**Configure Microsoft Dataverse (25–30%)**

**Manage the data model**

* Create or modify standard, activity, or virtual tables
* Create new tables or modify existing tables
* Determine which type of relationship to implement, including one-to-many and many-to-many
* Configure table relationships including behaviours and cascading rules
* Create new or modify existing columns, including calculated, rollup, and formula
* Configure table properties

**Manage Dataverse**

* Configure Dataverse search, and manage the search index
* Manage auditing
* Describe options for importing and exporting data
* Configure duplicate detection settings
* Configure bulk deletion

**Configure security settings**

* Manage business units
* Create and manage security roles
* Create and manage users and teams
* Create and manage column security
* Configure hierarchy security
* Configure Microsoft Entra ID group teams

Getting Started with Dataverse

Each ***environment*** is created under a Microsoft Entra ID tenant, and only users within that tenant can access its resources. An environment (Dataverse DB and resources created in that env) is also bound to a geographic location, like the United States. Can create more than 1 env (e.g., dev and prod).

Dev env - A Power Platform **admin** can create Developer environments for other users and can restrict users from creating their own Developer environments.

Creating env - once you create an environment with Dataverse in it, you can access that Dataverse data from other environments as well. Dataverse creation with an environment reserves up to 1 GB in storage.

Dataverse

Dev env

Dev env

Dataverse

Dataverse

Prod env

Prod env

Users & Roles - After you have created an environment, users from Microsoft Entra ID that are associated with your tenant and have a selected security role are automatically added to the environment.

\*We recommend that you create Microsoft Entra ID groups and associate roles with those security groups to simplify permissions and data access.

Notes: The two Environment **roles** that are built into **every** environment are System Administrator and Environment Maker. All other roles are user security roles.

The following **operation** can be done on the **Environment**:

Change env type – prod to sandbox, sandbox to prod  
Delete env - to recover storage space and to remove personal data  
Recover env - within seven days of deletion  
Reset env – useful when we need to create new project or free up resources and personal data  
Copy env – there are two level of copy 1) Everything, and 2) Customizations and schemas only  
Restriction apply such as:  
- can only copy to an environment in the same tenant and region  
- can’t copy to prod env  
- copy and restore operations can take up to 8 hours – 24 hours

Backup and Restore

System backup – with Dynamic enabled, it stored 28 days, if not, only 7 days

Manual backup – sandbox/dev for 7 days, prod with Dynamic is 28 days

Restoring env - To restore to a production environment, you must first change its type to sandbox.

Administration mode - is useful when you want to make operational changes and not have regular users affect your work, and not have your work affect end users (non-admins).

Table

To create a table, go to <https://make.powerapps.com/>, you must be in correct environment.

Click “Table” menu item from the left menu bar. Select Create. Below are three properties:

* 1. Schema name: This is the internal name of the table created in the database
  2. Type: Standard, Activity, Virtual or Elastic.
  3. Record ownership: User or Team or Organizational

You can’t change these options once a table is created.

Table types:

1. **Standard**: A table where you can store data and add to the navigation in model-driven apps.
2. **Activity**: Store interactions with time dimensions such as phone calls, tasks, and appointments. Dataverse has a set of activity tables. These tables share the same set of columns and share security privileges. Many of the table options, including the primary column, are fixed and can't be changed. Activity tables appear in the timeline on model-driven app forms.
3. **Virtual**: Connect to data from an external data source such as Microsoft Azure SQL Database. Virtual tables appear in your app to users as regular table rows but contain data that is sourced from an external database dynamically at runtime.
4. **Elastic**: Used when your table includes a large dataset in excess of tens of millions of rows. Elastic tables are powered by Azure Cosmos DB.

Licensing Requirements for each table type

A screenshot of a computer program

Description automatically generated

Tables with complex business logic (**or Complex tables**) such as code plug-ins or real time workflows will require that users have a Power Apps or Power Automate per user license.

**Restricted tables** aren't standard tables within Dataverse but are included in one of the customer engagement apps in Dynamics 365 or a third-party solution. They'll require a Dynamics 365 license.

More details on Licensing Requirements for each table type are [here](https://learn.microsoft.com/en-us/power-apps/maker/data-platform/data-platform-entity-licenses).

More details on Restricted tables requiring Dynamic 365 license are [here](https://learn.microsoft.com/en-us/power-apps/maker/data-platform/data-platform-restricted-entities).

Table Ownership

1. User or team owned: Data belongs to a user or team. Actions that can be performed on these rows can be controlled at the user level.
2. Organization-owned: Data belongs to the organization. Access to the data is controlled at the organization level.

Tips: Dataverse came with many standard tables and recommended using them (by changing the column name if needed) instead of creating a new table. You can also hide the standard tables if not using them. To hide a standard table, change the security role privileges for your organization to remove the Read privilege for that table. This removes the table from most parts of the application.

A screenshot of a computer

Description automatically generated

We can customize Views and Forms for a table.

Create a Dataverse Table:

<https://learn.microsoft.com/en-us/training/modules/get-started-with-powerapps-common-data-service/3-create-a-cds-entity>

Import SharePoint list data into Dataverse Table:

<https://learn.microsoft.com/en-us/training/modules/get-started-with-powerapps-common-data-service/4-import-data-into-your-database>

Create table relationships:

<https://learn.microsoft.com/en-us/training/modules/get-started-with-powerapps-common-data-service/create-table-relationships>

Create custom table, business rule and import Excel spreadsheet data to Dataverse table

<https://learn.microsoft.com/en-us/training/modules/get-started-with-powerapps-common-data-service/4a-use-data-cds-exercise>

Dataverse Logic and Security

Reduce repetitive code in app - Tables within Dataverse can use rich server-side logic and validation to ensure data quality.

