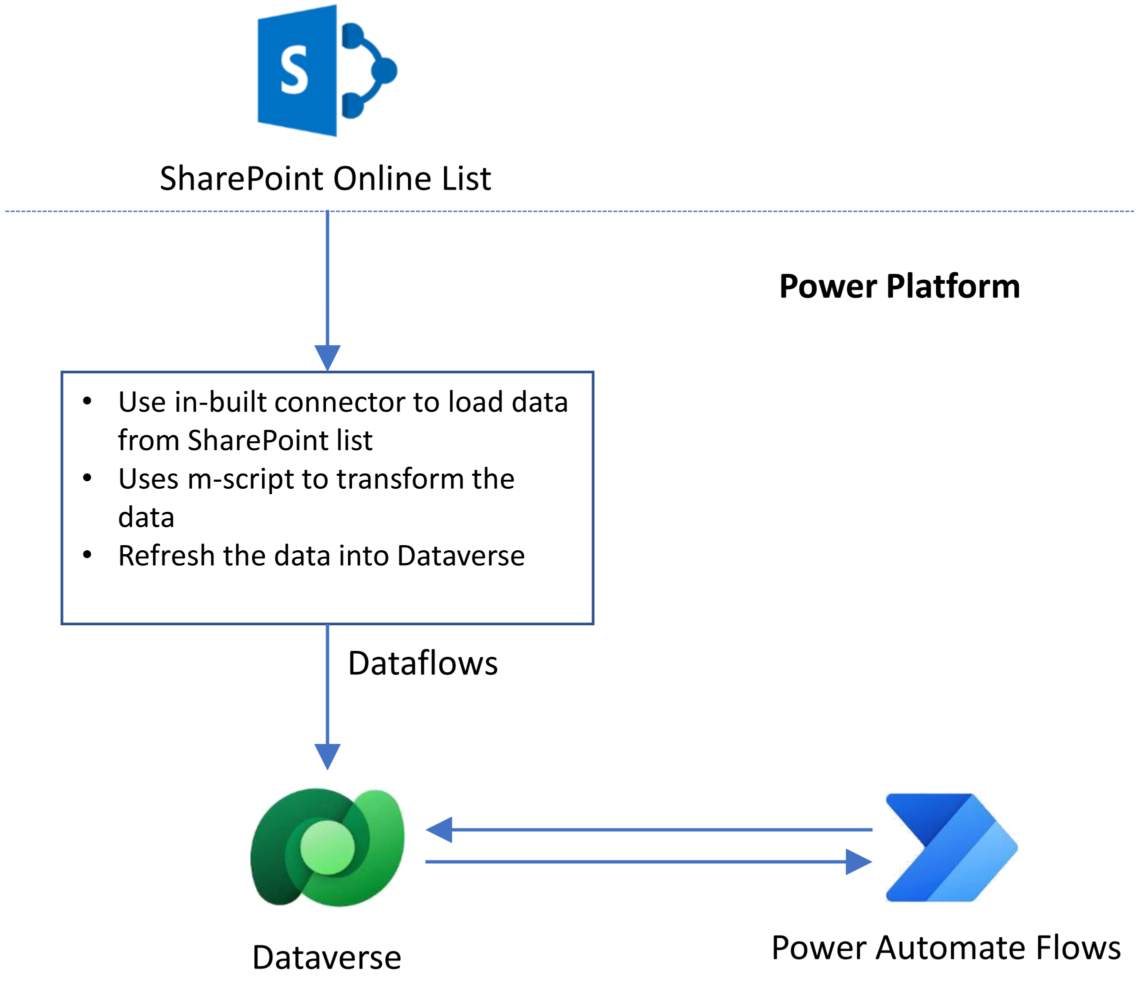
Application/Business Context

To migrate data from SharePoint list Dataverse table, no third-party tools are available as on today

- ❖ Dataflows in Power Platform can be used to migrate SharePoint list data into Dataverse table
- ❖ Dataflows uses the same Power Query tool to load and transform data into desirable structure
- ❖ Use Power Automate Flows to update complex columns of each rows

Technology Stack

- SharePoint
- Dataflows
- Dataverse
- Power Automate Flows



SharePoint Online List

Power Platform

- Use in-built connector to load data from SharePoint list
- Uses m-script to transform the data
- Refresh the data into Dataverse

Dataflows

Dataverse

Power Automate Flows

Features of Dataflows

- It is declarative way to load and transform data from multiple data sources
- Use rich m-script to perform complex data transformation
- Load data into Dataverse table once or in a predefined schedule
- Allow delta migration based on predefined identity column
- Dataflows can migrate the data into an existing Dataverse table or can also migrate to a new table to be created as defined in Dataflows

SharePoint list data migration

1) Dataflows can be used to migrate data from lists in both on-prem and SharePoint online site.

2) For SharePoint online, no additional setup or license is required.

3) For SharePoint on-prem the On-prem data gateway needs to be configured to access SharePoint on-prem sites in Dataflows.

4) The primitive column types are fully supported in Dataverse table and can be migrated without any additional Power Query steps

5) The complex SharePoint list columns cannot be migrated as it was in SharePoint

SharePoint Columns	Dataverse Table Column	Comments
Text	**Text**	
Multiline Text	**Multiline Text**	
Multiline Text with RTF	**Multiline Text**	**Modify the Form to add RTF field component for this multiline text field**
DateTime	**DateTime**	
Number	**Number**	

Choice Field (Single Select)

Dataverse supports choice fields. But choice options in Dataverse must have internal value field which is not the case for choice options in SharePoint. In SharePoint choice options are a list of text only. Exclude this column if you are migrating to a new table. If you are migrating data into an existing table follow the below steps:

- Define a Choice column in Dataverse table
- Create the choice options with numeric values in 1,2,3.. Format
- Edit the dataflow with advanced editor
- Add m-script to replace the Choice option with corresponding 'Value' of choice option in Dataverse:

```
          getPriorityChoice=(priority as text) as text=>
            if priority = "Low" then "1"
            else if priority = "Normal" then "2"
            else if priority = "High" then "3"
            else "",
  #"with priority choice" = Table.AddColumn(Navigation,"PriorityChoice", each getPriorityChoice([Priority])),
```

- The above m-script creates a new columns 'PriorityChoice' for 'Priority' choice column
- The Priority column text are replaced by choice options in Dataverse
- In the data load screen map the PriorityChoice column to the destination column Dataverse

SharePoint list Complex column migration

SharePoint complex columns are supported in Dataverse but there are architectural changes.

SharePoint Column	Dataverse Column	Remarks
Multi select Choice	**Multi select choice**	Add transformation in dataflow to create comma separated text of Dataverse choice values
Single valued lookup	**Text**	Configure dataflow to migrate the value as text field, add a lookup column and develop a manually triggered flow to set the lookup values
Multi valued lookup	**Text**	Configure dataflow to migrate the lookup values as comma separated text, add the many to many relationship with the look up table, develop a Power Automate flow to add the establish the relationship for all rows.
Person or Group	**Text**	In dataflow, transform the column to emit comma separated email id of all users. In the Dataverse table create many to relationship with the Users table. Develop Power Automate flow to relate rows for each email id. In the Power Automate Table form add sub grid to show related users.

HCLTech | Supercharging Progress™

hcltech.com