



Administering a PowerApps Enterprise Deployment

Whitepaper

Summary: This is a technical whitepaper outlining considerations for planning, deploying and managing an enterprise PowerApps deployment.

Writers: David Yack (Colorado Technology Consultants)

Technical Contributors: Julie Yack (Colorado Technology Consultants), George Doubinski (solutions.NET), John Landgrave, Manas Maheshwari, Jennifer Monroe, James Oleinik, Saurabh Pant, Kent Weare, Imad Yanni

Published: July 2018

Contents

- Introduction.....4

- PowerApps Platform Overview5
 - Usage Scenarios7
- Platform Architecture8
 - Environments8
 - Common Data Service for Apps..... 13
 - Types of PowerApps..... 16
 - Microsoft Flow 19
 - Connectors..... 21
 - On-premises Gateways..... 26
 - Solution Packages 27
- Licensing and License Management..... 29
 - Organization level accumulated entitlements..... 31
 - Use of connectors..... 31
 - PowerApps Community Plan 31
 - FAQs and more information..... 33
- Security 35
 - Controlling access to PowerApps..... 36
 - Common Data Service 36
- Data Loss Prevention Policies..... 40
- Management and Monitoring..... 45
 - Working with the Admin Portals 46
 - Automation of tasks with PowerShell..... 49
 - Automation of tasks with Microsoft Flow 51
- Deployment Scenarios 52
 - Canvas app or Flows that are built to share with others..... 52
 - Canvas app or Flows with connectors violating existing DLP..... 53
 - Canvas app or Flows with existing CDS database..... 53
 - Canvas or model-driven apps and/or Flow with CDS – Multiple Teams..... 53
- Application Lifecycle Management..... 54
 - Getting ready for a new application..... 55

Tools to help Manage, Plan, Track, and Deploy.....	55
Exporting from the source environment	56
Ongoing application maintenance	57
Moving reference data to another environment.....	59
Using the Dynamics 365 Package Deployer.....	60
Compliance and Data Privacy.....	60
Trust Center	60
Data Location.....	60
Data Protection	61
Resources to manage GDPR Compliance	61
Office 365 Security and Compliance Center	62
Support and Learning Resources.....	64
Next Steps.....	66

Introduction

PowerApps is a high-productivity application development platform from Microsoft. The platform is used by Microsoft to build their own 1st party applications Dynamics 365 for Sales, Service, Field Service, Marketing and Talent. This means these applications are built natively on the platform. Enterprise customers can also build their own custom line of business applications using the same technology. Individual users and teams within your organization can also build personal or team productivity applications with no-code or low-code.

Purpose of this whitepaper

This whitepaper is targeted toward the enterprise application administrator responsible for planning, securing, deploying, and supporting applications built on the PowerApps platform. The goal of the paper is to help you understand what currently is in your environment, how to proactively plan for applications being developed and deployed and finally how to handle day to day administrative tasks to manage deployments.

In this whitepaper, we will cover key concepts, platform architecture, and decisions that will be necessary. Where possible we will help you develop best practices for your organization to ensure successful deployments and high productivity for users using the platform.

Scope of this Whitepaper

Unless specifically noted, all features mentioned in this whitepaper are available as of July 2018.

The following topics are out of scope for this whitepaper:

- Power BI and other parts of the broader Microsoft Power platform
- PowerApps fundamentals for building applications
- ISV deployment scenarios, which are handled differently from enterprise deployment scenarios
- Performance tuning of applications
- Full deployment and management of first party Dynamics 365 applications
- Third party solutions which integrate with PowerApps.

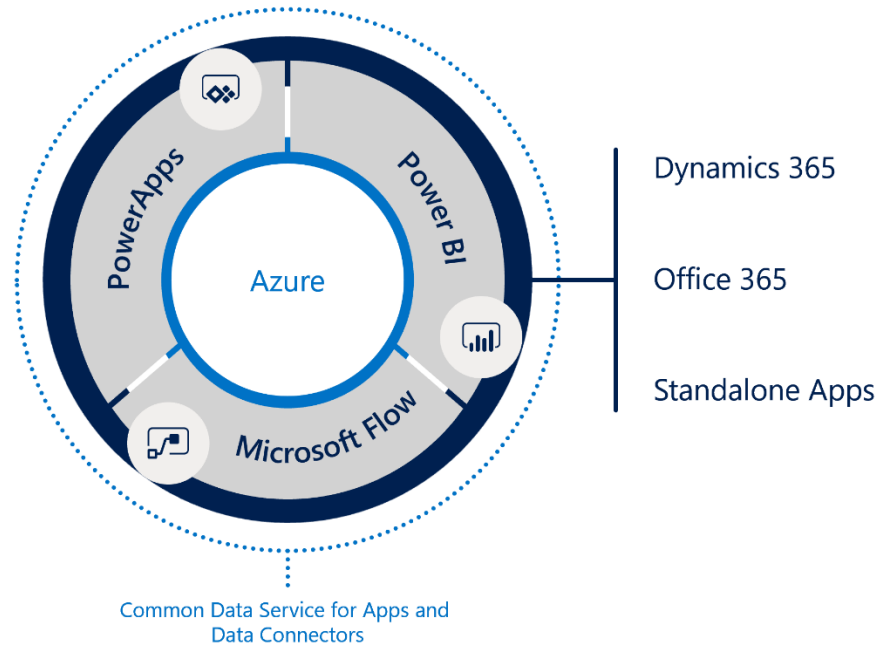
How to get started

While we of course recommend absorbing the whitepaper in its entirety, we thought it might be useful to give you some suggested areas on which to focus.

First 30 days	<p>Focus on gaining a good understanding of what you have in place already from both a licensing and application point of view. Take time to learn your organization's goals for the platform and how you can help them succeed. Here is your checklist:</p> <ul style="list-style-type: none">- Check to see what users have already built- Get familiar with the Admin Portals- Create some default Data Protection Policies- Setup a flow to notify you when users create new connections
First 6 Months	<p>Develop and implement your plan for environments, security and data protection policies. Put in place how you want teams to work with you to deploy new applications and establish an ongoing cadence for updates.</p>
After 6 months	<p>Use monitoring and insights to look for any needed adjustments in your environments. Look for ways you can improve your ALM processes by leveraging automation such as PowerShell and Microsoft Flow for common admin tasks</p>

PowerApps Platform Overview

Microsoft Power Platform



The PowerApps platform is part of the larger Microsoft Power Platform that also includes PowerBI and Microsoft Flow, leveraging the common infrastructure of the Common Data Service for Apps and Data Connectors. These capabilities are built on and leverage Microsoft Azure cloud services. Applications built on the PowerApps platform can also include Azure cloud services to scale from individual productivity to enterprise mission critical line of business applications.

PowerApps includes several key concepts/components you should be aware of, for many of them we will dive deeper into as we progress forward in the whitepaper. Here are some of the key ones to get started:

PowerApps Applications	These are the applications that users interact with on their desktop or mobile devices. There are two styles of applications; Canvas and Model-driven. PowerApps Canvas applications can also be embedded into SharePoint, Teams, Power BI and Dynamics 365 applications.
Microsoft Flow	Automated workflows that orchestrate across services using connectors. Flows can be triggered to run when events occur in other systems and services or scheduled to run at a specific time. Users can also interact with Flows in the mobile app by pressing virtual buttons.
Common Data Service for Apps (CDS)	A cloud scale datastore to manage data used by business applications. Data is stored within a set of entities. An initial schema is defined by the Common Data Model. CDS provides built-in capabilities for business rules, workflows, calculated and rollup fields and more.

Common Data Model	An open-sourced definition of standard entities that represent commonly used concepts and activities. Every CDS database starts with the entities defined as “core”. Application builders can add their own custom entities to support specific business scenarios.
Connectors	There are 200+ connectors that make it easy for application builders to connect to both Microsoft and 3 rd party services, from Dynamics 365 to Dropbox. The connectors allow Canvas Apps and Flows to easily use API (application programming interfaces) services without any developer knowledge. Custom connectors can also be configured to allow use of APIs that aren’t covered by the public connectors.
On-premises Gateways	On-premises gateway allows PowerApps and Flow to reach back to on-premise resources to support hybrid integration scenarios. The gateway leverages Azure Service Bus relay technology to security allow access to on-premise resources.

Usage Scenarios

PowerApps is a flexible platform and can be utilized in several different types of scenarios:

- 1 Individual/Team Productivity Applications**

With self-service scenarios, users are empowered to take their own ideas of how they can optimize what they do every day and express them in the form of a PowerApps app or Microsoft Flow automation. These assets can be shared with other team members and when successful promoted to be broader enterprise assets. Previously, these scenarios were out of reach and required high cost development resources to succeed. As an enterprise administrator your role is to put in place the guard rails to foster a healthy individual productivity while at the same time safeguarding sensitive business data and ensuring continuity when individuals leave your company.
- 2 Dynamics 365 Applications**

These 1st party Microsoft applications are built on and therefore deployed into PowerApps environments and utilize the Common Data Service for Apps (CDS) for data storage and core platform services. These applications are the quickest way to tackle common business scenarios like customer engagement, while still allowing tailoring to your company’s individual requirements. Custom PowerApps apps and Flows can be built to embed into or extend Dynamics 365 applications even further.
- 3 Apps from AppSource**

In addition to Microsoft built apps, 3rd party ISVs can also build on top of the PowerApps platform and are found via the AppSource marketplace. These apps can install into your existing environments or into their own depending on your unique needs

4 SharePoint, Outlook, Teams and Excel

PowerApps apps can also be embedded into the applications users already use. Often this increases user adoption because they don't have to learn a totally new application from what they are already using. PowerApps is now the primary way to customize SharePoint Online list forms. In the past, this required higher maintenance developer code to accomplish. As an administrator you will be enabling these experiences and ensuring users have the right permissions and policies to interact with the applications.

5 Mission critical line of business applications

Using the same tools and technique Microsoft uses to build Dynamics 365, enterprise customers can build their own line of business applications. These differ from the individual productivity scenario above in that they often solve broader more complex problems. These applications are also often built by dedicated teams tasked with implementing them. The teams typically follow a more defined process for building the application. As an enterprise administrator you will be helping them put in place the necessary Application Lifecycle Management (ALM) to facilitate development and day to day operations.

These are the key scenarios you will encounter, but not an endless list as it is really up to the capabilities and the creativity of your organization to determine how it leverages the platform. As an enterprise administrator, you can choose to either be a blocking force in the way of that creativity, or an enabler. As an enabler, you will put in place the necessary licensing, policies and processes needed to ensure success of the teams.

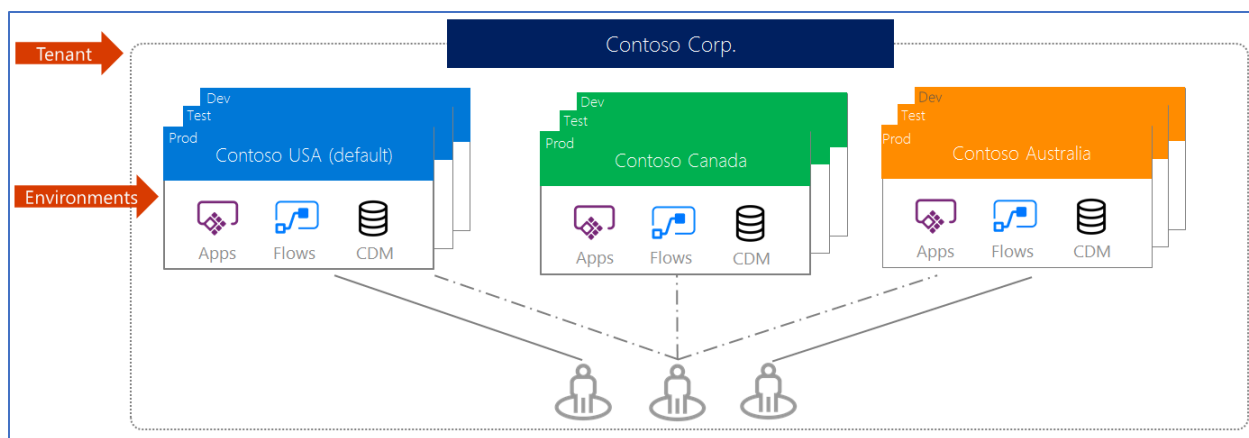
Platform Architecture

In this section we want to start drilling down in more detail on the components that make up the PowerApps platform.

Environments

Environments are containers that administrators can use to manage apps, flows, connections, and other assets; along with permissions to allow organization users to use the resources. Environments are tied to a geographic location that is configured at the time the environment is created. Environments can be used to target different audiences and/or for different purposes such as dev, test and production. The actual number and purpose of environments in your tenant is up to you as an administrator. In the ALM section of this paper we will cover some potential scenarios to help you choose what is best for you.

Common Data Service for Apps (CDS) databases are created in the context of environments. Each environment, if you are licensed for CDS, can have at most one database. If your organization signs up one of the Dynamics 365 Customer Engagement apps an environment with a CDS database will be created to support that application.



Environment Security Roles

Environments use security roles to determine what a user is able to do in the scope of that environment. The default roles that are available differ depending on if a CDS database has been created in the environment.

Environments without a CDS database have two built-in security roles: Environment Administrator and Environment Maker. Environment Makers can create and share apps, connectors, gateways etc. in the environment. Users in the Environment Maker, or Office 365 tenant Global Administrator role can all manage the environment which includes adding/removing users, creating the CDS instance, viewing and managing all resources created and setting Data Loss Prevention policies.

Once a CDS database has been created in an environment all users of the Environment Admin role will now be members of the System Administrator role instead. The CDS security roles will now take over for controlling security in the environment. Users or groups previously assigned Environment Maker role will need to be re-assigned manually one of the CDS security roles. The following are the initial CDS security roles that exist prior to you creating any custom roles.

Role	Description
System Administrator	This role takes over for the Environment Admin and has complete ability to customize and administer the environment. Users of the role also have full read-write access to data in the database. The role cannot be updated to change the privileges granted. Care should be taken in assigning this to the right people.
System Customizer	This role has full permission to customize the environment. The role's data access is focused only on data owned by the user. This role can be modified but it is not recommended to modify.
Environment Maker	Can create new resources associated with the environment including apps, connections, gateways and flows. There is no default privileges to data included. This role can be modified but it is not recommended to modify.
Common Data Service User	This is a basic user role, with ability to run apps and perform common tasks but no ability to customize the system. The data access is focused on Read access to most Common Data Model core entities with full access to records owned by the user. This is a good role to consider copying to make a custom security role for users.

Delegate	This is a special role really design to give a user permission to Act on behalf of another user. More details can be found here https://docs.microsoft.com/en-us/dynamics365/customer-engagement/developer/org-service/impersonate-another-user
----------	----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

In addition to these default roles, you can also create custom security roles. Custom security roles should be created to support applications built in your organization. Custom security roles can also come with applications you install from App Source or if your users sign up for Dynamics 365. You will read more about CDS security roles in the Security section of this paper.

Types of environments

There are multiple types of environments. The type of environment indicates the purpose and determines the environment characteristics. The following table summarizes the current types of environments that you might encounter.

Type	Description
Production	This is intended to be used for permanent work in an organization. It can be created and owned by an administrator or anyone licensed with a PowerApps P2 license. These environments are also created for each existing Dynamics 365 CDS database when it is upgraded to version 9.0 or later. Production environments are what you should use for any environments on which you depend.
Default	These are a special type of production environments. Each tenant will have a default environment created automatically and it has special characteristics described below in further detail.
Sandbox	These are non-production environments and when associated with a CDS database instance offer features like reset.
Trial	Trial environments are intended to support short term testing needs and are automatically cleaned up after a short period of time.
Developer	Developer environments are created by users with the Community Plan license. They are special environments intended only for use by the owner. Sharing with other users is not possible in these environments.

Default Environment

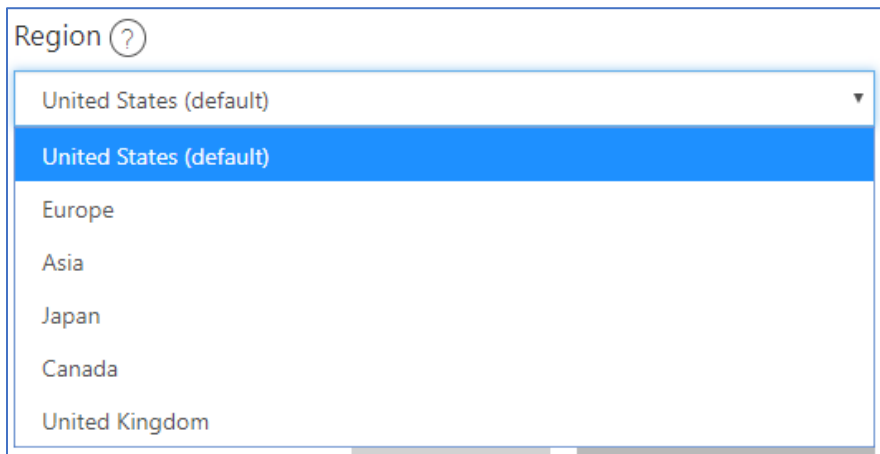
Each tenant will have a default environment created automatically in the region nearest the Azure Active Directory (Azure AD) tenant. This environment has a few unique characteristics from other environments that you create. This environment can't be disabled or deleted. All tenant users are added automatically to the maker role for the default environment and you can't remove them from that role. They are not however added automatically to the environment administrator role. This makes the default environment the perfect place for people to build personal productivity apps and flows.

The default environment is also the only place you can currently create gateways to connect to on-premises resources. So, if you have an application that needs on-premise resources the app, its connector and the gateway must be created and run from your organization's default environment. It is planned to allow creation of gateways in the non-default environments in the future.

Another unique consideration of the default environment is you can't create a Common Data Service for Apps database in the default environment. This however will be supported in the future.

Environment Regions

When you create an environment, you will pick a geographic location. Application components, including the CDS database will reside in that region. Generally, you will want to choose a location closest to the majority of your users that will be using applications in the particular environment. If you are connecting to other existing external resources, you should consider their location as well. You should also consider any data residency issues when choosing a location.

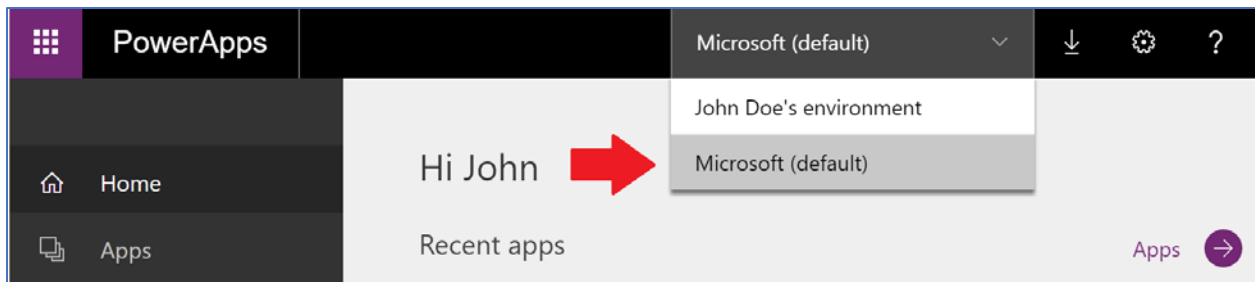


Who can create environments

As a global administrator in the admin portal you will be able to see a list of all environments created by users in your tenant. Administrators and users that have paid P2 or trial P2 licenses will be able to create new environments. Also users with the Community plan license can also create one Developer environment.

Impact of multiple environments on users

While it might be tempting to have users partitioned off into smaller environments it is important to consider the impact on the users in that decision. When users access the PowerApps Canvas App Player or the Flow application from the Web Browser or Windows Store the user will select and work within a single environment. By default, that environment will be set to the tenant default environment. Users can change their environment in the players and portals using the environment selector.



Having users' applications and other assets spread across multiple environments will result in the user frequently having to adjust their environment setting. The best user experience is when the user stays within a single environment for most of their daily use.

In the mobile applications the user is presented with a consolidated list of applications across the environments they have access to. Each application indicates the environment. This reduces the need to switch, however it introduces the need for the user to choose the correct application. For example, imagine if you had an application Device Ordering and it was deployed to environment Test and environment Production. If the user had access to both environments it would show up twice on the list. The user would have to differentiate between the two. Some of this can be minimized by only granting access as needed and then only temporarily to the Test environment.

Impact of multiple environments on Connectors

When an application uses a public connector (available for all tenants), the connector is configured for use within the context of an environment. Custom connectors are also configured in the context of an environment. If an app is moved to another environment the public connector references will be recreated upon import. Custom connectors must be re-configured manually in that target environment.

Applications that use the Common Data Service connector currently only can communicate with CDS databases in the same environment. This works well for apps that need to move between a dev, test and production instance because it adjusts automatically when imported into the next environment. Where this can be challenging is if you have two environments; one named Team Apps and another named CRM Data (which held your Dynamics 365 instance) an application using the CDS connector in the Team Apps environment would not be able to access data in the CRM Data instance. A current work around for this is to use the Dynamics 365 connector instead of the CDS connector since it can connect to multiple instances. That flexibility does result in more complexity if the application is moved from a dev, test to production and the instance needs to change as it is promoted, this must be done manually in the app once imported.

Impact of multiple environments on CDS

When thinking about how to organize your environments you should consider where your data lives. Having a single production environment with your CDS is the simplest configuration as it makes accessing data from apps the easiest. Having multiple environments, each with their own CDS database, might make sense in a few different scenarios. First, users have data that is geographically separated, and they don't share across those boundaries. Second, data from different applications that have conflicting incompatible use of CDS. Third, where users are building personal or team productivity applications that need CDS data but as an organization you aren't ready to mix that with the rest of your enterprise data.