* Business rules: Business rules validate data across multiple columns in a table, and provide warning and error messages, regardless of the app that's used to create the data.
* Business process flows: Business process flows guide users to ensure they enter data consistently and follow the same steps every time. Business process flows are currently supported only for model-driven apps.
* Real-time workflows: Workflows automate business processes without requiring user interaction.
* Business logic with code: Business logic supports advanced developer scenarios that extend the application directly through code.

Business Rules

Business rules can be used for data validation, enforce requirement, set value, perform calculation, etc. Business rules depend on the scope of the rule which can be:

1. **Individual form**: The rule applies only to the specified model-driven app form.
2. **All forms**: The rule applies to all model-driven app forms.
3. **Entity**: The rule applies to all model-driven app forms and when the row is created or updated on the Dataverse table. This is the default setting.

Two main components here are Flow and Actions. Flow is the condition check and action is the tasks to perform such as showing error messages, setting field visibility or setting/clearing default value, etc.

Note: Business rules defined for a table apply to both canvas apps and model-driven apps if the table is used in the app BUT in canvas apps, show or hide, enable or disable and creating business recommendations based on business intelligence are not supported.

Dataverse Security

Standard security roles System Administrator, Environment Maker and Basic User. But should create custom security roles because standard security roles cannot be customized for your tables. You can only modify custom security roles.

You can manage security roles from the Power Platform admin center. Select your environment and select Settings, expand Users + permissions and select Security roles.

Dataverse Auditing

Three levels of auditing: Environment, Table and Column.

Dataverse auditing is supported on all custom and most customizable tables and columns. Audit logs are stored in Dataverse and **consume log storage capacity**. You can view audit logs in the Audit History tab for a single record. You can view audit logs in the Audit Summary view for all audited operations in a single environment. Audit logs can also be retrieved using the Web API or the SDK for .NET.

It's possible to refer to audit log data in a model driven app under **Audit history**.

Note that auditing won’t run if is turned off for your environment.

Dua-write vs Virtual tables/entities

Dual-write – is an out-of-box infrastructure (automated data flow) that provides near-real-time interaction between customer engagement apps and finance and operations apps. Dual-write also supports online and offline mode, and it follows the no-code/low-code principle.

Virtual Entities – tables dose not reside in Dataverse. Instead, it continues to reside in the app where it belongs but support CRUD operations. The entities must be made available as virtual entities in Dataverse to support CRUD.

Together, virtual entities and dual write are part of the shared data layer for the convergence of finance and operations apps and the Dataverse platform. They're complementary technologies that are intended to work **together**.

When you must have access to your data offline, consider Dual write. If data is for read-only then consider virtual entities which also save the storage space in Dataverse.

**Primary Column** – a way to identify a record in a table and is not a unique identifier (NOT a Primary Key). The Unique identifier column is a GUID (autogenerated). To make the primary column unique, then create an alternate key (with unique value) and assign the Primary Name column to the new key. You can create up to five keys for each table. All values in the key are unique.

**Choice Column** – Two options: Local Choice vs Global Choice.

Local choices can only be used by the table and column that they're created against and can't be reused on other tables. A global choice is a separate component and can be reused for multiple columns on multiple tables. The list of choices is shared for the table columns that reference the global choice.

**Dataverse Data Export Options** – Beside exporting Excel format, offer both Power Automate and Azure Synapse Link.

Power Automate – use cloud flows with the Dataverse connector to query table rows and generate output files. Power Automate can extract exactly the information that you require in the format that you need.

Azure Synapse Link – after enabling the Azure Synapse Link for Dataverse in the Power Apps maker portal, you choose the tables in Dataverse and Dataverse will perform a continuous replication of data to Azure to either an Azure Data Lake storage Gen2 account or an Azure Synapse Analytics workspace.

Using Dataflow to import data into Dataverse

Create dataflow with destination storage and schedule data refresh. This can be done in Power App Portal under “Dataflows” (may need to click on …More menu option).

Importing data using dataflow link is [here](https://learn.microsoft.com/en-us/training/modules/load-export-data-create-data-views/5a-import).

**Action-based Data Source (Service Name or Action Connectors)**

Differ from the more popular tabular data sources. You use functions to interact with the data source instead of just reading and writing data. But it cannot be used with **Forms Control**. Most action-based data sources provide functions for updating the data as appropriate. An action-based data source can also be for things like sending emails or other notifications, not necessarily only for reading and writing data. Power App supports both tables and action-based data sources.

More on how to work with action-based data source is [here](https://learn.microsoft.com/en-us/training/modules/connect-to-other-data-in-powerapps-canvas-app/2-work-with-action-based-data-sources).

Power Automate is a companion of Power App

Power Automate helps you to pull data from many different tables and action connectors, so it's a valuable companion for Power Apps. Power Automate is a connector to Power App.

Use case: Send email when someone submit form – this can be done using Power App, but

Use case: Start approval process after sending email – this is where Power Automate comes in

Explanation: You can have Power Apps trigger a Power Automate flow when the user submits the data. Power Automate can then look up who the user's manager is and send the manager an approval request. Power Automate will then help get a response from the manager, update the data source with the status based on their response, and send the original submitter an update.

User case: Consuming third-party API that return complex JSON data and need to purify before storing it to our table. Power Automate can do that and store data to table and then Power App will take care of displaying or working with those data further.

Power App  
is a Trigger Point

Trigger (with params)

Power Automate  
take those params and perform action on the flow

More use cases can be found [here](https://learn.microsoft.com/en-us/training/modules/connect-to-other-data-in-powerapps-canvas-app/3-flow-and-powerapps).