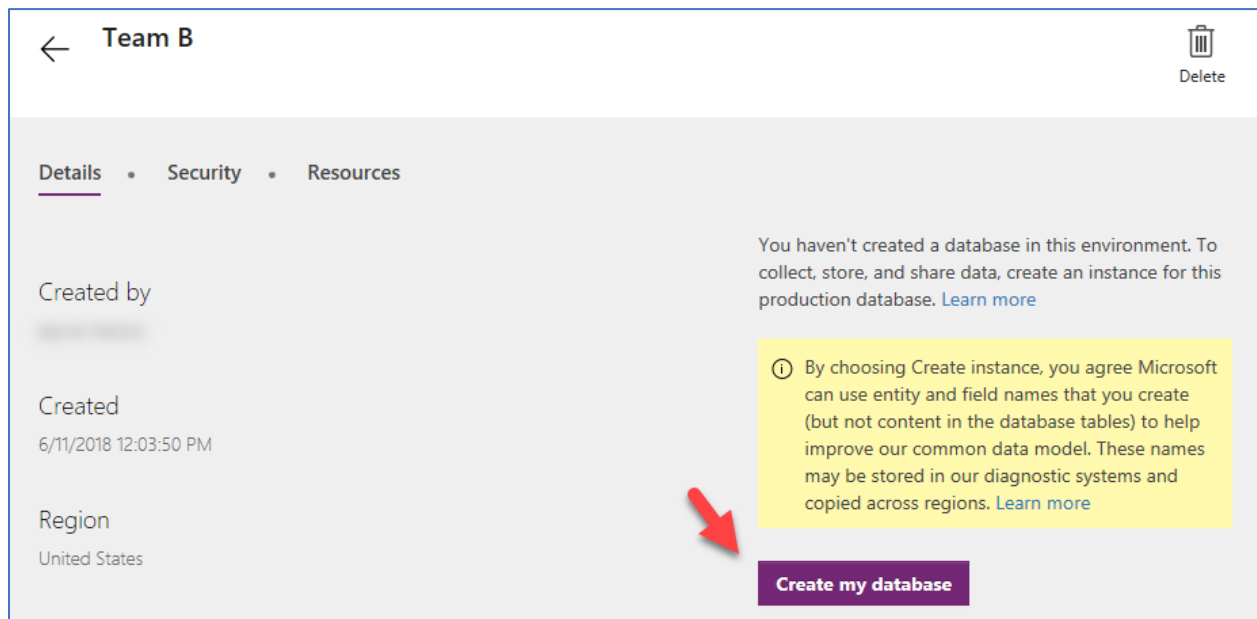
Common Data Service for Apps

The Common Data Service (CDS) for Apps is a cloud scale database used to securely store data for business applications built on PowerApps. CDS is an abstraction on top of underlying Azure cloud data management services to make it easier to build business applications. CDS provides not just data storage, but a way to implement business logic that enforces business rules and automation against the data. Data in CDS is organized as entities, for example account and contact would be two examples of entities. These entities can have relationships that define the business connection between the data stored in an entity. For example, John works for Contoso would be expressed as a relationship. The security model of CDS enables data protection down to the field level on individual records. A more thorough discussion of security will be covered in the security section of this paper.

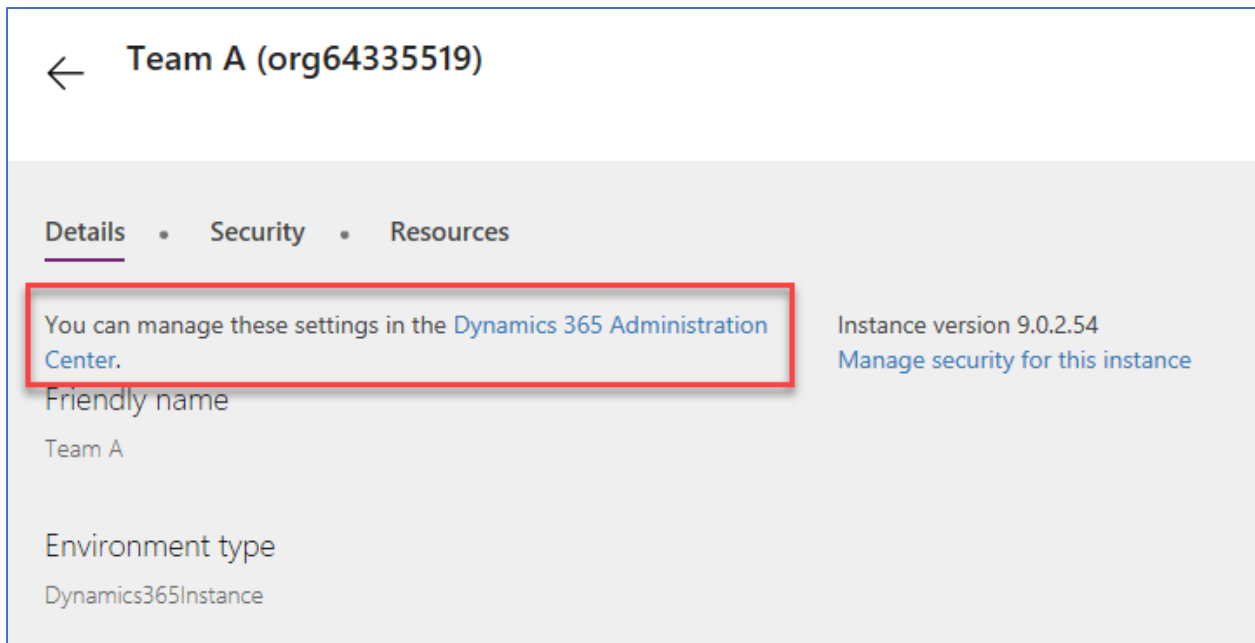
CDS databases are created in the context of a PowerApps environment. Each environment can have only a single CDS database. CDS databases can be provisioned by you or licensed individuals in your organization to support their custom applications. CDS databases are also automatically provisioned when a Dynamics 365 Customer Engagement application is added to your tenant.

Managing CDS database instances

The easiest way to know if you have a CDS database associated with your environments is to look at the detail page of the environment from admin.powerapps.com. If you see the Create my database button then you don't have one in that environment yet and can create one.



On the other hand, if you don't see the Create my Database link then the CDS database instance exists and you can click on the Dynamics 365 Administration center link to navigate to the list of all your CDS databases.



From the Dynamics 365 Admin center you can open the instance as well as manage and view some of the instance details. The actions you can take on each instance depends currently on if it started as an instance for Dynamics 365 Customer Engagement or if you started it with just the core CDS entities. For core CDS instances, you can only copy or set notifications. On Dynamics 365 CDS instances you also have the ability to reset the instance if it is of type sandbox, and potentially convert an instance to a sandbox to then test or reset.

CDS Backups and Restore

From here you can also see the database backups. As you can see from the following image, backups run automatically every day. No action is required by you, or any administrator to ensure daily backups.

Microsoft | Dynamics 365

Dynamics 365 Administration Center

INSTANCES UPDATES SERVICE HEALTH **BACKUP & RESTORE** APPLICATIONS

Manage backups

Backups for: Team A New backup

LABEL	DATE (UTC)	CREATED BY	STATUS	STORED ON	EXPIRES ON (UTC)
	6/11/2018 7:00:00 AM	System	Available	Dynamics 365	6/14/2018 7:00:00 AM
	6/10/2018 7:00:00 AM	System	Available	Dynamics 365	6/13/2018 7:00:00 AM
	6/9/2018 7:00:00 AM	System	Available	Dynamics 365	6/12/2018 7:00:00 AM
	6/8/2018 7:00:00 AM	System	Available	Dynamics 365	6/11/2018 7:00:00 AM
	6/7/2018 7:00:00 AM	System	Available	Dynamics 365	6/10/2018 7:00:00 AM
	6/6/2018 7:00:00 AM	System	Available	Dynamics 365	6/9/2018 7:00:00 AM
	6/5/2018 7:00:00 AM	System	Available	Dynamics 365	6/8/2018 7:00:00 AM

Dynamics 365 RESTORE

You do however have the option to manually take a backup. A great use for this is before doing big data imports or changes or deploying new releases of applications.

Backup for Team A

* Label:

Notes:

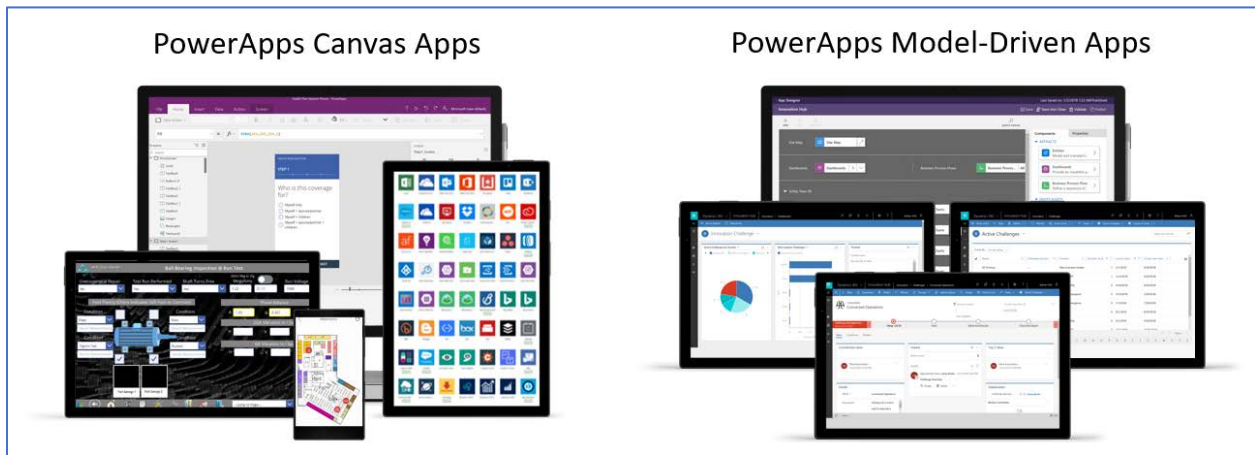
Product version: 9.0.2.54
 State: ready
 Purpose:

Note: Some backups may not be restorable. See: [Backup and restore instances](#)

Once the manual backup is completed it will show in the list of other backups allowing you to select it as the restore point.

Types of PowerApps

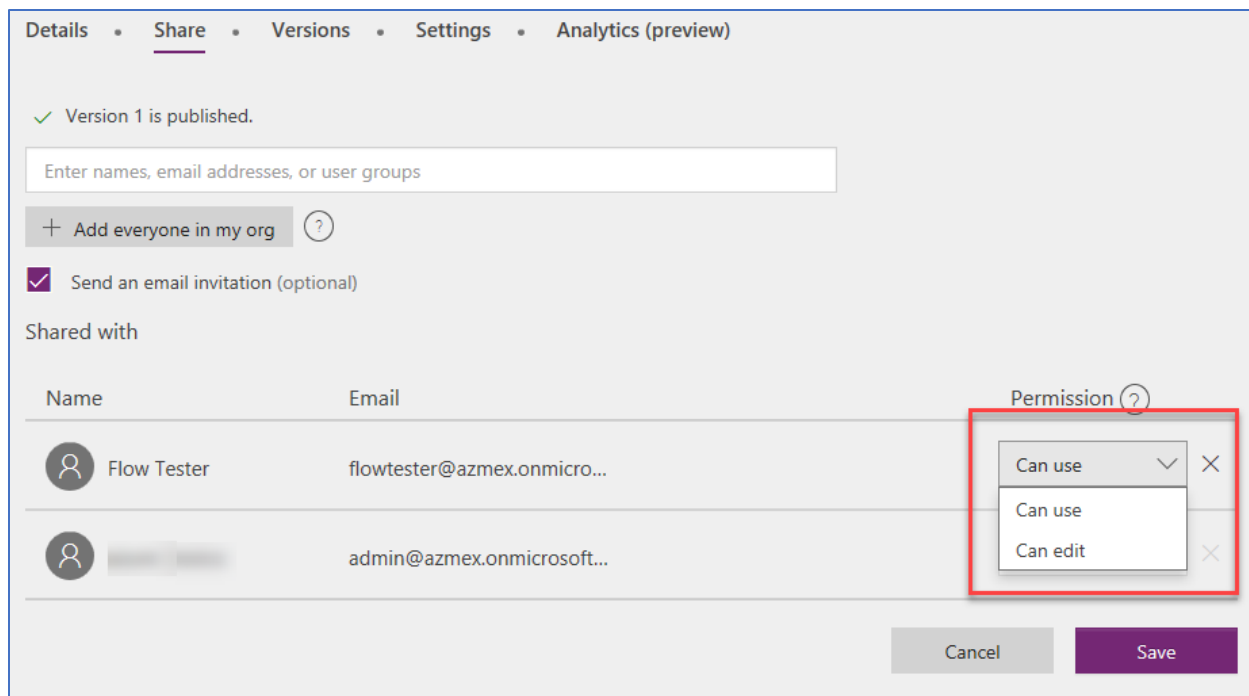
In the overview we hinted that there are two distinct types of applications PowerApps Canvas apps and PowerApps Model-driven apps and in this section, we will drill deeper into what you should be concerned with as an administrator. First, model-driven apps require a CDS database and are built on top of the data modeled in that database instance. Model-driven apps materialize views and detail screens based on the data structure. Because of this, they offer users a more consistent look and feel from one screen to the next without much effort by the creator. Canvas apps on the other hand can be built with or without a CDS database. They use connectors to access data and services. Canvas apps start with a blank screen like an artist's canvas and the creator manually lays out each screen. This allows the creator to have complete control of placements of controls on the canvas. Regardless of the two types, apps will be built in the context of a PowerApps environment.



It is also possible as the scenarios get more complex that your solution contains both types of apps.

User access to apps

Users obtain access to apps by having them shared with them. The technical specifics of how that sharing works is different between canvas apps and model-driven apps. For canvas apps they are shared with users, Azure AD Security Groups or with the whole organization. Model-driven apps you share by adding a user to a CDS security role that is associated with the application. We will cover more on CDS security roles in the Security section of this paper. The following is an example of sharing an app, where you can choose to also allow them access to edit in addition to using the app.



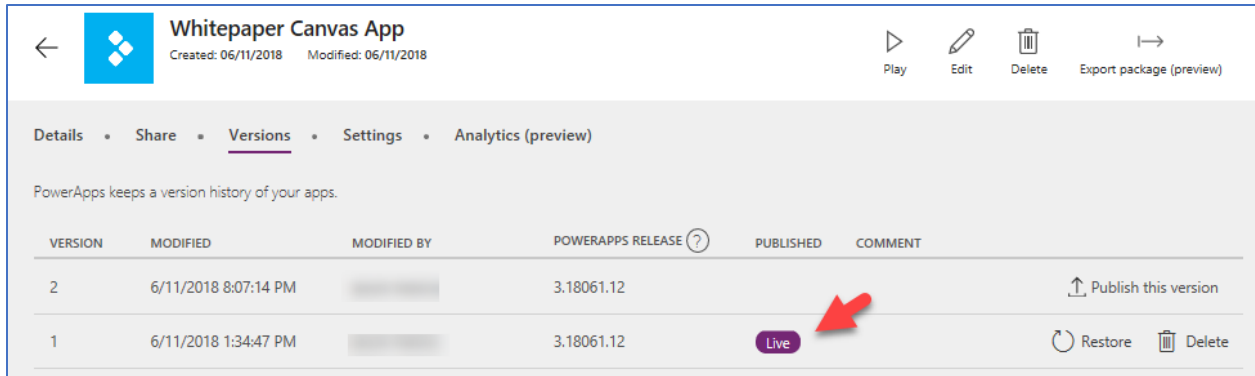
When you share an app the user will need access to the resources the app depends on. Some of the resources are shared automatically, others require the users you shared the app with to take action prior to use. We will cover more connectors shortly in the Connectors section.

Application Players

Both types of applications can be used as web applications from mainstream web browsers. Both types of applications can be discovered from web.powerapps.com. Dynamics 365 users can also discover them from home.dynamics.com application list as well as in the common application navigation list. Mobile users can run the application in a device installed player app on both phones and tablet devices. Currently, the player application for canvas apps is different from model-driven apps, but in the future that will be unified to a single player app.

Versions of the application

When you save a PowerApps canvas app it creates a new version of the application and it is published for the owner of the application and anyone that has permission to edit the app. Any other user that that application is shared with will still see the "live" version. Once ready, the new version can be published by explicitly clicking on the "Publish this version" link.



In the event the new version has problems, a prior version can be restored by clicking the Restore button next to that version. In the example above, there are two versions of an app. If the Restore button is clicked on version 1, PowerApps will create a new version 3 of the application that is identical to version 1. In this way history and audit information is preserved and the maker could elect to return to version 2 and fix issues at a later date. This light weight application lifecycle management (ALM) is perfect for productivity applications built by your organizations users without introducing them to the additional overhead of deploying to multiple environments.

For model-driven applications there is also a concept of publish that happens after change of most visual components in the application. For example, if you change the application navigation, users in the same environment will not see the change until Publish is completed. Restore is typically accomplished with model-driven applications by exporting a solution version and re-importing it to restore. We will cover more on the Solution Framework later in this paper.

Exporting and Importing apps

Both types of applications can be exported and then re-imported into other environments, both in the same tenant and in different tenants. Both export into a zip file, however the organization of the apps are different in their respective packaging. Canvas apps export standalone and model-driven will export along with any related CDS components. In the future, canvas app export functionality will be included in the CDS solution framework allowing you to have one solution package that represents all the components in your application. Exporting and Importing would allow a more complete application lifecycle management (ALM) than the light weight ALM versioning we described previously.









Today, when you export a canvas app, you will choose the action that will be taken in the target environment. You can also choose to add a comment on each resource.

Review Package Content

Choose your export options and add comments to provide instruction or add version notes.

#1 - Select the wrench icon or import setup link if you need to change how each resource will be imported in the target environment

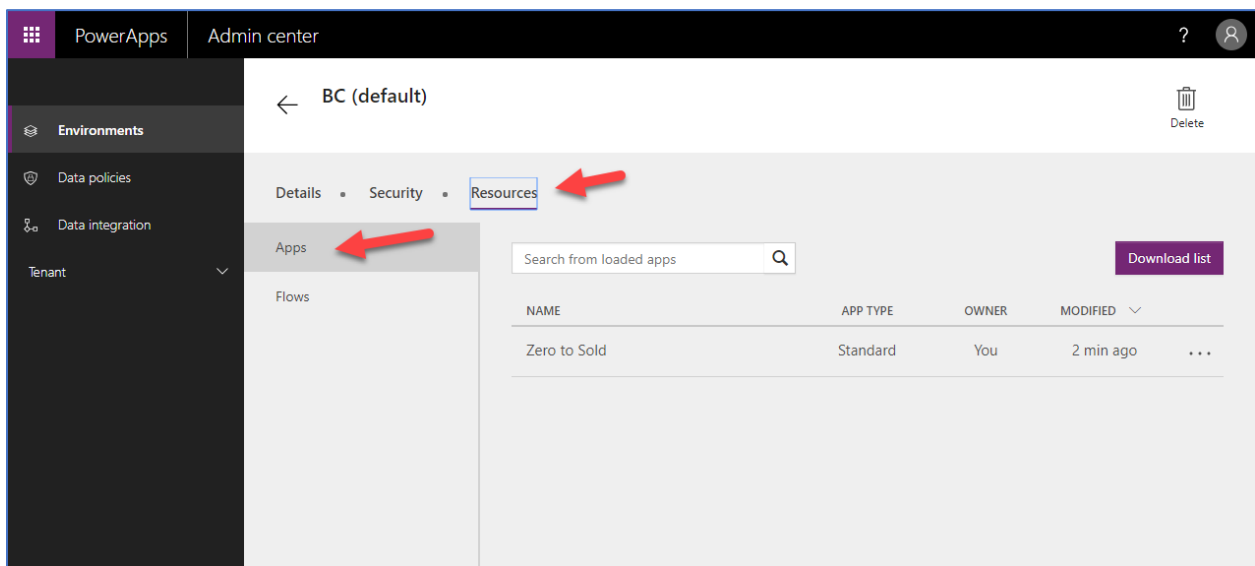
#2 - Select the comment icon to add a note or comment for a particular resource, these comments will be available during package import

NAME	RESOURCE TYPE	IMPORT SETUP	ACTION
Message Board App	App	Update	 
Related resources			
NAME	RESOURCE TYPE	IMPORT SETUP	ACTION
MLB	Connector	Select during import	 
Send notification email	Flow	Update	 
admin@powerflowbuild.onmicrosoft.com	Office 365 Outlook Connection	Select during import	 

On import, prior to completion of the import the related resources will need to be configured to have the proper connections established in the target environment. Custom Connectors and CDS customizations will need to be established prior to the import. If the Update action is chosen on import, the new version will be saved as a draft and will need to be “Published” before users will be able to use it. This allows an opportunity to test the application in the environment without impacting existing users.

What Apps already exist?

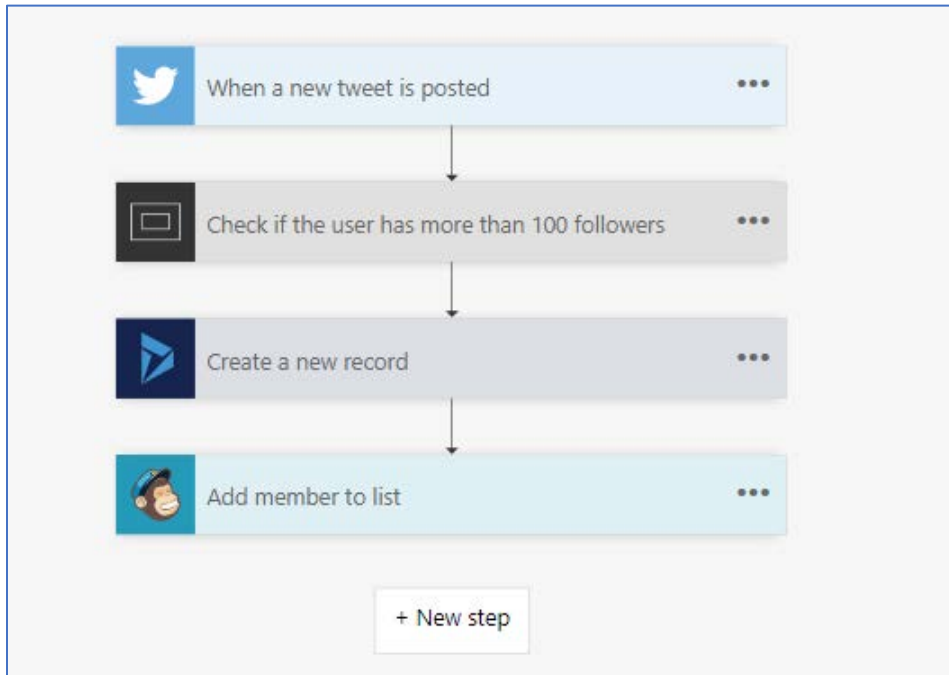
From the Admin Center admin.powerapps.com you can look at each environment and inside the Resources see a list of any apps that are associated with that particular environment.



Microsoft Flow

Microsoft Flow is an online workflow service that allows automating tasks across multiple services using connectors. Flows are started when a triggering event occurs, this could be a record is created or a scheduled execution, or even a button click from the Microsoft Flow mobile application. Once triggered, the flow proceeds to execute the actions in the flow. Conditions are used to guide the flow to the proper actions. You may find that it helpful to create some flows yourself to support your administration of your company’s PowerApp environments.

The following is a simple example of a Flow, with a trigger using the Twitter connector and three other actions that will run in sequence.



User access to Flows

By default, only the owner of the flow can execute the flow. The owner can invite other users and groups to be owners and this creates a “team flow”. All owners of a team flow can view the history, manage properties on the flow, edit the flow, add and remove other owners (but not the creator), and delete the flow.

Microsoft Flow vs Logic Apps

Microsoft Flow is built on top of Logic Apps. Logic Apps is an orchestration engine and part of the Microsoft Azure cloud service. Both services can be used to automate tasks and perform integration across systems. Using Logic Apps directly is more Pro Developer/Integrator focused and whereas Microsoft Flow is more focused on individual and team productivity. As flows start being able to be packaged with the CDS solution framework (coming soon) and moved between environments together with other CDS customizations expect to see more enterprise line of business applications relying on Flow in places where Logic Apps might have been the previously preferred solution.

A key advantage of Microsoft Flow is that it shares the same environments and connectors as PowerApps. This allows for easy interoperability between PowerApps and Flow. A PowerApp can directly invoke a Flow: for example, a scheduling app could call a Flow that asynchronously sends calendar invites to attendees. Or a Flow could use the push notification connector to send notifications to users’ mobile devices.

When used directly Logic Apps run in the context of an Azure subscription and you pay for each action that is invoked in the Logic App run. Microsoft Flows on the other hand are owned and run in the context of a user and the billing model counts the full execution towards an allotment you get with your licenses.

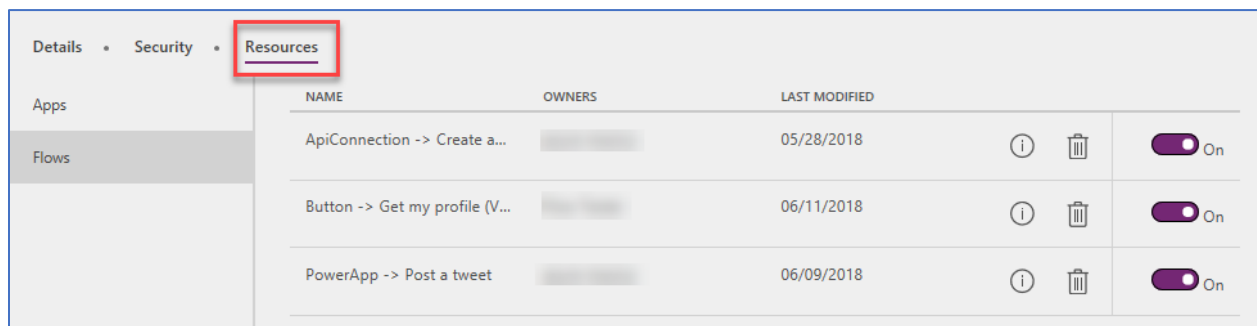
Exporting and Importing Flows

Flows can be exported and then re-imported into other environments, both in the same tenant and in different tenants. Today, flows export into their own zip file, separate from applications and CDS other components. In the future, flow export functionality will be included in the CDS solution framework allowing you to have one solution package that represents all the components in your application.

Flows can also be exported in a Logic App format, allowing conversion of the flow to a Logic App. This capability allows you to move from the flow execution model to the Logic App execution model as well as take advantage of some of Logic Apps more advanced features.

What Flows already exist?

From the Resources -> Flows tab on the environment details page you can get a list of all flows and who owns the flow. You also have the ability as an administrator to turn on/off the flow as well as delete it.



NAME	OWNERS	LAST MODIFIED			
ApiConnection -> Create a...		05/28/2018	i	🗑️	🔘 On
Button -> Get my profile (V...		06/11/2018	i	🗑️	🔘 On
PowerApp -> Post a tweet		06/09/2018	i	🗑️	🔘 On

You can also explore this information from the PowerShell cmdlets – we cover more details on that in the Management and Monitoring section

Connectors

Connectors are essentially proxy wrappers around the APIs provided by services that allow Microsoft Flow, PowerApps and Logic Apps to easily interact with the service. Connectors can be either public or custom. There are currently over 200+ public connectors that can be used by all organizations.

Examples of public connectors are Office 365, Common Data Service, Twitter, Dropbox and more.

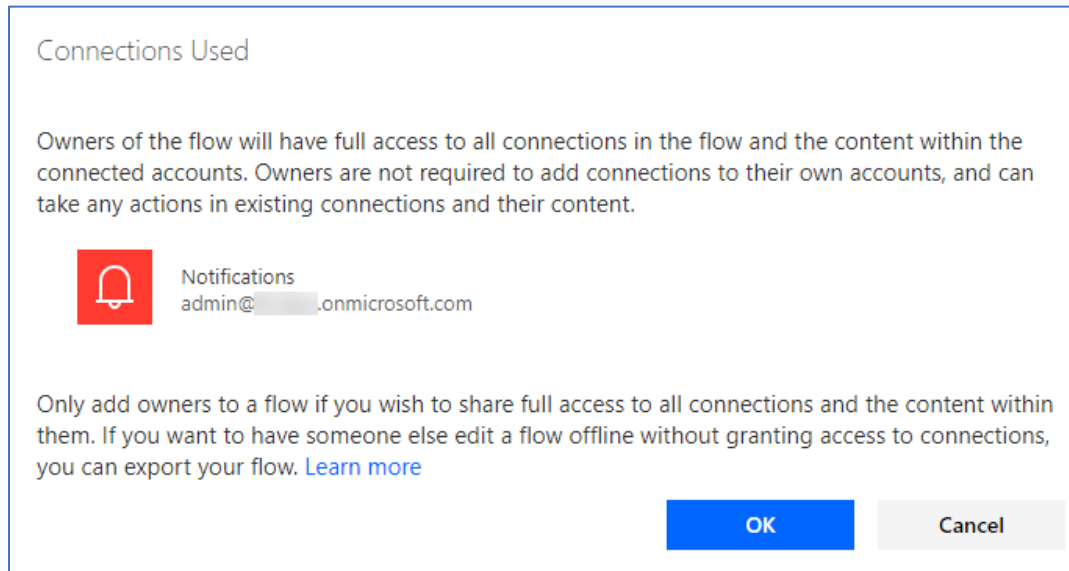
Custom connectors are defined in the context of an environment and are only available to apps and flows within that environment. Connectors make triggers and actions available that can be used by the apps and flows. Triggers are used by flow or Logic Apps to start the execution of the workflow. Actions are used by apps and flows to perform a defined set of actions during execution.

Sharing of canvas apps that use Connectors

Some connectors are shared automatically when you share the app. Others require that the user the app is shared with create their own connections. From web.powerapps.com you can check the connection and see if the share tab is present, if it is then the connection will be shared automatically. Otherwise, the user will need to create their own connection. Custom connectors are shared, but users must create their own connection to it. This means that the user shared with needs to have credentials or key if required by the custom connector.

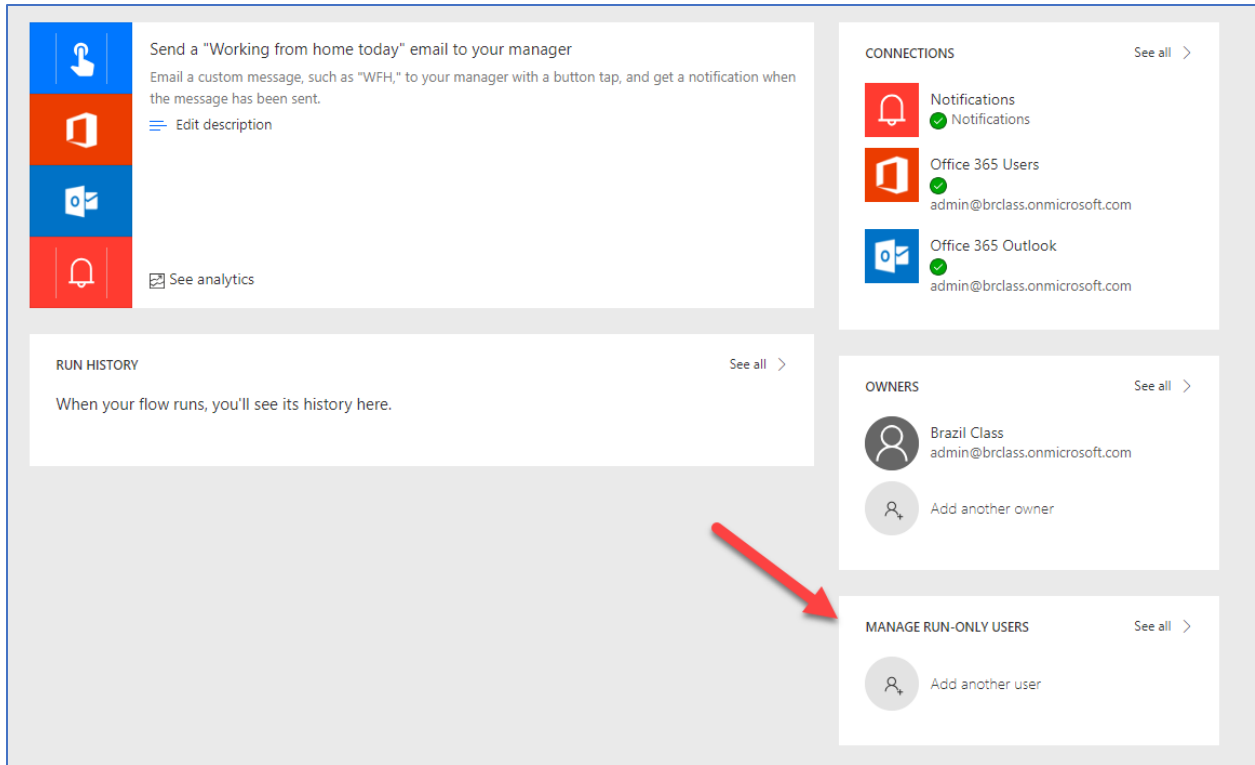
Sharing of Flows that use Connectors

Flows can be shared with other users either as co-owners or run-only users. When a user adds another user or group as an owner of a Flow those users will have full access to all the connections used in the flow. This means if they run the flow it will take the action in the context of the user signed into the connection. Because they are co-owners of the Flow they will also be able to modify the flow using the connections that already exist. They may also change the login on the connection, however they are not required to do so. Co-owners are limited to use the connection with that flow, they can't create a new flow and use the same connection. The following is an example of the warning that is presented when you add a co-owner.

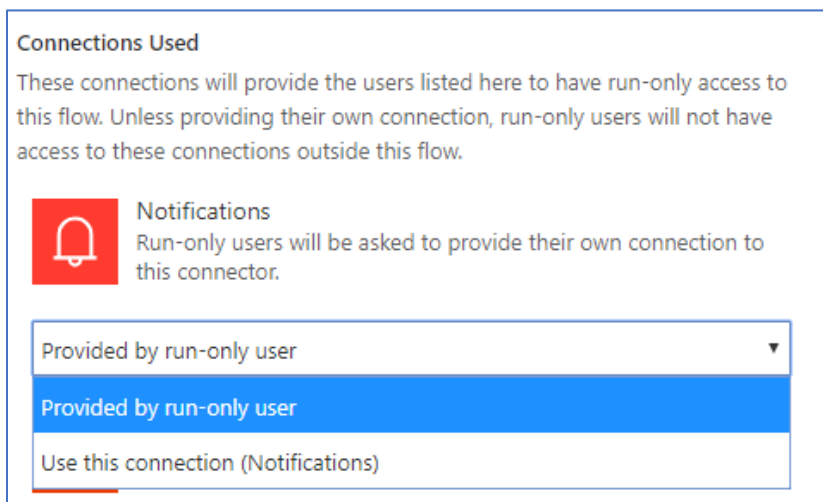


Now in this example notice it is an Admin account. Since it is sending a notification probably not a great concern but if it was a more sensitive connector this could allow escalation of privileges beyond what is intended in your security models.

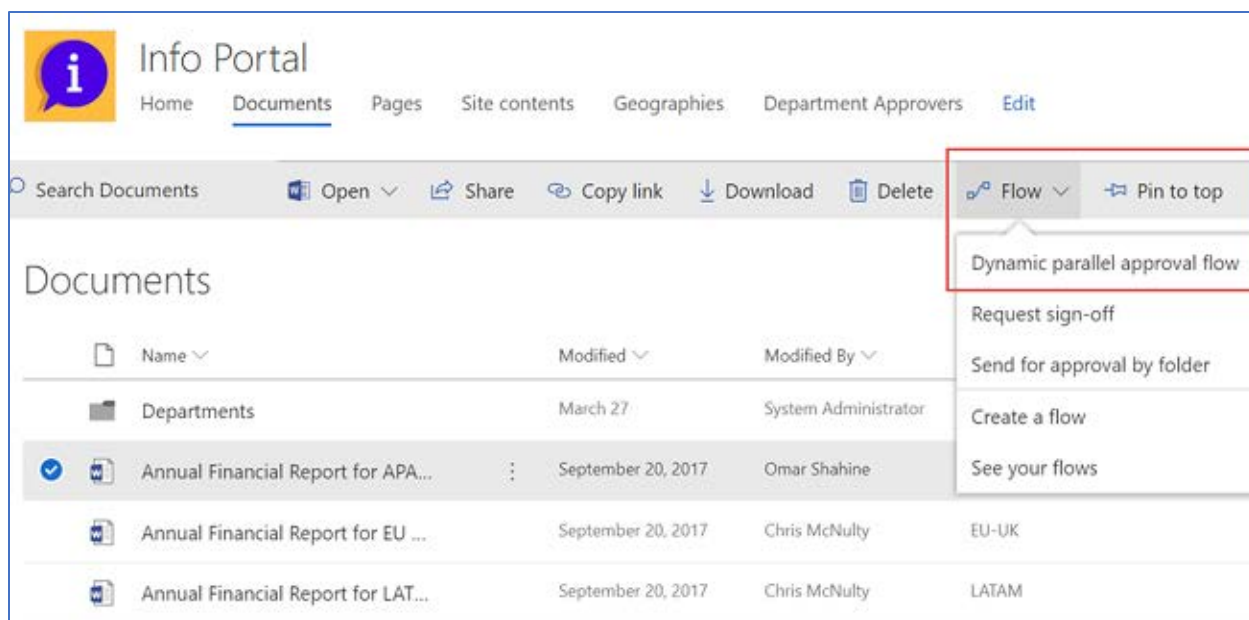
Run-only sharing is an option when the flow is manually triggered. This option allows greater control because first of all the user does not have ability to edit the flow just to run it. Second, when you invite the user you can specify to reuse the existing connection or require the user to provide their own. To manage the Run-Only users drill down on the Flow from the list of Flows and you will see the following:



From here you will see a dialog to specify the user or group as well as a list of the connections and the choice for each on how to grant access. The following shows the connection configuration and how you can choose to force the user to sign-in to their own connection.



One of the more recent additions is the ability to share a flow with a SharePoint List or an Office 365 Group. In this scenario, the Flow is available to all members of the group in the case of Office 365 groups. For SharePoint Lists, anyone with edit access to the list would have access to the flow. The flow would then show up with the ability to execute it from the application navigation.



Restricting Use of Connectors

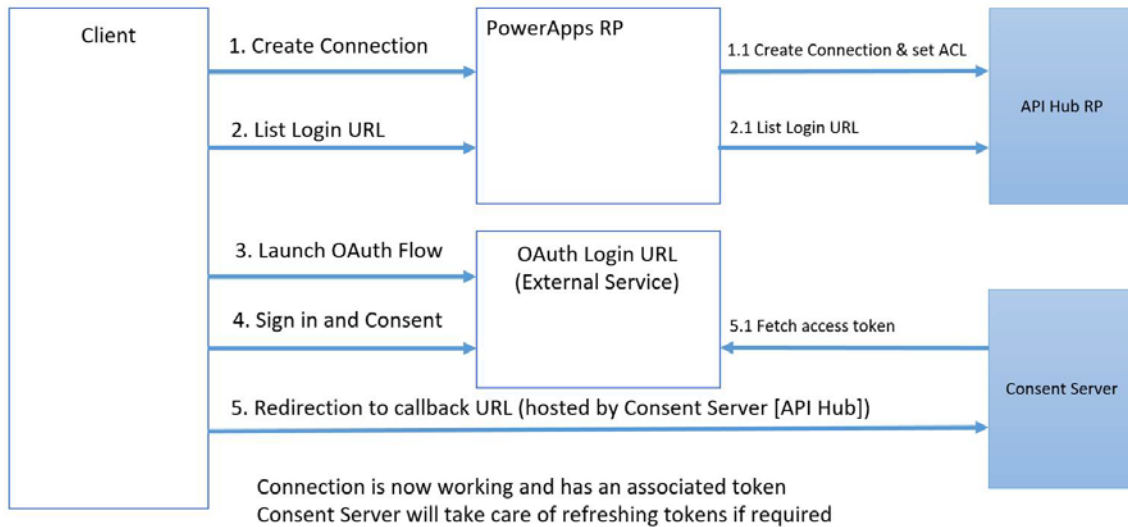
Within each environment using data loss prevention policies you can limit what connectors can be used together in a single application or flow. More on this in the section where we cover Data Loss Prevention (DLP) policies in this paper.

Connector Authentication Patterns

PowerApps and Flow authenticate with connectors to create a connection instance. It is that instance that contains the specific configuration information necessary for the app or flow to talk to the connector API that is used in each interaction. Connectors could choose to use no authentication, basic authentication, API key authentication or OAuth 2.0. The most common are OAuth and API Key.

OAuth if you aren't familiar with it is an authorization framework that allows external applications to obtain controlled access to a target service. Many APIs support it including CDS, Facebook and Twitter to name a few. The goal of authentication is to allow the user to sign in to a familiar login dialog, consent to the application using the service, and then setup to allow tokens to be acquired. It is the tokens that are used on each request to prove who the user is and their right to use the API. In the PowerApps and Flow usage, a Consent Server is involved that helps manage the tokens and their lifecycle including storing the renewal token in the Consent Server and handling the refresh cycle. The following is a step by step look at what happens when you authenticate a connection using OAuth.

OAUTH Consent Flow



The API Key is a little less complex as it typically involves the API assigning a key that is passed on each request. That key is provided when the connection is established for the connector and is stored in the environment with the other connection information in a secure way. An example of an API Key authentication connector is the Azure Storage Blob. As you can see below it wants the Storage Account Name as well as the Access Key.

10
01
Azure Blob Storage - When a blob is added or modified (properties on...
...

*** Connection Name**

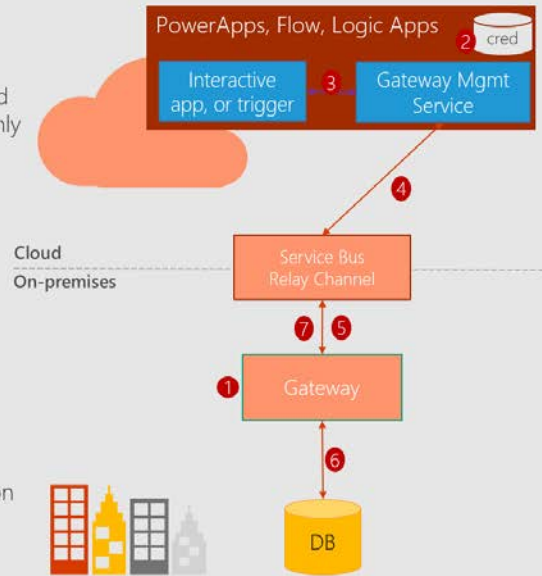
*** Azure Storage Account name**

*** Azure Storage Account Access Key**

When on-premise gateways are involved the process is even a little more complex. The following diagrams what happens when you establish a connection with the gateway data source.

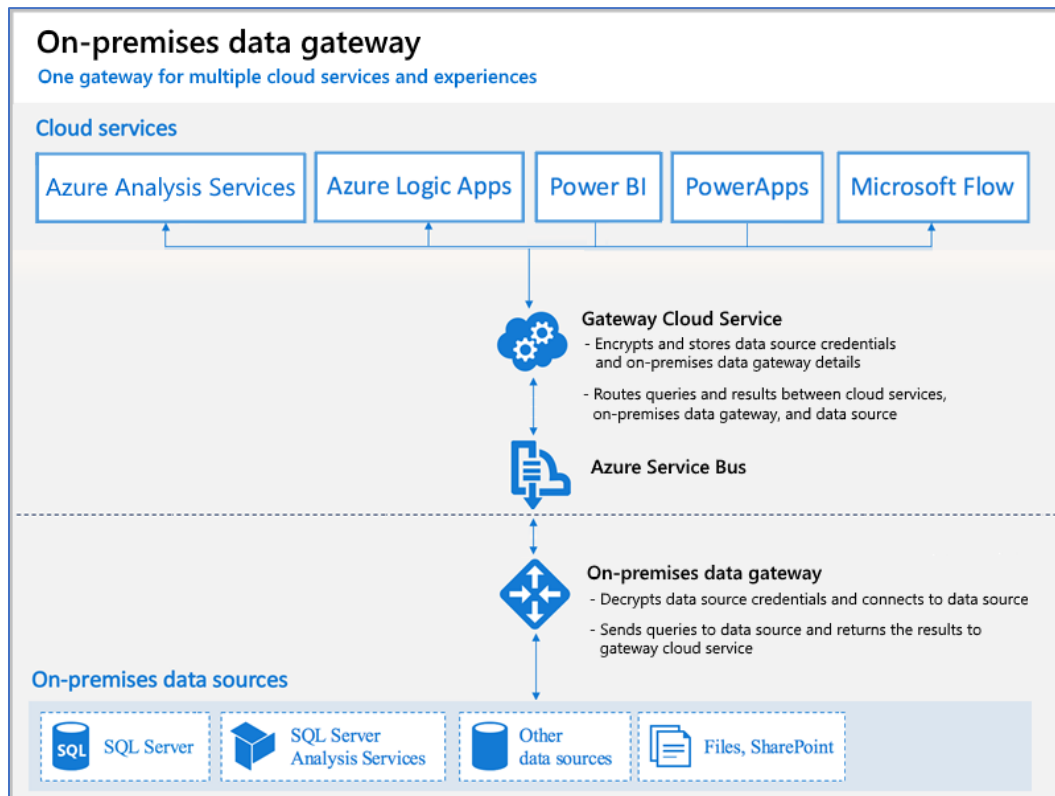
Architecture: Connections in PowerApps, Flow, Logic Apps

1. Gateway is installed & configured. During configuration, a corresponding service bus instance is also configured.
2. Credentials configured for a connection are encrypted, tested, and then stored encrypted with the connection object in the cloud. Only the corresponding on-prem gateway can decrypt the credentials.
3. A user in a PowerApps application, or a trigger/schedule in Flow/Logic Apps, starts a request using an on-prem connection.
4. The cloud service checks ACLs and the connector interacts with the Gateway Management Service to acquire permission to communicate with the associated gateway's Service Bus communication channel.
5. Gateway polls its Service Bus channel for pending requests. It accepts the pending request.
6. Gateway gets the request details, decrypts the credentials, sends query/API call to the target data source/API endpoint for execution
7. After execution, gateway securely streams the data back to the calling cloud service.



On-premises Gateways

The on-premises gateway allows PowerApps and Flow to reach back to on-premise resources to support hybrid integration scenarios. The gateway leverages Azure Service Bus relay technology to securely allow access to on-premise resources.



Gateway On-premise Install

The gateway service must run on a local server in your on-premise location. The server does not have to be the same one as the resources it will proxy access to, however it should be on the same local network to reduce latency. It does however need to be able to access the target resource with as low of latency as possible. Multiple application and flow connections can use the same gateway install. You can only install one gateway on a server.

During the install the gateway is setup to use NT Service\PBIEgwService for the Windows service logon. You can switch this to a domain user or managed service account if you'd like.

Gateway Administration Access

By default, you have this permission on any gateway that you install. As the administrator you can grant another users permission to co-administrate the gateway. It is recommended you always have multiple administrators specified to handle employee events in your organization.

Use of stored credentials

When you setup a data source on the gateway you will need to provide credentials for that data source. All actions to that data source will run using these credentials. Credentials are encrypted securely, using asymmetric encryption before they are stored in the cloud. The credentials are sent to the machine, running the gateway on-premises, where they are decrypted when the data source is accessed.

Port Usage

The gateway service creates an outbound connection to Azure Service Bus so there are no inbound ports required to be open. The outbound connection communicates on ports: TCP 443(default) , 5671, 5672 9350 through 9354.

It is recommended that you whitelist the IP addresses for the data region in your firewall. You can download the latest list here <https://www.microsoft.com/en-us/download/details.aspx?id=41653> These IP addresses are used for outbound communication with Azure Service Bus.

Gateway Access

Most of the PowerApps and Flow licenses have access to use the gateway with the exception of some of the lower end Office 365 licenses (Business and Office Enterprise E1 SKUs) .

Updates to the Data Gateway

Updates are not auto-installed for the On-premises data gateway. It is highly encouraged to remain current with the latest data gateway version as the updates to the gateway is are released on a monthly basis.

Gateway Disaster Recovery

A recovery key is assigned (i.e., not auto-generated) by the administrator at the time the On-Premises Data Gateway is installed. The recovery key is required if the gateway is to be relocated to another machine, or if the gateway is to be restored. Therefore, the key should be retained where other system administrators can locate it if necessary.

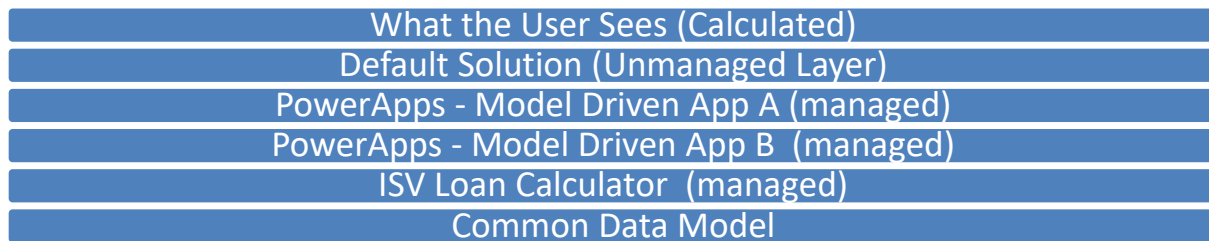
Solution Packages

The Common Data Service Solutions Framework provides solutions as containers to track and manage customizations in a CDS instance. This includes entity metadata, forms, views, and other resources

required to run the app including developer compiled code assets. A project solution starts in the CDS instance where the app is created, and the container is used to track any change made to support the app. The solution can then be exported from that CDS instance for transit to other CDS instances. This is commonly used to promote an application from a development instance to test and then finally to a production CDS instance. Today, canvas apps and flows have their own packaging and are not included in the CDS solution package but will be in the near future.

Types of Solutions

There are two types of solutions, managed and unmanaged. Solutions start out as unmanaged, meaning their components can be modified. Managed solutions are locked down, meaning you can't directly modify the components. Managed solutions are created by exporting an unmanaged solution and requesting it be exported as managed. That solution when imported into another target CDS instance is then installed in a managed state. Components in the managed solution can't be directly modified, but they can be added into another unmanaged solution that tracks changes as a separate layer. Multiple managed solutions that are installed in the same CDS instance create layers that combine for what the users see as the effective set of customizations.



Creating Solutions

Each PowerApp environment has a default solution created automatically as an empty solution when the CDS instance is created in the environment. Directly in the CDS instance you can create additional unmanaged solutions and manage their components using Solution Explorer.

Installing Solutions

Solutions can be installed into a CDS instance if all their dependencies have been met. A solution becomes dependent when it uses something from another solution. Those dependent solutions must be installed first. Solutions can be installed directly into a target CDS instance from the Solution Explorer. Solutions can also be deployed using the Package Deployer tool which can deploy a set of solutions along with data into a CDS instance. Package deployer can be run interactively, or from PowerShell. Package Deployer is how Microsoft AppSource marketplace installs apps. Importing a managed solution is different than importing an unmanaged solution. When you import an unmanaged solution, the changes are merged in with other unmanaged changes in that CDS instance. These merged changes can only be removed by manually removing each individually. The administrator must also publish the unmanaged changes to have any non-schema (e.g. display labels) changes be visible to other users.

Uninstalling Solutions

Solutions are uninstalled by deleting them from the CDS instance. The result of the delete action varies greatly between managed and unmanaged solutions. Because unmanaged solutions are merged in with

other changes, it is not possible to remove them as a unit. Removing an unmanaged solution simply removes the solution container and all the components remain in the instance. The remaining components must manually be removed one by one. In fact some unmanaged changes must be reverted manually such as a label change. Managed solutions act more like a true uninstall, it removes all the solution components that were installed if nothing new has taken a dependency on them. This includes any data from entities that were only defined and used by that solution being removed. So, take care when removing solutions that you no longer need the data. In many cases you might find that you want to first export the data before the remove/uninstall.

Licensing and License Management

Organizations can obtain licenses by either licensing Microsoft PowerApps or Flow specifically or by it being included in the license of another Microsoft cloud service offering. For example, both Office 365 and Dynamics 365 provide entitlements for PowerApps and Microsoft Flow. As with most Microsoft licensing, you can mix and match for users as appropriate giving some additional entitlements.

Regardless of how obtained all licenses are user based. In the rest of this section we will highlight some of the key points of licensing, but it is not the product licensing documentation, you should consult that for any of the latest details. Links for pricing and specific plan details can be found later in this section of the paper.

First, let's look at what you, as the administrator, will need to have the best administrator experience. While you can do basic administration with any of the licenses with PowerApps entitlements, the best experience is with PowerApps P2 and that is what Microsoft recommends for administrators. This provides the ability to create additional environments as well as Common Data Service for apps. It also provides the best experience in the administration centers for controlling the environments.

The following summarizes the access based on various administrator role and license combination, as you can probably easily tell Global Admin with a PowerApps P2 license provides the most complete administrator experience. Without a PowerApps P2 the administrator can view some information but only able to administer their own assets.

License Type	User/License Management	PowerApps Admin Portal	Manage Environments	Data Loss Policies	Dynamics 365 Admin Center
Global Admin without PowerApps P2	Yes	Sees only DLP policies and Tenant level user reports and statistics	No	Can create but only for all environments	Can view CDS instances
Global Admin with PowerApps P2	Yes	Full access	Yes – all environments	Full ability to view, create, modify and remove	Full access
User Management Role	Yes	No access	No	No access	No access

User Management role with PowerApps P2	Yes	No tenant level info, only own environments	Only own	Only own	No Access
Dynamics Admin role	View Only	No access	No	No access	No access
Dynamics Admin role with PowerApps P2	View Only	No tenant level info, only own environments	Only own	Only own	Full access

PowerApps has two primary standalone licensing tiers. PowerApps P1 which is best suited for business users who need to use basic PowerApps applications. PowerApps P2 is more focused towards Makers and Administrators who want to create data models in the Common Data Service. Each PowerApps P2 user is entitled to create two Production environments each having a Common Data Service for Apps database. Users of the apps built on the Common Data Service for Apps only require PowerApps P1 unless they use advanced features like plug-ins and real-time workflows or work with Dynamics 365 restricted entities (a list of these can be found here <https://docs.microsoft.com/en-us/powerapps/maker/common-data-service/data-platform-restricted-entities>).

For example, John could create a PowerApps Canvas application that stores data in the Contact entity and two custom entities in a Common Data Service environment he created. For this, he would need a PowerApps P1 or P2 license to build the application and customize CDS. Mary and Henry are users of the application John built. John had shared the application with them, so they could use it. Mary and Henry would be ok with just a PowerApps P1 license or a license that came with Dynamics 365. If either of them had only PowerApps that came with Office 365 they would however need to upgrade their license to at least a PowerApps P1.

Building on that example, John asked George, a developer at the company to create a plug-in on one of the custom entities. The business logic would do some automated processing every time the data was updated. This advanced business logic usage would require now for all users that wanted to use the canvas app to be upgraded to a PowerApps P2 license. You can find additional Entity Licensing examples here <https://docs.microsoft.com/en-us/powerapps/maker/common-data-service/data-platform-entity-licenses>

PowerApps licenses include an equivalent Microsoft Flow license. It is also possible however to license flow by itself. Flow also has a free plan. All Flow plans offer unlimited creation of Flows but vary based on number of runs included and the time delay for checking for new work to perform. In addition to the key differences documented in the chart below, it is important to note that with the free plan the runs are per person where runs for other licenses aggregate at the tenant level.

Flow Plan	Number of Runs	Check for new work
Free	750	Every 15 minutes
Office and Dynamics 365	2,000	Every 5 minutes
P1	4,500	Every 3 minutes
P2	15,000	Every 1 minute

Organization level accumulated entitlements

Licenses for Flow include a specific number of executions of the automation for each user. These allowances are accumulated for all users in your organization on all plans except for the Free plan. For example, if you had 100 users each with 2000, your organization would have 200,000 monthly runs. That means if one user uses a large number you will not be penalized as long as you stay under your organization total accumulated allowance.

Use of connectors

Apps and Flows use connectors to interact with services. Connectors can be standard, premium or custom. To use premium connectors users must be licensed with PowerApps P1 or P2 licenses.

Trial Plans

Trial plans are available for both PowerApps and Microsoft Flow plans 1 and 2. Free trials last 30 days for PowerApps and 90 days for Microsoft Flow plans. Users can self-service sign up for these trials in your organization. This can be done by explicitly visiting the pricing pages or by being prompted when they attempt an action in the apps that require additional licensing.

For Microsoft Flow, an unlicensed user that signs in to flow.microsoft.com will be setup with the free Flow plan. If later they try to perform an action like sharing a Flow, they will be prompted to sign up for a trial. In this example, if the user accepted the offer for trial they would be signed up for a Flow Plan 2 trial. This trial would not show up under the user licenses in the Office 365 Portal, however you would be able to see it in the Power Apps license report discussed later in this security section.

For PowerApps, if a user signs up for a PowerApps P1 trial they will be upgraded to a PowerApps P2 trial if needed for any of the actions they take such as creating an environment. If they sign up for the trial by visiting web.powerapps.com it will start as a PowerApps P2 trial.

As the administrator, you will likely be assisting users that had started in a trial and either want to continue experimenting or are ready to get a regular license to keep working with the app they are building. If you are moving to a regular license for a user, it would also be a good time to work with them to see if their app should stay where it was built or should be moved according to the environment strategy you adopt. For those not ready to get a full license but want to keep experimenting you could help them get setup on the community plan and help them move their application and flow assets into their new developer environment.

PowerApps Community Plan

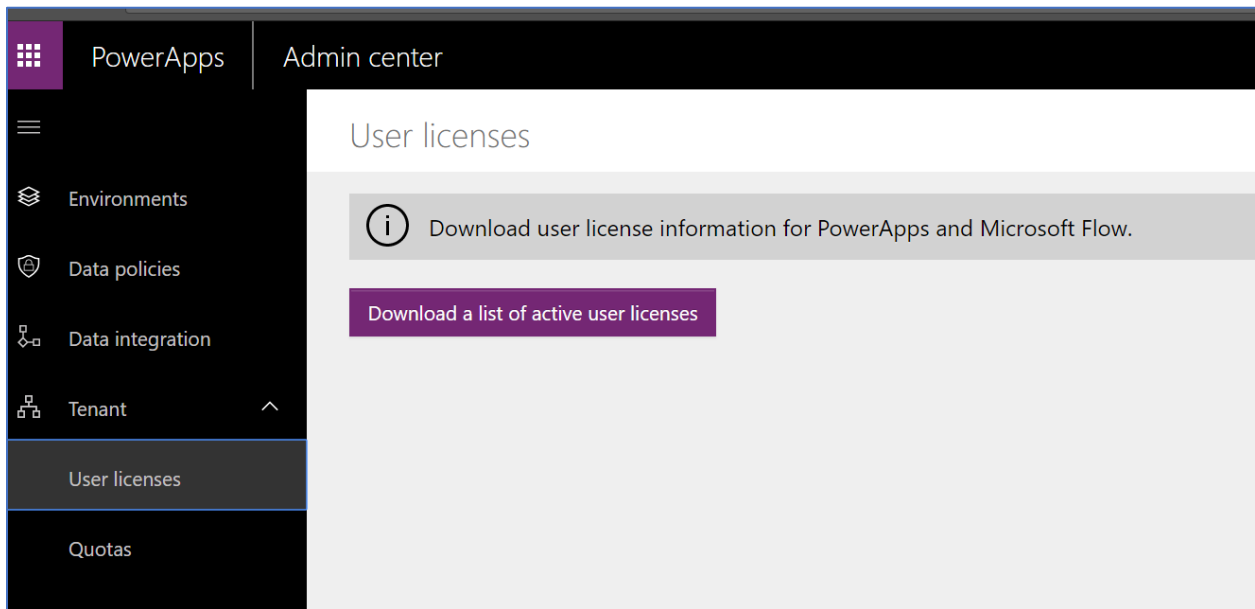
In addition to the trial plans, there is also a free PowerApps Community Plan. This is a special plan that allows individual self-service sign up and it provides an individual environment that the user can use to build apps and flows. These environments will show up on the administrator's list of environments and will list the type of environment as "Developer". The environments are for individual use, so there is no

ability to share with other users. Users in your organization can self-service signup for this plan even if they have PowerApps and Flow license entitlements via another licensing plan. Signup for the community plan can be found here <https://powerapps.microsoft.com/en-us/communityplan/> and more details on its features here <https://docs.microsoft.com/en-us/powerapps/maker/dev-community-plan>

What users are licensed

You can always look at individual user licensing in the Office 365 admin center by drilling into specific users. From the PowerApps administration center you can also produce a report focused on PowerApps licenses. This is one of the steps we recommend you do right away as a new administrator trying to understand your current licensing.

You can download the report from **admin.powerapps.com -> Tenant -> User Licensing**



The report is an Excel workbook that once downloaded you can use all of Excel's features to filter the data to what you are looking for. The following is an example of the downloaded workbook.

	A	B	C	D	E	F
1	User Name	Email Address	Service	License	License Assigned Date	Is Trial
2	Alex Wilber	AlexW@BAPpartners.onmicrosoft.com	Flow for Office 365	Office 365 Enterprise E5	11/29/2016 18:48	FALSE
3	Alex Wilber	AlexW@BAPpartners.onmicrosoft.com	Flow Free	Flow Free	8/24/2017 19:45	TRUE
4	Alex Wilber	AlexW@BAPpartners.onmicrosoft.com	PowerApps for Office 365	Office 365 Enterprise E5	11/29/2016 18:48	FALSE
5	Alex Wilber	AlexW@BAPpartners.onmicrosoft.com	Flow Plan 2 Trial	Flow Plan 2 Trial	8/24/2017 19:45	TRUE
6	Alex Wilber	AlexW@BAPpartners.onmicrosoft.com	PowerApps Trial	PowerApps Plan 2 Free Trial	6/19/2017 18:31	TRUE
7	Nestor Wilke	NestorW@BAPpartners.onmicrosoft.com	Flow for Office 365	Office 365 Enterprise E5	11/29/2016 18:57	FALSE
8	Nestor Wilke	NestorW@BAPpartners.onmicrosoft.com	PowerApps for Office 365	Office 365 Enterprise E5	11/29/2016 18:57	FALSE
9	Grady Archie	GradyA@BAPpartners.onmicrosoft.com	Flow for Office 365	Office 365 Enterprise E5	11/29/2016 19:08	FALSE
10	Grady Archie	GradyA@BAPpartners.onmicrosoft.com	PowerApps for Office 365	Office 365 Enterprise E5	11/29/2016 19:08	FALSE
11	Isaiah Langer	IsaiahL@BAPpartners.onmicrosoft.com	Flow for Office 365	Office 365 Enterprise E5	11/29/2016 18:58	FALSE
12	Isaiah Langer	IsaiahL@BAPpartners.onmicrosoft.com	PowerApps for Office 365	Office 365 Enterprise E5	11/29/2016 18:58	FALSE
13	Patti Fernandez	PattiF@BAPpartners.onmicrosoft.com	Flow for Office 365	Office 365 Enterprise E5	11/29/2016 18:50	FALSE
14	Patti Fernandez	PattiF@BAPpartners.onmicrosoft.com	PowerApps for Office 365	Office 365 Enterprise E5	11/29/2016 18:50	FALSE
15	Megan Bowen	MeganB@BAPpartners.onmicrosoft.com	PowerApps AppSource Trial	PowerApps App Source Trial	5/11/2017 22:12	TRUE
16	Megan Bowen	MeganB@BAPpartners.onmicrosoft.com	Flow for Office 365	Office 365 Enterprise E5	11/29/2016 19:03	FALSE
17	Megan Bowen	MeganB@BAPpartners.onmicrosoft.com	PowerApps for Dynamics 365	Dynamics 365 Plan 1 Enterprise Edition	3/3/2017 20:59	FALSE
18	Megan Bowen	MeganB@BAPpartners.onmicrosoft.com	Flow Free	Microsoft Power Apps & Flow	6/26/2017 13:22	TRUE
19	Megan Bowen	MeganB@BAPpartners.onmicrosoft.com	Flow Free	Flow Free	1/16/2017 18:14	TRUE
20	Megan Bowen	MeganB@BAPpartners.onmicrosoft.com	PowerApps for Office 365	Office 365 Enterprise E5	11/29/2016 19:03	FALSE
21	Megan Bowen	MeganB@BAPpartners.onmicrosoft.com	Flow for Dynamics 365	Dynamics 365 Plan 1 Enterprise Edition	3/3/2017 20:59	FALSE
22	Megan Bowen	MeganB@BAPpartners.onmicrosoft.com	Flow Plan 2 Trial	Flow Plan 2 Trial	2/28/2017 21:20	TRUE
23	Megan Bowen	MeganB@BAPpartners.onmicrosoft.com	PowerApps Trial	PowerApps Plan 2 Free Trial	6/12/2017 20:59	TRUE
24	Megan Bowen	MeganB@BAPpartners.onmicrosoft.com	PowerApps Trial	Microsoft Power Apps & Flow	6/26/2017 13:22	TRUE
25	Diego Siciliani	DiegoS@BAPpartners.onmicrosoft.com	Flow for Office 365	Office 365 Enterprise E5	11/29/2016 19:10	FALSE
26	Diego Siciliani	DiegoS@BAPpartners.onmicrosoft.com	Flow Free	Flow Free	3/14/2017 18:36	TRUE
27	Diego Siciliani	DiegoS@BAPpartners.onmicrosoft.com	PowerApps for Office 365	Office 365 Enterprise E5	11/29/2016 19:10	FALSE

FAQs and more information

We have found some common questions on licensing and plan options. We've included several here with their answers. However, if you find you need more details, you can find that on PowerApps plans here <https://powerapps.microsoft.com/en-us/pricing/> and Microsoft Flow plans here <https://flow.microsoft.com/en-us/pricing/>. For additional information on the mechanics of managing users, please refer to this [link](#).

PowerApps Pricing FAQ

Which Microsoft Office 365 plans include PowerApps?

See the PowerApps Licensing overview page for the list of Office 365 plans that include PowerApps capabilities.

Which Microsoft Dynamics 365 apps and plans include PowerApps?

See the [PowerApps Licensing overview page](#) for the list of Dynamics 365 apps and plans that include PowerApps capabilities.

Can I connect to Microsoft Dynamics for Finance and Operations?

Yes, you can use the Dynamics 365 for Finance and Operations connector to build canvas apps using this data.

How long is the free trial period?

Free trials last 30 days.

Is there a plan for developers?

Yes, we have a free Community Plan to learn and build skills on PowerApps, Microsoft Flow and Common Data Service. [Learn more](#)

What happens when I use all the data storage, file storage, and flow runs included in my per user licenses?

You can buy additional data storage, file storage and flow runs. See the [PowerApps Licensing overview page](#) for more information.

Where can I find more information about pricing?

Find more detailed Q&A and answers in our [pricing documentation page](#).

Who can buy PowerApps Plan 1 or Plan 2?

Any customer can sign up for a free trial. Office 365 admins can buy PowerApps plans for their teams or organization. Contact your Office 365 admin when you're ready to buy.

Do all my users need to be licensed with the same PowerApps plan, or can I mix plans?

You can mix and match PowerApps licenses, and licenses that include PowerApps capabilities, across the users in your organization. For example, if there are 100 users in your organization, 50 may be licensed with Office 365, 20 with Dynamics plans, 25 with PowerApps Plan 1, and 5 with PowerApps Plan 2. Compare the features in each plan to choose the mix that meets your team's or organization's needs.

[Microsoft Flow Pricing FAQ](#)

Which Dynamics 365 plans includes Microsoft Flow?

These Dynamics 365 applications include 'Microsoft Flow for Dynamics 365' plan:

- Dynamics 365 Enterprise Sales
- Dynamics 365 Enterprise Field Service
- Dynamics 365 Enterprise Marketing
- Dynamics 365 Enterprise Customer Service
- Dynamics 365 Enterprise Project Service Automation
- Dynamics 365 Enterprise Operations
- Dynamics 365 Business Edition Financials

These Dynamics 365 plans include 'Microsoft Flow Plan 2':

- Dynamics 365 Enterprise, Plan 2
- Dynamics 365 Enterprise, Plan 1
- Dynamics 365 Business Edition Plan

Flow for Dynamics 365 is also included in existing CRM Online Enterprise, Professional, Basic, and Essential subscriptions.

[Compare plans](#)

How long is the free trial period?

Free trials are 90 days long.

Is there a way to develop my Microsoft Flow skills for more than 90 days?

Yes, with the PowerApps Community Plan you get a free environment for individual use with

functionality including the Common Data Service (CDS). In this environment you can explore and learn everything about Microsoft Flow and PowerApps for free, but the PowerApps Community Plan is not intended for production use.

[Learn more](#)

Which Office 365 plans includes Microsoft Flow?

These Office 365 plans include 'Microsoft Flow for Office 365' plan:

- Office 365 Business Essentials
- Office 365 Business Premium
- Office 365 Education
- Office 365 Education Plus
- Office 365 Enterprise E1 *
- Office 365 Enterprise E3 *
- Office 365 Enterprise E5

* Office 365 Enterprise E2 includes the same capabilities as Office 365 Enterprise E1, and Office 365 Enterprise E4 includes the same capabilities as Office 365 Enterprise E3.

Office 365 Enterprise F1 includes the same capabilities as Flow Free, but an SLA is available and the number of flow runs is aggregated across all users in the company.

[Compare plans](#)

Are the flow runs included in the per user licenses limited to the licensed user?

Flow runs included in Microsoft Flow Free can only be used by the licensed user.

Flow runs included in the Office 365, Dynamics 365, Microsoft Flow Plan 1 and Plan 2 are pooled across all users in the company.

What happens when I use all the flow runs included in my per user licenses?

You can buy more flow runs in increments of 50,000 flow runs per month.

Security

In this section of the paper we are going to look at how the PowerApps platform handles security from user authentication to authorization which allows users perform actions with data and services.

Conceptually, security in the platform is there to ensure users can do the work they need to do with the least amount of friction, while still protecting the data and services. Security in the platform can be implemented as a simple security model with broad access all the way to highly complex security models where users have specific record and field level access. The following is a high-level look at how a security model is implemented in PowerApps.

- Users are authenticated by Azure Active Directory (AAD)
- Licensing is the first control-gate to allowing access to PowerApps components

- Ability to create applications and flows is controlled by security roles in the context of environments
- A user's ability to see and use PowerApps is controlled by sharing the application with the user. Sharing of PowerApps canvas apps is done directly with the user or AAD group. Sharing of PowerApps model-drive apps is done via CDS security roles
- Environments act as security boundaries allowing different security needs to be implemented in each environment
- Flows and Canvas apps use connectors, the specific connections credentials and associated service entitlements determine permissions when apps use the connectors
- Environments with a Common Data Service for Apps (CDS) instance add support for more advanced security models that are specific to controlling access to data and services in that CDS instance.

Controlling access to PowerApps

PowerApps relies on Azure Active Directory (AAD) for authentication. This means that you can leverage the full functionality of AAD to manage and restrict access to users. This includes using Conditional Access Policies and other premium features of AAD. Developers can also register applications with AAD and use the OAuth2 authorization framework to allow their code to access the platform APIs.

External users from other AAD tenants can be added as Business Guests in your AAD. They can be configured to work with some limitations with PowerApps model-driven apps. Business Guests are not supported currently for PowerApps canvas apps and Microsoft Flow. Other external users beyond the capability of Business Guests, including Azure B2C is not currently supported.

Common Data Service

One of the key features of the Common Data Service is its rich security model that can adapt to many business usage scenarios. This security model is only in play when there is a CDS database in the environment. As an administrator you likely will not be building the entire security model yourself, but will often be involved in the process of managing users and making sure they have the proper configuration as well as troubleshooting security access related issues.

Role based security

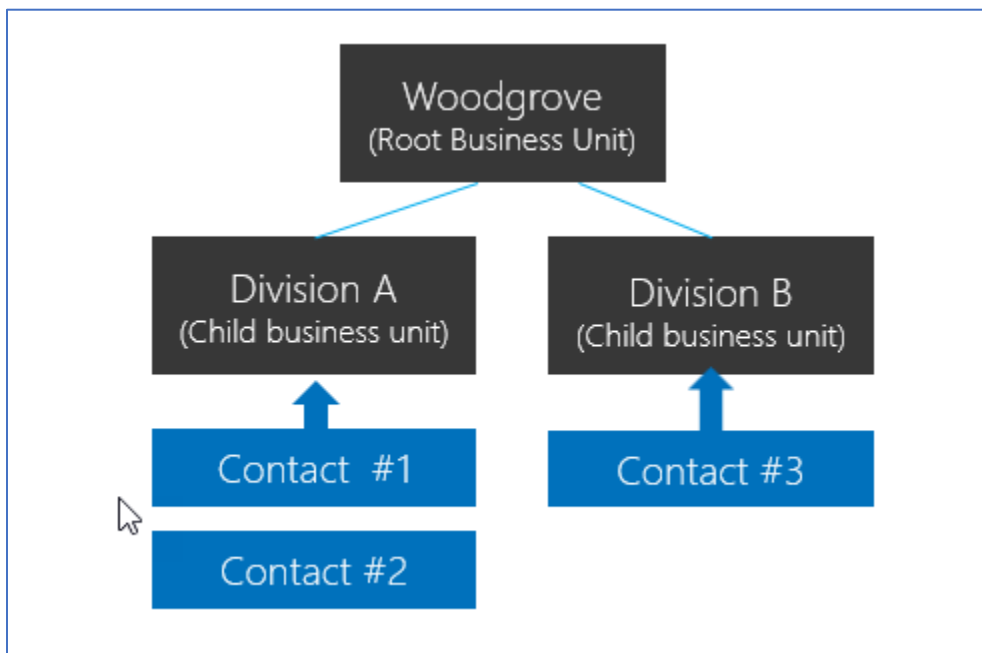
CDS uses role-based security to group together a collection of privileges. These security roles can be associated directly to users, or they can be associated with CDS teams and business units. Users can then be associated with the team, and therefore all users associated with the team will benefit from the role. A key concept of CDS security to understand is all privilege grants are accumulative with the greatest amount of access prevailing. Simply put, if you gave broad organization level read access to all contact records, you can't go back and hide a single record.

Business Units

Business units work in conjunction with security roles to determine the effective security that a user has. Business units are a security modeling building block that helps in managing users and the data they can access. Business units define a security boundary. Every CDS database has a single root business unit.

You can create child business units to help further segment your users and data. Every user assigned to a CDS instance will belong to a business unit. While business units could be used to model 1:1 a true organization hierarchy, more often they lean more towards just defined security boundaries to help achieve the security model needs.

To better understand let's look at the following example. We have three business units. Woodgrove is the root business unit and will always be at the top, that is unchangeable. We have created two other child business units A and B. Users in these business units have very different access needs. When we associate a user with this CDS instance, we can set the user to be in one of these three business units. Where the user is associated will determine which business unit owns the records that user is the owner of. By having that association allows us to tailor a security role to allow the user to see all records in that business unit.



Entity/Record Ownership

CDS supports two types of record ownership. Organization owned, and User or Team owned. This is a choice that happens at the time the entity is created and can't be changed. For security purposes, records that are organization owned, the only access level choices is either the user can perform the operation or can't. For user and team owned records, the access level choice for most privileges are tiered Organization, Business Unit, Business Unit and Child Business Unit or only the user's own records. That means for read privilege on contact, I could set user owned, and the user would only see their own records.

To give another example, let's say User A is associated with Division A, and we give them Business Unit level Read access on Contact. They would be able to see Contact #1 and #2 but not Contact #3.

When you configure or edit security role privileges you are setting the access level for each option. The following is an example of the Security Role privilege editor.

Entity	Create	Read	Write	Delete	Append	Append To	Assign	Share
Account	●	●	●	●	●	●	●	●
ACIViewMapper	●	●	●	●				
Action Card	●	●	●	●	●	●	●	
Action Card User Settings	●	●	●	●				●
Activity	●	●	●	●	●	●	●	●
Announcement	●	●	●	●		●		
Application File	●	●	●	●				
Azure Service Connection	●	●	●	●	●	●		
Connection	●	●	●	●	●	●	●	●
Connection Role	●	●	●	●	●	●		
Contact	●	●	●	●	●	●	●	●
Customer Relationship	●	●	●	●	●	●	●	●

In the above you can see the standard privilege types for each entity Create, Read, Write, Delete, Append, Append To, Assign and Share. You can edit each of these individually. The visual display of each will match the key below as to what level of access you have granted.

Key			
○ None Selected	● User	● Business Unit	● Parent: Child Business Units
			● Organization

In the above example, we have given organization level access to Contact which means that the user in Division A could see and update contacts owned by anyone. In fact, one of the most common administrative mistakes is getting frustrated with permissions and just over granting access. Very quickly a well-crafted security model starts looking like swiss cheese (full of holes!).

Teams

Teams are another important security building block. Teams are owned by a Business Unit. Every Business Unit has one default team that is automatically created when the Business Unit is created. The default team members are managed by CDS and always contain all users associated with that Business Unit. You can't manually add or remove members from the default team, they are dynamically adjusted by the system as new users are associated/disassociated with business units. There are two types of teams, owning teams and access teams. Owning Teams can own records, which gives any team member direct access to that record. Users can be members of multiple teams. This will allow it to be a powerful way of granting permissions to users in a broad way without micromanaging access at the individual user level. Access teams are discussed below as part of Record Sharing.

Record Sharing

Individual records can be shared on a one by one basis with another user. This is a powerful way of handling exceptions that don't fall into the record ownership or member of a business unit access model. It should be an exception though because it is a less performant way of controlling access. Sharing tougher to troubleshoot because it is not a consistently implemented access control. Sharing can be done at both the user and team level. Sharing with a team is a more efficient way of sharing. A more advanced concept of sharing is with Access Teams which provides auto creation of a team and sharing of record access with the team based on an Access Team Template (template of permissions) which is applied. Access teams can also be used without the templates, with just manual add/remove of it's

members. Access teams are more performant because they don't allow owning records by the team or having security roles assigned to the team. Users get access because the record is shared with the team and the user is a member.

Record level security in CDS

You might be wondering – what determines access to a record? That sounds like a simple question but for any given user it is the combination of all their security roles, the business unit they are associated with, the teams they are members of and the records that are shared with them. The key thing to remember is all access is accumulative across all those concepts in the scope of a CDS database instance. These entitlements are only granted within a single database and are individual tracked in each CDS database. This all of course requires they have an appropriate license to access CDS.

Field Level Security in CDS

Sometimes record level control of access is not adequate for some business scenarios. CDS has a field level security feature to allow more granular control of security at the field level. Field level security can be enabled on all custom fields and most system fields. Most system fields that include personal identifiable information (PII) are capable of being individually secured. Each field's metadata defines if that is an available option for the system field.

Field level security is enabled on a field by field basis. Access is then managed by creating a Field Security Profile. The profile contains all fields that have field level security enabled and the access granted by that specific profile. Each field can be controlled within the profile for Create, Update and Read access. Field Security Profiles are then associated with a user or Teams to grant those privileges to the users to the records they already have access to. It's important to note that Field Level Security has nothing to do with Record Level security, a user must already have access to the record for the Field Security Profile to grant them any access to the fields. Field level security should be used as needed and not excessively as it can add overhead that is detrimental if over used.

Managing Security Across Multiple Environments

Security roles and Field Security Profiles can be packaged up and moved from one environment to the next using CDS solutions. Business Units and Teams must be created and managed in each CDS environment along with the assignment of users to the necessary security components.

Configuring Users Environment Security

Once roles, teams and business units are created in an environment it is time to assign the users their security configurations. First, when you create a user you will associate the user with a business unit. By default, this is the root business unit in the organization. They are also added to the default team of that business unit.

In addition, you would assign any security roles that user needs. You would also add them as members of any teams. Remember teams can also have security roles, so the effective rights of the user is the combination of directly assigned security roles combined with those of any teams they are members of. Security is always additive offering the least restrictive permission of any of their entitlements. The following is a good walkthrough of configuring environment security <https://docs.microsoft.com/en-us/powerapps/administrator/database-security>

If you have used Field Level security, you would need to associate the user or a team of the user to one of the Field Security Profiles you created.

Security is a complex topic and is best accomplished as a joint effort between the application makers and the team administering the users permissions. Any major changes should be coordinated well in advance of deploying the changes into the environment.

Data Loss Prevention Policies

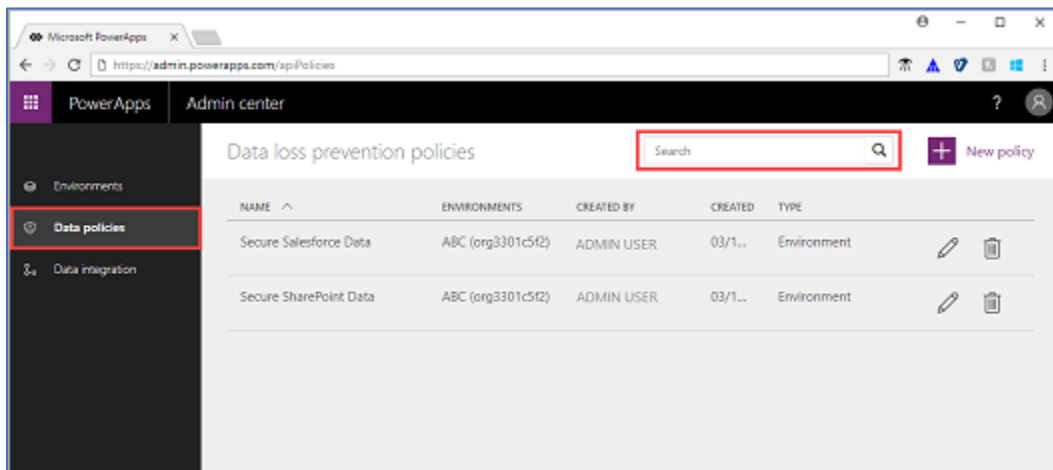
Your organization's data is likely one of the most important assets you are responsible for safeguarding as an administrator. The ability to build apps and automation that uses the data allows your company to be successful. PowerApps and Microsoft Flow allow rapid build and rollout of these high value applications that allow users to measure and act on the data in real time. Applications and automation are increasingly becoming more connected across multiple data sources and multiple services. Some of these services might be external 3rd party services and might even include some social networks. Users will often have good intentions but might overlook the potential for exposure from data leakage to services and audiences that shouldn't have access to the data.

Data Loss Prevention (DLP) policies that help protect organizational data from unintended exposure, are available for administrators to create. They can act as guardrails to help prevent users from unintentionally exposing the data. DLP policies can be scoped at the environment and tenant level offering flexibility to craft policies that are sensible and do not block high productivity.

DLP policies enforce rules of what connectors can be used together by classifying connectors as either Business Data only or No Business Data allowed. Simply, if you put a connector in the business data only group, it can only be used with other connectors from that group in the same app. Keep reading and we will cover some scenarios for using this later in this section.

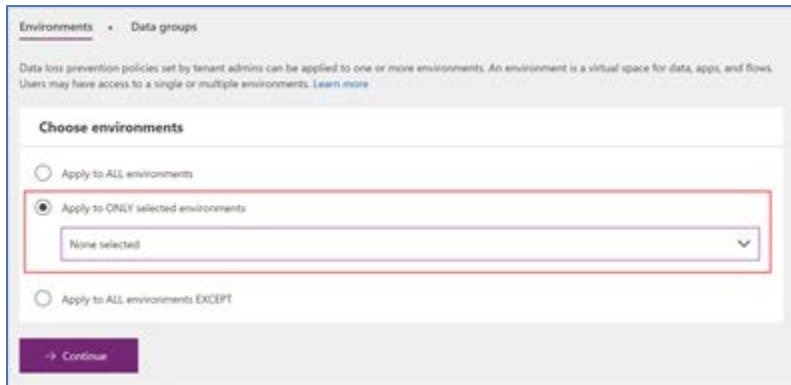
What policies do we already have?

From the PowerApps Admin Center (admin.powerapps.com) you can see the current policies you have in place in your tenant. This should be your first stop as a new administrator to understand what is currently active.



Creating new DLP Policies

When you create a new DLP policy you first decide on the scope. If you are only an environment administrator, you will see a selection to choose one of your environments to associate with the DLP policy. If you are a tenant administrator you will have the ability to apply to All Environments, Selected Environments or All Environments EXCEPT.

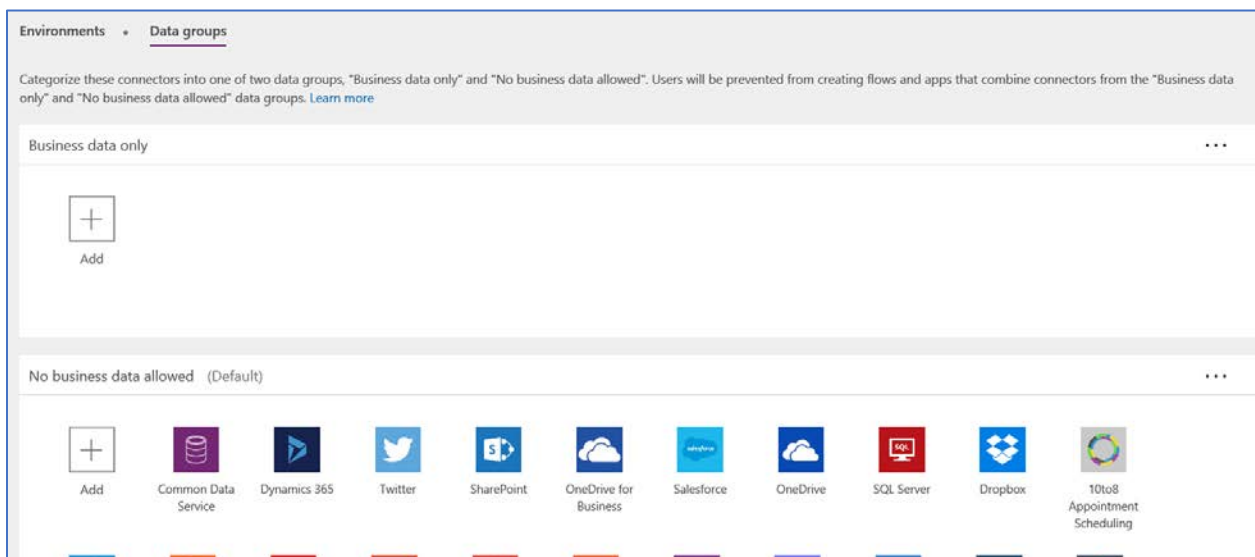


Environment only admins do have the ability to view policies created by tenant admins to understand what might apply to their environment.

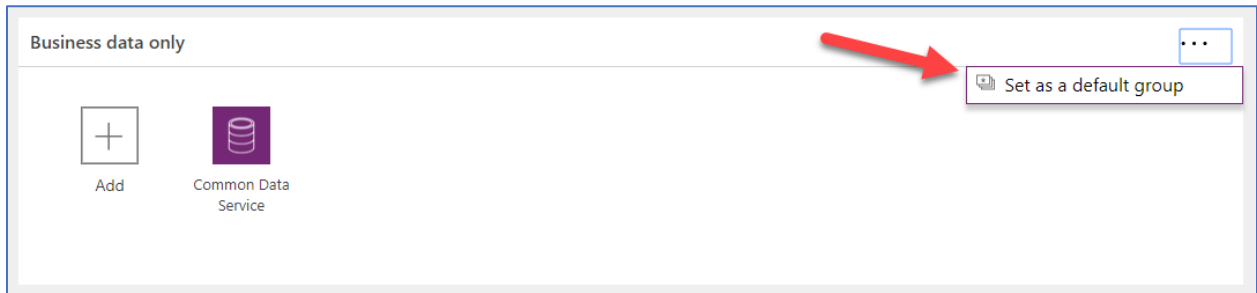
One thing to consider is that environment specific policies can't override tenant-wide DLP policies. For example, if you only allow use of CDS connectors in an environment, an individual user that is only an environmental admin can't override that policy to allow social network connectors to be used.

Configuring connectors for a DLP policy

By default, all connectors are considered part of the No business data allowed list and no connectors are included in the business data only group. This effectively means that all connectors can be used with other connectors.



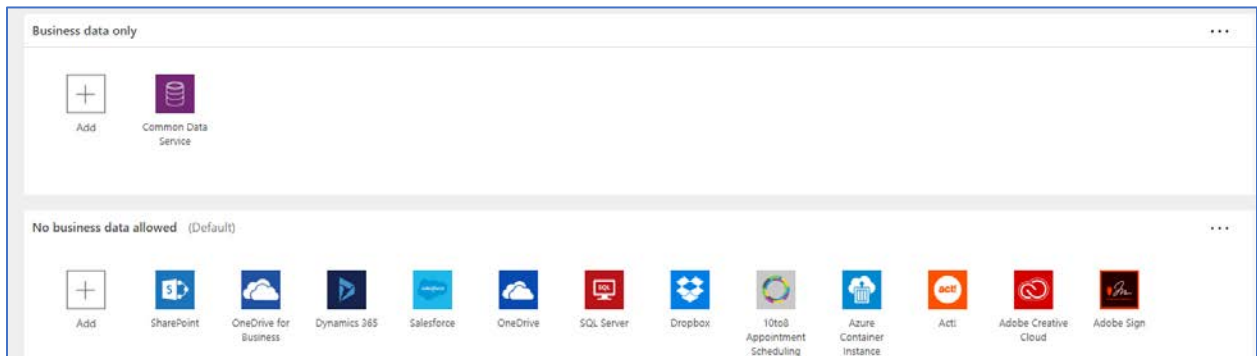
When new connectors are added they are added to the Default category which is No business data allowed. If you would prefer you can change which category is considered the default, and then all new connectors will be classified in that category by default.



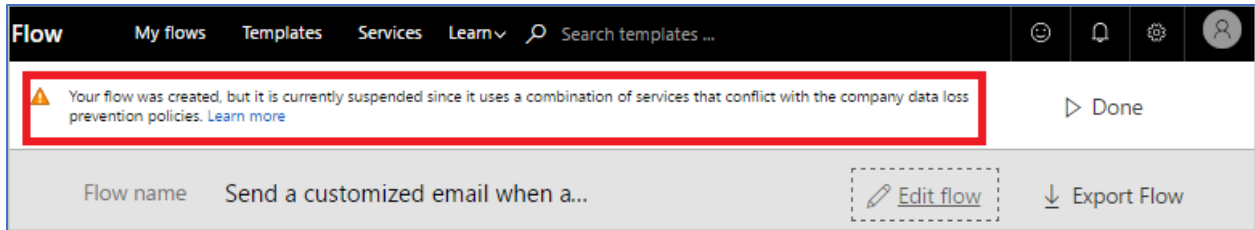
Typically, though most companies will want to treat new connectors as No business data allowed until they evaluate if it is appropriate to use with what they have classified as business data.

Let's look at an example if we were to create a new tenant wide DLP policy that had just the Common Data Service added to the Business Only Data and all others in No Business Data. Let's look at a few application examples and the outcome of this policy.

Connectors used in Application or flow	Impact of DLP
SharePoint and OneDrive	This would be allowed
Common Data Service	This would be allowed
Common Data Service and SharePoint	This would not be allowed
SharePoint and Twitter	This would be allowed
SharePoint, Twitter and Common Data Service	This would not be allowed



Users accessing a PowerApp or Flow impacted by the DLP policy will see a message informing of the DLP policy conflict. As an administrator you should have a process and plan in place to handle these types of support needs if you are using DLP policies.



One thing to keep in mind, DLP policies created for a connector do not understand that that connector could be configured to talk to Dev, Test and Production, etc. When you configure a DLP policy it is all or nothing. So, if you want to allow Dynamics 365 connector to talk to a test database in the test environment, but not allow it to connect to the production database in that same test environment, then DLP policies won't help you restrict that. Another way to say the same thing, is DLP policies are Connector aware, but do not control the connections that are made using the connector.

Strategies for creating DLP policies

As an administrator taking over an environment or starting to support use of PowerApps and Microsoft Flow DLP policies should be something you evaluate and create within the first 30 days. This ensures a base set of policies are in place before too many users start creating connections that might violate your policies.

For smaller environments where the users are highly capable and are trusted you could start out with no DLP policies taking only the default options. This is the most flexible option and can be changed at any time. Keep in mind introducing more restrictive policies later could conflict with existing assets. These conflicts could have business impact when existing apps and flows stop working until either the app / flow is brought into compliance or the DLP policy relaxed.

For larger environments it is recommend you have a plan in place for DLP policies. It is best to do this in conjunction with your plan for managing environments in your organization. While there is an endless combination of connectors you might have in your own environment we will be using an example that you can tailor to fit your own needs. Let's setup a framework for a generic DLP policy template that could apply to many organizations, only modifying it for some of their specific needs.

First, let's look at our environment setup and assumptions. The following are the environments we are expecting to manage in our organization.

Environment	Expected Use / Policy
Contoso – Default	This is the default environment, and anyone can create apps and flows in it.
Contoso Enterprise Apps	This is a Production environment with applications managed with formal review before being promoted here. This could also be more business unit aligned e.g. Marketing, Finance etc.
Community Plan Environments (0...N)	These will be automatically created by any users in our org that sign up for the free Community Plan

User Owned Environments (0...N)	These are Production or Trial Environments created by users with full P2 licenses or with Trial P2 licenses
---------------------------------	-------------------------------------------------------------------------------------------------------------

We now are going to design a tenant wide default DLP policy. Our goal is to ensure that as people create their own environments and test and explore they minimize mix of core business data without us first working with them.

Our goal is to apply this default global policy to all environments except Contoso Enterprise Applications which we are going to manage by a separate DLP policy.

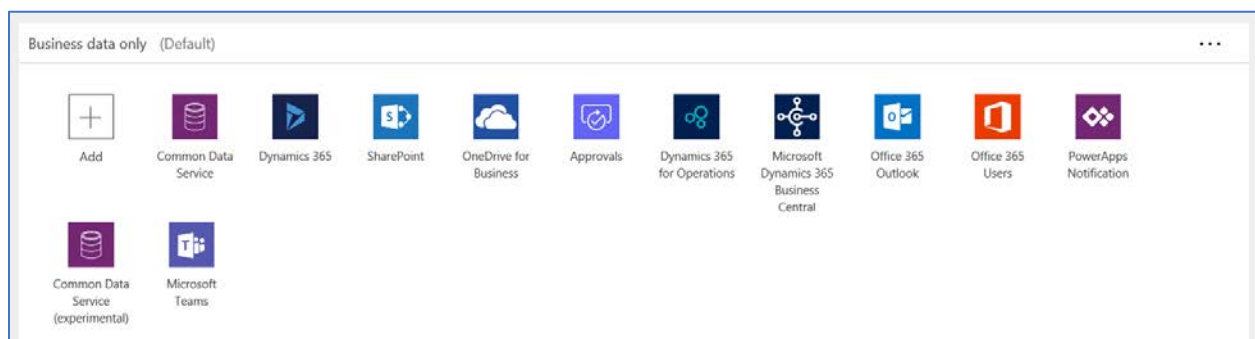
Choose environments

Apply to ALL environments
 Apply to ONLY selected environments
 Apply to ALL environments EXCEPT

Contoso Enterprise Applications

→ Continue

We have identified the following connectors as our initial set of business only data allowed connectors (remember you can always add to this list at any time!).



With this policy in place any use outside of those business connectors will need to have exceptions handled and we will cover that shortly.

For Contoso Enterprise Application environment since we excluded it from our policy we have two choices. We can either leave it wide open since we only deploy to it trusted applications that we as administrators install and configure or we establish a DLP policy for it to match its application needs. The following new DLP policy shows how we would create a DLP specific for that environment.

Choose environments

Apply to ALL environments

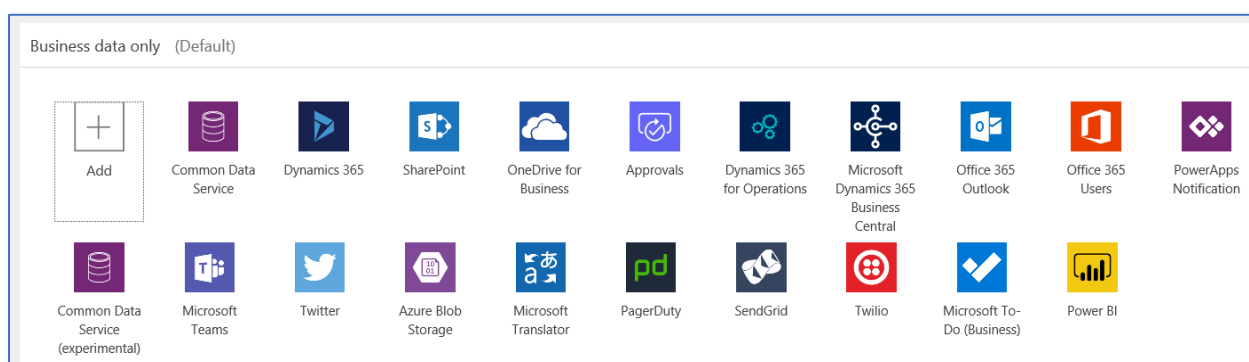
Apply to ONLY selected environments

Contoso Enterprise Applications ▼

Apply to ALL environments EXCEPT

→ Continue

The following is an example that might look like a super set of our global one – notice it includes some social network and 3rd party connectors – but since these are all trusted apps and flows that is ok.



Now with this in place, you need a plan on how to handle exceptions. You really have three choices

1. Deny the request
2. Add the connector to the default DLP policy
3. Add the users' environments to the All Except list for the Global default DLP and create a user specific DLP policy with the exception included.

Hopefully that helps you understand how you might apply DLP policies in your organization. These are just some of the many options you could configure with DLP policies.

Management and Monitoring

In this section we will focus on the tools you can use to manage and monitor what is going on in your environments. Tooling falls into the following three categories:

Portals offer an interactive experience for performing administrative tasks. This is typically considered the primary path for completing administrative activities. From a monitoring point of view this channel is used mostly for ad hoc interactive discovery.

PowerShell cmdlets offer a way to automate both management and monitoring tasks using PowerShell. These cmdlets can be used in a sequence to automate multi-step administrative actions.

Connectors offer the ability to use the platform’s own tools to manage and monitor itself. The Flow Management connector is specifically designed to help with administrative management and monitoring. But you can also use any of the other 200+ connectors and approval process capabilities to automate your own admin work.

Working with the Admin Portals

In a perfect world as an administrator you would only visit a single portal to perform all your administrative tasks but given the scope and breadth of the different products involved and their differing release cycles, there are multiple portals with which you will interact. The following outlines the different portals and the most common tasks you perform there.

Portal	Common Tasks
Power platform admin center https://admin.powerplatform.microsoft.com	The new unified administrative portal for Power platform admins. Currently this portal can be used for CDS Instance Management, to submit Dynamics 365 & flow focused support tickets, and to view PowerApps and Flow admin analytics. Over time the following admin experience will be migrated & replaced by the Power platform admin center : <ol style="list-style-type: none"> 1. PowerApps Admin Portal 2. Microsoft Flow Admin Portal 3. Business platform admin center 4. Dynamics 365 admin center
PowerApps Admin Portal https://admin.powerapps.com	Creating and managing environments including security starts here. Within each environment you can manage the apps and flows. Monitoring to see who is licensed and building things. Managing Data Loss Prevention Policies. Manage CDS Data Integration projects. Over time this will be migrated & replaced by the Power platform admin center .
PowerApps Maker Portal https://web.powerapps.com	This portal is focused on building PowerApps but can also view and manage CDS components, manage connectors and gateways. You can also see application statistics from details on apps here.
Microsoft Flow Admin Portal https://admin.flow.microsoft.com	This points to the same site as admin.powerapps.com. Over time this will be migrated & replaced by the Power platform admin center .
Business platform admin center https://admin.businessplatform.microsoft.com	This points to the same site as admin.powerapps.com. Over time this will be migrated & replaced by the Power platform admin center .
Dynamics 365 admin center https://port.crm.dynamics.com/G/manage/index.aspx	The Dynamics 365 Admin Center, that can be leveraged to perform certain CDS Environment management activities like renaming, deleting, and resetting.
Dynamics 365 Instance Management <a href="https://port.crm<N>.dynamics.com/G/Instances/InstancePicker.aspx">https://port.crm<N>.dynamics.com/G/Instances/InstancePicker.aspx	This instance management portal is reached from admin.powerapps.com when managing the CDS database or from the Dynamics 365 admin center. Here you will see a list of all the CDS databases and can perform actions such as backup, as well as other actions on a per instance basis.

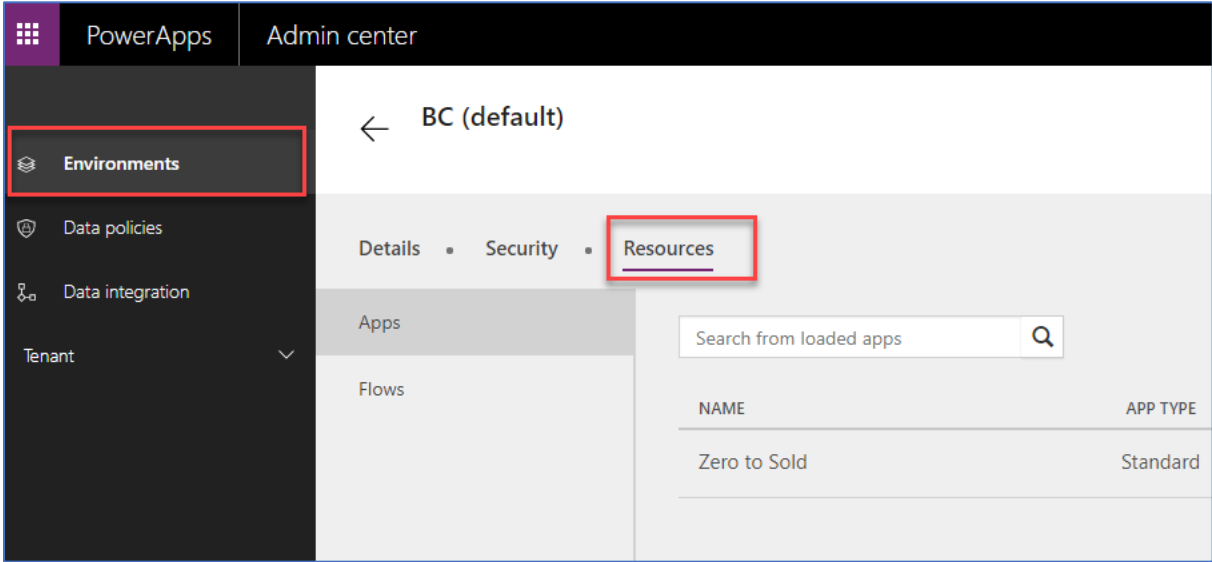
Microsoft 365 admin center https://portal.office.com/AdminPortal	Here you will manage users and their license assignment as well as you can launch into many of the individual admin centers from here.
Microsoft Azure https://portal.azure.com	Advanced Azure AD management tasks like conditional access is managed here. Also if you support any developer application registration it is also done here. This is also where you start setup of your on-premises gateways.
Security & Compliance Center https://protection.office.com	In addition to the general compliance tasks, administrators can come here to search the Audit log to see Flow audit events

Over the near-term future we will see consolidation of the PowerApps, Flow and the Dynamics 365 administration portals into a more unified administrative portal experience.

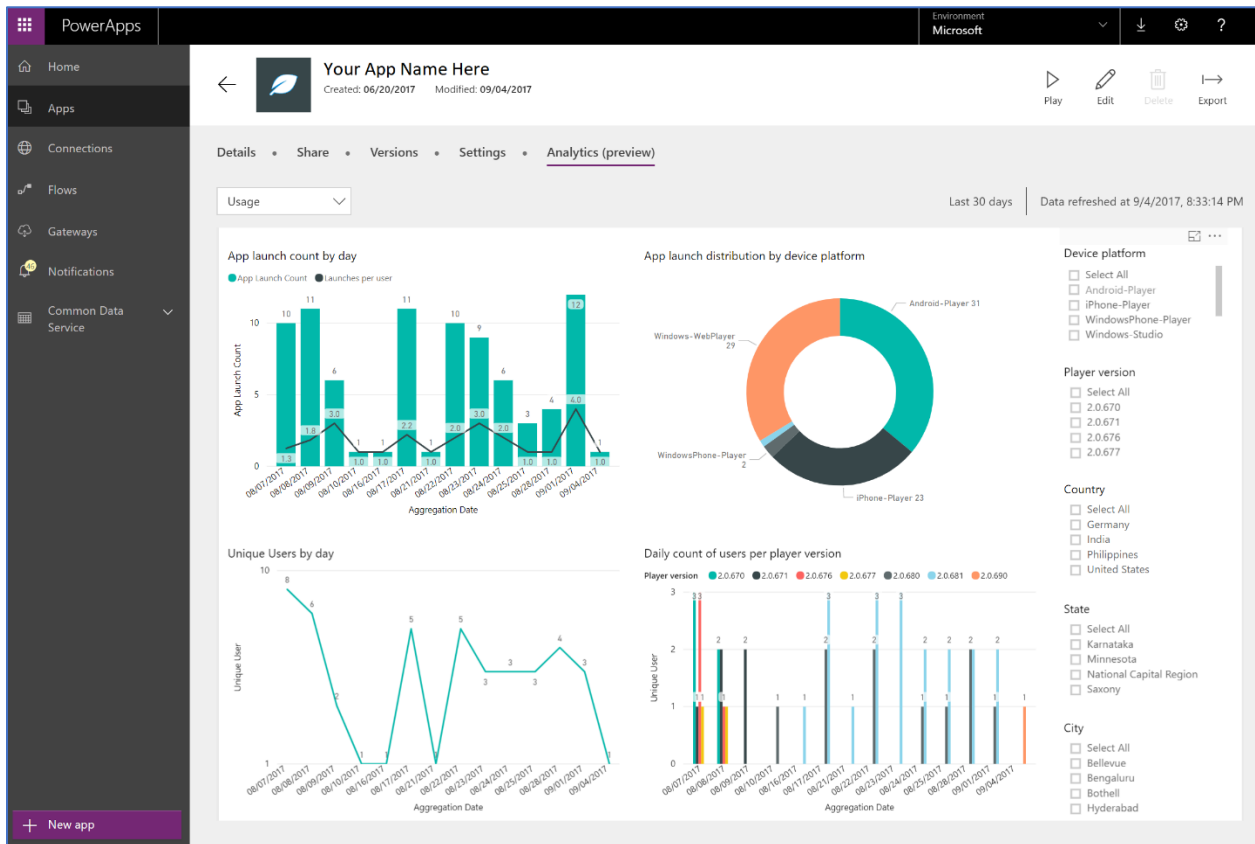
For partners helping their customers manage their cloud services using delegated administration capabilities you will not be able to use delegated access to the PowerApps and Flow portals. Currently, you would need to have a user in the customers tenant and assign that user a P2 license.

Common Portal Tasks

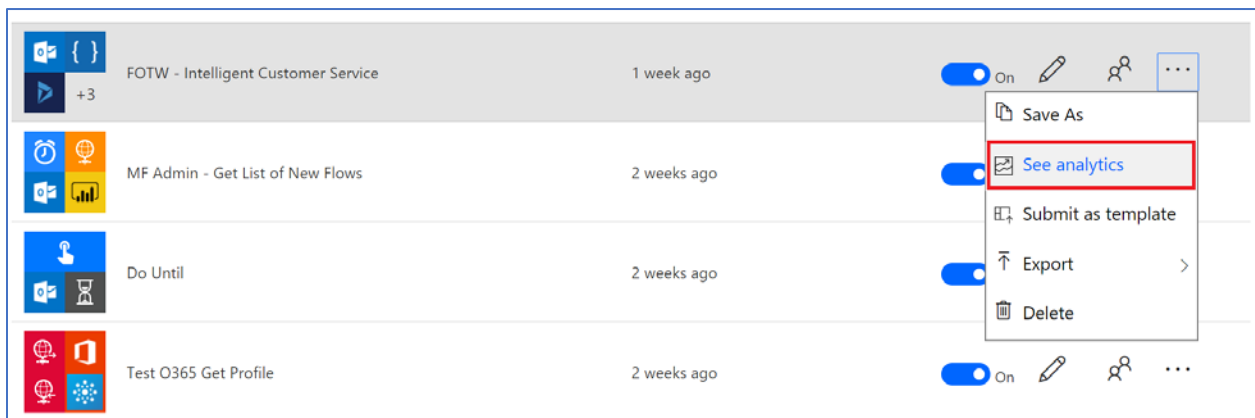
Managing Flows and Applications in an Environment



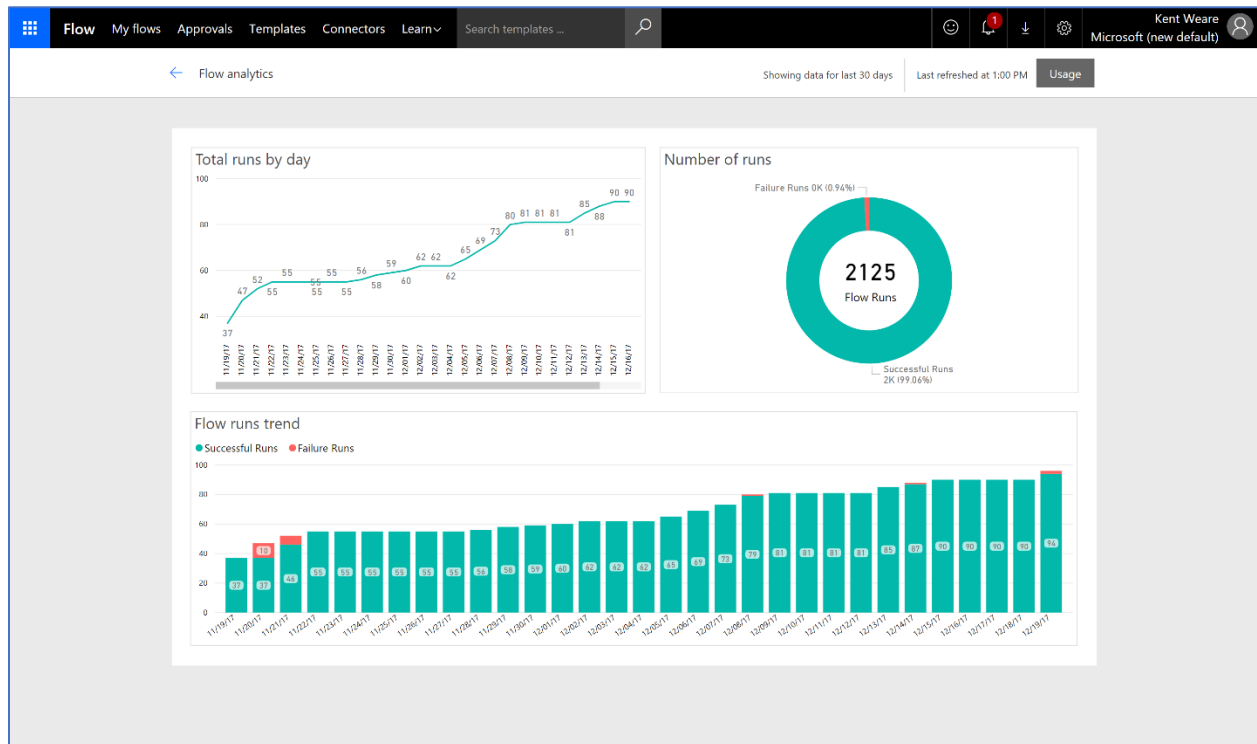
View Application Analytics



View Flow Analytics can be found from drilling down from the list of flows.



From here you will see the following details for that specific flow



Automation of tasks with PowerShell

The PowerShell cmdlets allow you to do similar tasks that you would do with the admin portals but do them in scripting where you can sequentially execute multiple commands or pipe output from one to automate common tasks. There are multiple PowerShell cmdlets that you can work with. The following is an overview of each that you would likely interact with.

PowerShell cmdlet library	Common Tasks
PowerApps cmdlets https://docs.microsoft.com/en-us/powerapps/administrator/powerapps-powershell	Designed for app makers and administrators to automate tasks with environments and associated apps, flows and connectors. Note: These cmdlets are currently in preview.
Office 365 cmdlets https://docs.microsoft.com/en-us/office365/enterprise/powershell/getting-started-with-office-365-powershell	These are focused on Office 365 related tasks and can be used to automate user-related actions and tasks, for example, assignment of licenses.
Dynamics 365 cmdlets https://docs.microsoft.com/en-us/powershell/dynamics365/customer-engagement/overview	These are useful if you have any environments with CDS for Apps databases. Modules include support for using the CDS online admin API, as well as to automate solution deployment to the CDS instances.

Microsoft Azure cmdlets
<https://docs.microsoft.com/en-us/powershell/azure/overview>

The Azure cmdlets are useful if you are including any Azure components in your overall solution. This could also be used to script setup of the on-premise application gateway.

Common PowerShell Tasks

Displaying a list of environments

Get-AdminPowerAppEnvironment

This will give you key information such as the Display Name and GUID of the environment. This is often what is needed for follow on operations.

Adding parameters such as -Default will allow you to generically find the default environment in the tenant

Get-AdminPowerAppEnvironment -Default

Using the GUID you got back (which is the non-display name for the environment) you can drill into details of that specific environment

Get-AdminPowerAppEnvironment -Environment 'EnvironmentName'

Which would produce the following detailed information:

```
PS C:\> Get-AdminPowerAppEnvironment -EnvironmentName fccc53da-0c81-4031-80ea-a93ea68e043a

EnvironmentName      : fccc53da-0c81-4031-80ea-a93ea68e043a
DisplayName           : CDS 2.0 Production (orgd0ed966d)
IsDefault            : False
Location             : unitedstates
CreatedTime          : 2018-07-09T04:52:34.4418252Z
CreatedBy            : @{id=2e804f76-5936-4cf1-bc7c-6faad3d34792; displayName=; type=User; tenantId=efec
                      userPrincipalName=}
LastModifiedTime     : 2018-07-09T04:52:34.4418252Z
LastModifiedBy      :
CreationType         : User
EnvironmentType      : Sandbox
CommonDataServiceDatabaseProvisioningState : Succeeded
CommonDataServiceDatabaseType : Common Data Service for Apps
Internal             : @{id=/providers/Microsoft.BusinessAppPlatform/scopes/admin/environments/fccc53da-0c81-4031-80ea-a93ea68e043a; location=unitedstates; properties=}
```

Another useful one is getting a list of connections in an environment. The following lists all the connections in the tenant's default environment

Get-AdminPowerAppEnvironment -Default | Get-AdminPowerAppConnection

And finally, a little more complex example. This one pipes the output from one cmdlet to others and presents a nice list of number apps in each environment in the tenant

```
Get-AdminPowerApp | select -ExpandProperty EnvironmentName | Group | %{ New-Object -TypeName PSObject -Property @{ DisplayName = (Get-AdminPowerAppEnvironment -EnvironmentName $_.Name | select -ExpandProperty displayName); Count = $_.Count } }
```

Which would produce the following detailed information:

```
PS C:\Users\jamesol\Source\Repos\Flow-PowerApps-PowerShell> Get-AdminPowerApp | select -ExpandProperty EnvironmentName | Group | % { New-Object -TypeName P
SObject -Property @{ DisplayName = (Get-AdminPowerAppEnvironment -EnvironmentName $_.Name | select -ExpandProperty displayName); Count = $_.Count } }

DisplayName      Count
-----
Ignite Demo Production    2
Jane Doe's Environment    2
Demo-Build              8
viral trial's Envir...    1
Fabrikam Inc. (defa...    22
Ignite Demo UAT          4
```

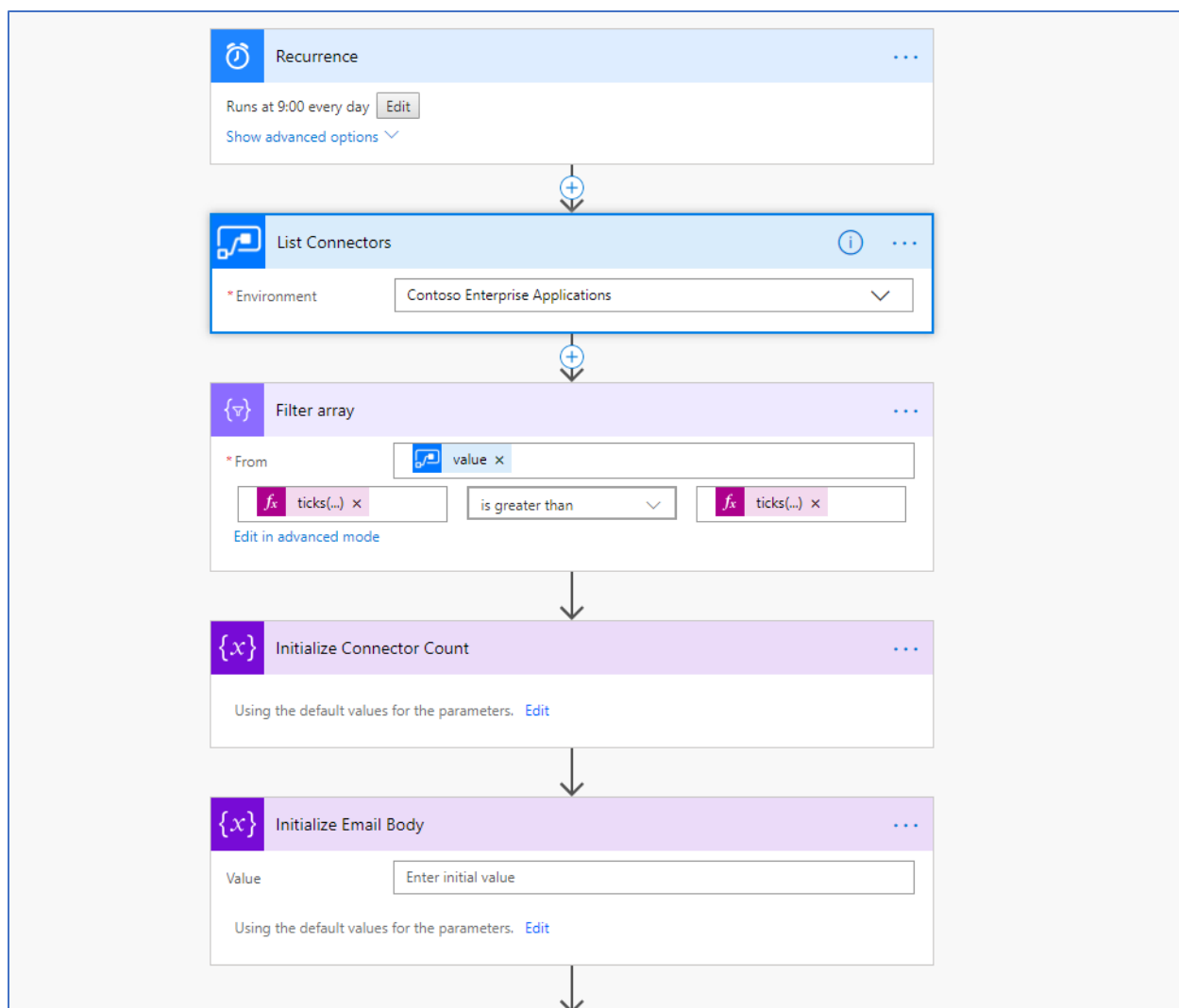
Automation of tasks with Microsoft Flow

One of the unique things about Microsoft Flow is you can use it to manage itself along with other parts of the Microsoft Power platform. The following connectors can be helpful to automate administrator tasks with Microsoft Flow.

Connector	Possible uses
Flow management connector https://docs.microsoft.com/en-us/connectors/flowmanagement/	Can be used to automate working with Flows including getting lists of new flows or connectors in your environments.
Office 365 Users connector https://docs.microsoft.com/en-us/connectors/office365users/	Useful for automating actions around users. For example, you could use the connector to get the manager of a user that owns an environment to be able to send them an e-mail for approval.
Approval connector https://docs.microsoft.com/en-us/connectors/approvals/	Often administrators need to get approvals and Flow offers a rich approval set of tasks you can automate this process.
Microsoft Forms https://docs.microsoft.com/en-us/connectors/microsoftforms/	Forms is an easy way to collect information to start an admin task. This can be combined with the Approval connector to get manager approval
Azure AD connector https://docs.microsoft.com/en-us/connectors/azuread/	Useful to perform tasks such as adding a user to a group or even creating the group.

Common Flow Tasks

List new connectors created is a simple template you can get started with right away. It simply triggers daily on schedule, and uses the Flow Management connector to get a list of the connection in the environment and sends you an e-mail. You can add it to your flows quickly using the template <https://us.flow.microsoft.com/en-us/galleries/public/templates/5a6ef26db3b749ed88b7afb377d11ecf/list-new-microsoft-flow-connectors/>



If you want to try building it yourself, there is a good walkthrough of creating the flow from scratch here <https://flow.microsoft.com/en-us/blog/new-flow-connector-notifications/>

Deployment Scenarios

Now that you have read through the platform architecture section and the data protection concepts, and have a good grasp of all the individual components, let's look at some scenarios and how you might handle deploying them. This assumes you created some default data loss prevention policies like what we suggested in the compliance and data protection section of the document. These scenarios represent possible deployment configurations but are not the only ways you could deploy the given scenario. Use them to inspire how you want to handle things in your organization.

Canvas app or Flows that are built to share with others

In this scenario a user built a flow in the default environment that uses only connectors that are allowed by your DLP policies.

For this scenario there is no need for additional DLP policies or environments. The user can share the flow themselves, with other users either as co-owners if they want them to be able to edit it, or for run-only.

Canvas app or Flows with connectors violating existing DLP

A user started building a PowerApp, canvas app or a Flow and after adding two connectors, was informed that one of the connectors violated the DLP policies. They approached you for how they could get an exception.

For this scenario you have three primary options; deny the request, add the connector to your existing tenant wide policy or create environment(s) to support the exception. If you decide to update your existing tenant-wide DLP do so understanding it would apply to all environments and all PowerApps and Flows; there are not exceptions to that policy.

If you decide to allow an exception in a special environment, this could be a shared environment that is used by any users you give an exception to or it could be a separate environment for the user or team needing the exception.

Canvas app or Flows with existing CDS database

A user or team wants to build an application that leverages data that already exists in CDS. They do not plan to make any schema changes to CDS.

For this scenario the CDS database would exist in an environment other than default (since you can't currently create a CDS instance in default). The canvas apps or flows can't therefore be built in the default environment using the CDS connectors but could if they use the Dynamics 365 connector which allows you to select the CDS instance from a separate environment.

The next decision comes down to if there is need for test data. If there is, then building the app in the test environment with the CDS connector would allow the app to be promoted to the production CDS environment once development and testing was completed. Since the app used the CDS connector it would be able to be simply exported and re-imported into the production environment without having to change the references to test. This assumes that test and production CDS environments have the same schema.

Canvas or model-driven apps and/or Flow with CDS – Multiple Teams

Multiple teams in your organization want to build applications with each having either a PowerApp model-driven app component or some of their own CDS schema customizations. In this scenario some teams' applications might want to leverage some of the data from other teams' applications. The goal is to have a centralized CDS that all these teams interact with, and not a silo of data for each teams' applications.

For this scenario you could have one main CDS production environment that contains all the applications once they are deployed for use by the broad set of organization users. Each team that is building an application would have their own CDS environment. Each team would release updates to their application in the form of managed CDS solutions. These managed CDS solutions would be imported into the CDS production environment. If there were test or staging or UAT environments that

would happen prior to the import to production to support testing. But it would be the same managed solution imported into each that was exported from the development instance.

If a team depended on other teams' schema or other CDS assets, they would import that dependent team's managed solution into their CDS development environment. That would of course make their solution dependent on the other team's application.

By having each team do their development work in their own environment allows each application to develop independently of the other applications in your organizations. While at the same time keeping a centralized data repository that all apps could interact across the enterprise data.

Some governance is needed in this type of environment to ensure applications coming into the shared environment do not make conflicting design decisions. For a simple example, some of the shared common entities like Account or Contact, you wouldn't want individual applications trying to rename those entities differently.

With this setup, the CDS environment could also contain Dynamics 365 applications co-existing with your internally built applications.

Application Lifecycle Management

Application Lifecycle Management (ALM) is important as the applications your organization builds becomes more complex and as more of your company depends on their stability. In other parts of the paper we discussed some of the ALM building blocks that just happen such as versioning of PowerApps canvas apps. We also covered some of the self-service actions that makers can do such as exporting and importing their CDS solutions. In this section we are going to have a more cohesive discussion about ALM bringing together some of these individual concepts and using them to handle more complex scenarios.

Let's look first at things you should consider as an administrator to consider to help guide the application through its lifecycles from new to production and then ongoing maintenance and enhancements. For purposes of this section, application refers to the whole set of components form PowerApps canvas or model-driven apps, flows and any CDS customizations.

New Applications	Existing Applications being upgraded
Who is the application owner, and who is involved in maintaining it?	Are any new connectors being used by the application?
Who are the users of the apps? Are they already licensed?	Is there any new reference data to update?
What environment did you build the app in?	Are there any new Canvas, Flows or CDS solutions added in this update?
Are there any PowerApps canvas or model-driven apps as part of the application?	Any changes to how users are assigned security roles?
Are there any flows?	Any impact on existing CDS data?
What connectors are the apps using?	Any changes in the required licenses?

Does anything require an on-premises gateway?	Potentially any of the considerations from the New Application column, if it was not a consideration at the time.
Does the application use CDS entities?	
Is the application dependent on any other existing applications or external services?	
Are there different security roles for different types of users?	
Is there any existing data that must be migrated into the new production system?	
Does the application have reference data that needs to be in the production environment?	
Who will be testing the application? Will it be in a separate environment?	
How will users report problems or enhancements?	
How frequently do you plan to do updates?	

The answers to these questions will help you put together an application profile and decide how best to support the team with deploying the application. This is not an exhaustive list, but a starting point for you to develop your own set of questions for applications.

Getting ready for a new application

Armed with the above information, consider each of the following as you get ready to deploy the new application:

- Licensing – acquire licenses and assign them for users
- Azure AD Group – consider if having a group that had all the app users would help with sharing the applications with them (good for canvas apps)
- Environments – if necessary create the new environments, considering how the application will be tested prior to production deployment
- Data Loss Prevention policies – do current ones support the app? Are new ones needed?
- Automation – is there any automation that would help with ongoing app administration?

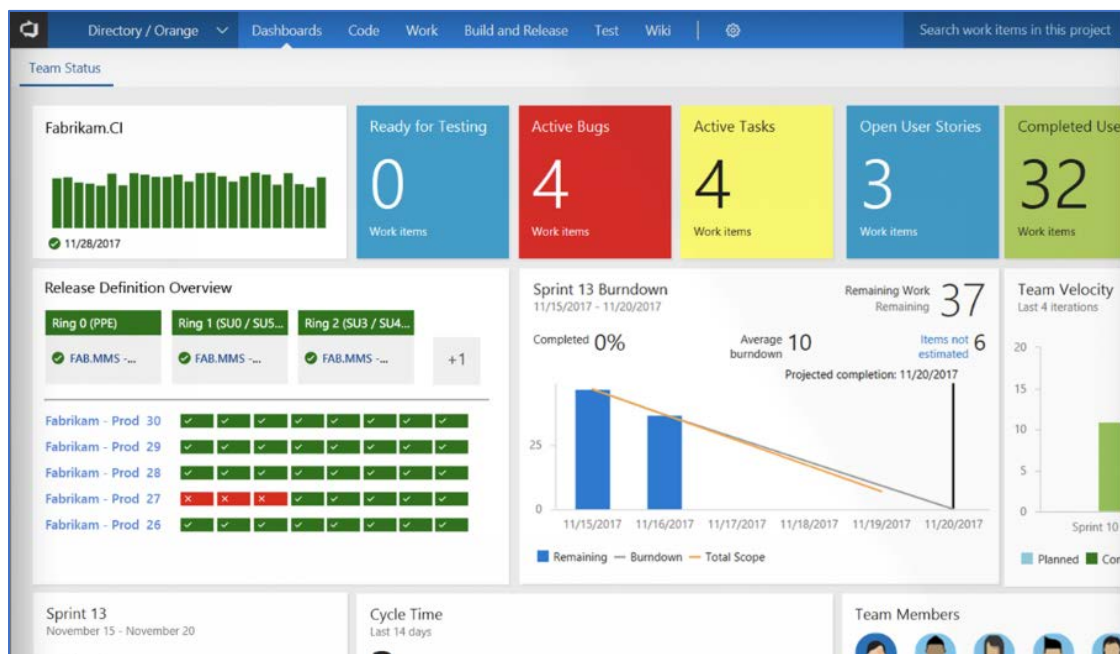
Tools to help Manage, Plan, Track, and Deploy

Depending on the complexity of the application, anything from using a SharePoint List to track work to be done and new features, and a OneDrive to store exported assets to a more complete solution like Visual Studio Team Services can help add some structure to your application life cycle process. What is appropriate for your organization depends on the size and maturity of the team that is building the overall application. The less technical will probably find a solution like OneDrive and SharePoint more approachable. Visual Studio Team Services (VSTS) has several features that are tailored to support

application lifecycle management. VSTS is also free to get started <https://visualstudio.microsoft.com/team-services/>. The following are some of those features:

- Work item planning and tracking
- Version control – offers a way to store exported assets – using Dynamics 365 SDK tools like Solution Packager allows this to scale up to larger teams working on CDS Solution package customizations
- Build and release automation – This can be helpful for automating everything from exporting of CDS solutions for backup, to compiling developer-built components. The release automation can take solutions and developer assets and coordinate deploying to test and production environments. These deployments can also leverage approval checkpoints as appropriate. Using community tools like Xrm.CI.Framework <https://marketplace.visualstudio.com/items?itemName=WaelHamze.xrm-ci-framework-build-tasks> you can deploy CDS solution packages from the release tasks.

The following is an example of the Team Status Dashboards that gives the team an all up view of their progress.



Exporting from the source environment

We've already covered the concept of exporting from PowerApps, Flow and CDS earlier in the document. Let's look at some additional things to consider when exporting as part of an application lifecycle management process.

- Always save a copy of the exported PowerApp, Microsoft Flow or CDS solution file.
- For CDS Solutions make sure if you are publishing a managed solution, that you also export an unmanaged solution as well. If you are not familiar with the differences, we cover that in the

Platform Architecture section.

- For CDS solution export you should always perform a publish on the solution or publish all for all solutions prior to export to ensure all changes are exported as expected.
- For Flows and canvas apps review the connectors that are used. Any custom connectors will need to be re-created prior to import in the target environment.

Importing into the target environment

We also covered import, but let's look at a few more things to consider.

- Always evaluate what is already in the target environment.
- Create any necessary custom connectors prior to import
- If you are importing a CDS solution that is dependent on other CDS solutions make sure those are already imported into the CDS instance
- If you import an unmanaged CDS solution make sure you publish all after import has completed
- Remember when you import an update to a PowerApps canvas application you must publish the new version before others will see it
- If you are importing CDS changes that remove any entities and data, consider a proactive on demand backup prior to the import.

Updating existing applications

Shown earlier, the import feature allows the maker to update an existing app in the target environment. Here are some considerations.

- Custom connectors updates must be performed first, as your app may rely on new data definitions.
- Custom connector updates may take a few minutes to be reflected in the portal. During that time, new operations may return a 404 error when invoked.
- If extensive changes are being made, consider creating a new custom connector and leaving the old connector intact. This can also be beneficial in the event the maker needs to roll back, as the previous version of the app will use the old (existing) connector.
- PowerApps uses caching for the web and mobile clients, so changes may not be immediate. For the web client, be sure to clear your cache to see the new changes. On the mobile client, swipe down to refresh app metadata.

Ongoing application maintenance

Once your application has been deployed you can mostly go into maintenance mode responding to user inquiries as needed. Here are a few things to consider while you are between updates.

- PowerApps canvas applications need to be periodically republished for best performance and stability. About every six months you should re-publish your deployed PowerApps canvas applications even if they haven't changed. This ensures the application picks up the latest runtime changes in the environments.
- Keep an eye on your CDS instance storage usage as well as your Flow quotas and adjust resources and licensing as needed.

Retiring and removing an application

As your organization evolves it's likely one or more of the applications deployed will no longer be needed. In this section we will walk through some of the things to consider when retiring an application.

- Confirm that if there are users they understand the shutdown. Consider shutdown notifications in advance to ensure business continuity and minimize impact
- Removing access to the application components is often a good first step. Leaving it in this state for a period of time also helps to ensure users know and have a chance to argue their case or save any data needed.
- Deleting an environment will remove all associated PowerApps, Flows and CDS data. This is not the approach to take if you have multiple applications sharing the environment and you are just retiring a single application.
- PowerApps canvas apps and Flows can usually be removed without lots of dependency considerations. Currently it is necessary to remove these one at a time even if you imported both a PowerApp canvas app and a Flow at the same time. The connections for these will not be removed automatically.
- When removing connections, you need to first consider the PowerApps canvas apps and Flows that might still be using them. This can be checked by looking at what is associated with the connection prior to deleting.
- Custom connections are sometimes better to be left if they might be reused later as they would require extra effort to re-establish in the future.
- To remove a PowerApps model-driven app depends if the CDS solution containing it was installed as managed or unmanaged. If it was installed as unmanaged you can delete the application module to remove it from users. Removing unmanaged CDS solution components requires manually removing one item at a time from the environment. Removing the CDS solution itself in this situation only removes the container and not the components. This is one of the key benefits of managed solution is the ability to uninstall them as a unit.

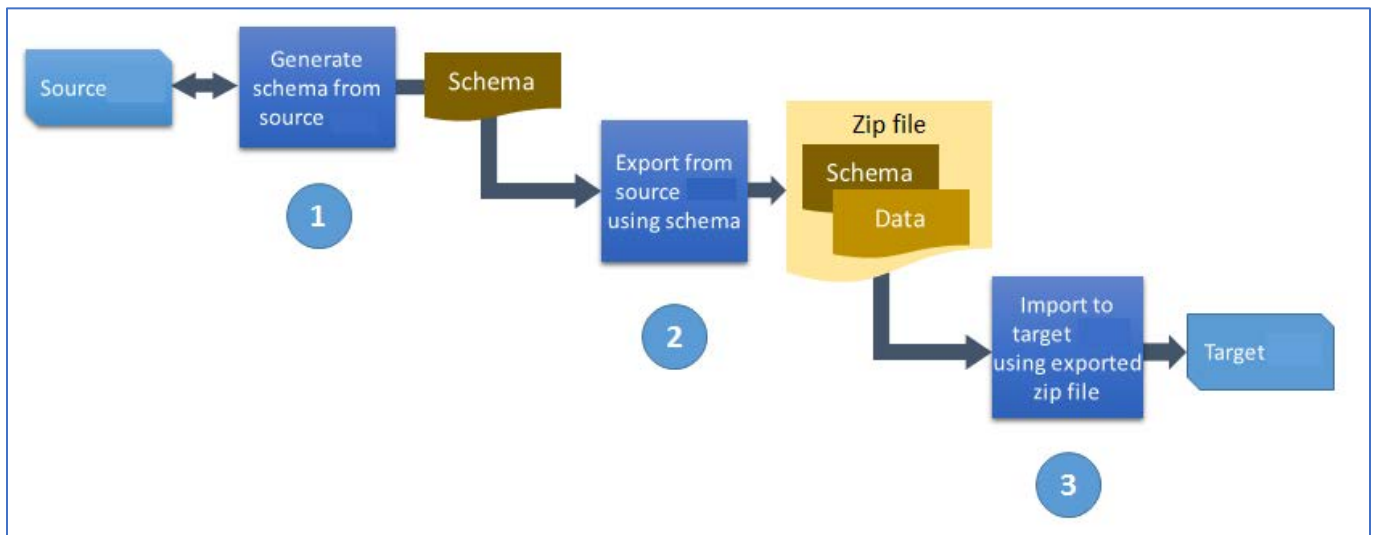
- If the solution installed is managed, you would uninstall/remove the CDS solution containing it from the instance. When you remove the CDS solution that contains that application it's important to note that also removes any other components and data as well. If only desiring to remove the application best approach would be to remove the application in the development environment for that CDS solution and then import the update in using the Stage for Upgrade option on import. This will cause only that component to be removed leaving all other components and data intact.

Moving reference data to another environment

Often applications have data that is configuration, or reference data. This could be, for example, a list of territories, product lists, or other data that configures and makes the app work. Often components in the application take dependencies on the IDs of this data. The Configuration Migration Tool is designed to move this type of data from one CDS instance to another. The key features of the tool are:

- Select only the entities and fields you for which you want to move data
- Maintain unique IDs of the records as they are moved
- Avoid duplicate records by defining a uniqueness condition for each entity based on combination of fields
- Support updating of existing records
- Ability to define a schema for what data is moved and use it over and over.

The following outlines the basic process for using the tool.



The output from the tool is a zip file containing the data and the schema file. The same tool can be used to import the data into the target CDS instance. You can also package the data with a Solution Deployer package that we will discuss shortly allowing it to be deployed alongside one or more CDS solutions. You can read more about how to use the tool here <https://docs.microsoft.com/en-us/dynamics365/customer-engagement/admin/manage-configuration-data> .

Using the Dynamics 365 Package Deployer

So far, we've only talked about importing CDS solutions manually via the user interface. The Dynamics 365 package deployer also works for CDS solutions. The package deployer allows building a package that contains one or more CDS solutions as well as one or more data files to import after the solutions are imported. It is also possible for developers to build custom code that reacts to events from the package deployment process. This code can be used to handle updates to the target environment. Once the package is built, the package can be deployed interactively via the tool, or by command line using PowerShell. You can read more about package deployer here <https://docs.microsoft.com/en-us/dynamics365/customer-engagement/developer/create-packages-package-deployer>.

Compliance and Data Privacy

Microsoft is committed to the highest levels of trust, transparency, standards conformance, and regulatory compliance. Microsoft's broad suite of cloud products and services are all built from the ground up to address the most rigorous security and privacy demands of our customers.

To help your organization comply with national, regional, and industry-specific requirements governing the collection and use of individuals' data, Microsoft provides the most comprehensive set of compliance offerings (including certifications and attestations) of any cloud service provider. There are also tools for administrators to support your organization's efforts. In this part of the document we will cover in more detail the resources available to help you determine and achieve your own organization requirements.

Trust Center

The Microsoft Trust Center (<https://www.microsoft.com/en-us/trustcenter>) is a centralized resource for obtaining information on Microsoft's portfolio of products. This includes information on security, privacy, compliance, and transparency. While this paper may contain some subset of this information for PowerApps, it is important to always refer to the Microsoft Trust Center for the most up to date authoritative information.

For quick reference, you can find the Trust Center Information for the Microsoft Power platform here <https://www.microsoft.com/en-us/TrustCenter/CloudServices/business-application-platform/default.aspx>. This will include information on PowerApps, Microsoft Flow and Power BI.

Data Location

Microsoft operates multiple data centers world-wide that support the Microsoft Power platform applications. When your organization establishes a tenant, it establishes the default geographical (geo) location. In addition, when creating environments to support applications and contain CDS for Apps data the environments can be targeted for a specific geo. A current list of the geos for the Microsoft Power platform can be found here <https://www.microsoft.com/en-us/TrustCenter/CloudServices/business-application-platform/data-location>

To support continuity of operations, Microsoft may replicate data to other regions within a geo, but the data will not move outside the geo to support data resiliency. This supports the ability to fail over or recover more rapidly in the event of a severe outage. There are some reasonable exceptions to keeping data in the specific geo that are listed on the above site primary focused on legal and support. It's also important to note, that you or your users can take actions that expose data outside of the geo. Other

services can also be configured to access the data and expose it outside of the geo. By default, authorized users can access the platform and your applications and data from anywhere in the world where there is connectivity.

Data Protection

Data as it is in transit between user devices and the Microsoft datacenters are secured. Connections established between customers and Microsoft datacenters are encrypted, and all public endpoints are secured using industry-standard TLS. TLS effectively establishes a security-enhanced browser to server connection to help ensure data confidentiality and integrity between desktops and datacenters. API access from the customer endpoint to the server is also similarly protected. Currently, TLS 1.2 (or higher) is required for accessing the server endpoints.

Data transferred through the on-premises data gateway is also encrypted. Data that users upload is typically sent to Azure Blob storage, and all metadata and artifacts for the system itself are stored in an Azure SQL database and Azure Table storage.

All instances of the Common Data Service for Apps database use SQL Server Transparent Data Encryption (TDE) to perform real-time encryption of data when written to disk, also known as encryption at rest.

By default, Microsoft stores and manages the database encryption keys for your instances so you don't have to. The manage keys feature in the Dynamics 365 admin center gives administrators the ability to self-manage the database encryption keys that are associated with instances of Dynamics 365 (online). You can read more about managing your own keys here <https://docs.microsoft.com/en-us/dynamics365/customer-engagement/admin/manage-encryption-keys-instance> but generally it is recommended have Microsoft manage the keys unless you have a specific business need to maintain your own.

Resources to manage GDPR Compliance

The European Union General Data Protection Regulation (GDPR) is one of the newest privacy regulations enacted that gives rights to people to manage their personal data. In this section we will look at some of the tools and resources available for the Microsoft Power platform to assist administrators in their efforts to comply with GDPR. Some of these resources and tools may also helpful to assist you in other data privacy related tasks not directly related to GDPR. A complete discussion of GDPR is beyond the scope of this paper, however in this section we will focus on the tools and resources to support your efforts. Additionally, Microsoft has a section on the trust center dedicated to GDPR resources and information that can be helpful. You can find that here <https://www.microsoft.com/en-us/TrustCenter/Privacy/gdpr/default.aspx>

First, let's review at some of GDPR's terminology that matters in this context:

Term	Relevance
Data Subject	GDPR identifies people as data subjects. It is their personal data that might have been collected by your organization either in the employment of the person or some interaction collecting their personal data

Data Controller	Organizations that collect and process data for their own purposes
Data Processor	Organizations that process data on behalf of others
Personal Data	Any information relating to an identified or identifiable natural person.

As an administrator one of the key activities in support of GDPR will be related to Data Subject Rights (DSR) requests. These are formal requests from a Data Subject to a Data Controller (likely your organization) to act on their personal data in your systems. GDPR gives rights to Data Subjects to obtain copies, request corrections, restrict processing of the data, delete the data and to receive copies in an electronic format so it could be moved to another Data Controller.

The following links point to detailed information to help you respond to DSR requests depending on the features your organization is using

Platform Feature Area	Link to detailed response steps
PowerApps	https://docs.microsoft.com/en-us/powerapps/administrator/powerapps-gdpr-export-dsr
CDS for Apps	https://docs.microsoft.com/en-us/powerapps/administrator/common-data-service-gdpr-dsr-guide
Microsoft Flow	https://docs.microsoft.com/en-us/flow/gdpr-dsr-summary
Microsoft Accounts (MSAs)	https://docs.microsoft.com/en-us/flow/gdpr-dsr-summary-msa
Dynamics 365	https://docs.microsoft.com/en-us/microsoft-365/compliance/gdpr-dsr-dynamics365

Office 365 Security and Compliance Center

You may also find Microsoft Compliance Manager helpful to manage your compliance efforts across Microsoft cloud services in a single place. More details about Compliance Manager can be found here <https://aka.ms/compliancemanager>.

Microsoft Flow Audit Log Events

In the compliance center Audit Log Search administrators can now search and view Microsoft Flow events. Events include Created flow, Edited flow, Deleted flow, Edited Permissions, Deleted Permissions, Started a paid trial, Renewed a paid trial. Using the portal you can choose what you want to search and a time window.

Search Clear Results

Activities Date IP address User Activity

Created flow, ... (7) ▾

× Clear all to show results for all activities

Search

Dynamics 365 activities

- Accessed out-of-box entity
- Accessed admin entity
- Accessed other entity type
- Accessed internal management tool
- Activated process or plug-in
- Accessed custom entity
- Performed bulk actions (such as delete and import)
- Accessed Dynamics 365 admin center
- Signed in or out

Microsoft Flow activities

Created flow	✓	Edited flow	✓
Deleted flow	✓	Edited permissions	✓
Deleted permissions	✓	Started a paid trial	✓
Renewed a paid trial	✓		

From the resulting query results when you drill down into an item you get a details page with the following type of information

Details ✕

Date: 2017-11-02 10:28:23

IP address:

User: kweare@Contoso2016g.onmicrosoft.com

Activity: Created flow

Item: <https://admin.flow.microsoft.com/environments/47ed052a-4945-4f78-9be7-9968688b5b68/resources/flows/6d270c28-fe0a-4863-83eb-15503b0bc2d7>

Detail:

Flow Details Url: <https://admin.flow.microsoft.com/environments/47ed052a-4945-4f78-9be7-9968688b5b68/resources/flows/6d270c28-fe0a-4863-83eb-15503b0bc2d7>

Result Status: Success

More information ▾

Close

The real good information comes from clicking on the More Information and drilling down into the real detail page:

More information ^

CreationTime:	2017-11-02T17:28:23
FlowConnectorNames:	Recurrence,Compose,MSN Weather,MSN Weather,If,Notifications,Notifications,Notifications,Notifications
FlowDetailsUrl:	https://admin.flow.microsoft.com/environments/47ed052a-4945-4f78-9be7-9968688b5b68/resources/flows/6d270c28-fe0a-4863-83eb-15503b0bc2d7
Id:	caa9cb79-c8f9-4c69-96be-f1f390f1d273
LicenseDisplayName:	
Operation:	CreateFlow
OrganizationId:	2f6e98fd-ed85-416f-b7d4-94ad61065d0f
RecipientUPN:	
RecordType:	30
ResultStatus:	Success
SharingPermission:	1
UserId:	kweare@Contoso2016g.onmicrosoft.com
UserKey:	kweare@Contoso2016g.onmicrosoft.com
UserType:	0
UserTypeInitiated:	1
UserUPN:	kweare@Contoso2016g.onmicrosoft.com
Version:	1
Workload:	MicrosoftFlow

Audit data is retained for 90 days. You can do CDSV exports of the data allowing you to move it into Excel or PowerBI for further analysis. You can find a complete walkthrough of using the audit information here <https://flow.microsoft.com/en-us/blog/security-and-compliance-center/>

Support and Learning Resources

In this section let's look at some of the options for gaining further knowledge, and if necessary getting support. Probably one of the best resources is the PowerApps and Microsoft Flow community sites. These are forums that you can post your question and both the community and Microsoft can respond. Often your issue or question has already been discussed and you can simply look at the prior answers.

The PowerApps community can be found here <https://powerusers.microsoft.com/t5/PowerApps-Community/ct-p/PowerApps1> and the Microsoft Flow community here <https://powerusers.microsoft.com/t5/Microsoft-Flow-Community/ct-p/FlowCommunity>

After you have exhausted community resources or have an issue that is critical, you can start the ticket process for PowerApps here <https://powerapps.microsoft.com/en-us/support/>; it will also list any top issues others are having and is used to share broader outage information as well. A similar site for Flow can be found here <https://us.flow.microsoft.com/en-us/support/>


Blogs

The PowerApps and Flow teams blog frequently on both new updates as well as ongoing examples of using the features of the products. The Flow blog for example has an ongoing series of beginner flows and intermediate flows. These are great ways to get ideas even if you don't need that exact solution, it can give you ideas on how to handle similar scenarios.

Submitting and Voting on Ideas

The best way to help provide ideas and shape the future of PowerApps and Flow is to post your idea or vote on existing ideas to help prioritize them. PowerApps maintains their idea list here <https://powerusers.microsoft.com/t5/PowerApps-Ideas/idb-p/PowerAppsIdeas> and Flow's list can be found here <https://powerusers.microsoft.com/t5/Flow-Ideas/idb-p/FlowIdeas>. This is also an easy way to see if something has gotten Microsoft's attention or even already completed. Items that can indicate if they are in planning, are in progress by being marked as started, or even already completed.


360
Vote

**Office 365 Planner**
by Erihje on 06-02-2016 05:04 AM

Hi,

Will you implement Flow for Office 365 Planner also, if so do you have any ETA for this?

Thanks for an excellent solution so far!

 **COMPLETED** Stephen (Flow Staff) responded
Good news: you can now create planner tasks with the new connector. Read more here: <https://flow.microsoft.com/blog/planner-community-and-licenses/> - if you have further requests for planner please open new suggestions. Thanks!

Guided Learning

Guided learning are short courses that can be consumed by both makers and administrators. The PowerApps courses can be found here <https://docs.microsoft.com/en-us/powerapps/guided-learning/> Administrators will likely find the managing application courses a good fit. For Flow you can find the courses here <https://docs.microsoft.com/en-us/flow/guided-learning/> There are also a couple courses on administering flows would be good for administrators.

Finding Consulting Partners

If you find that you or your teams are looking for some outside assistance you can use the Partner Finder to locate a partner that specializes in PowerApps and Flow. You can find the list of partners here

<https://powerapps.microsoft.com/en-us/partners/> The Partner Showcase is also a good place for inspiration as well as to take a look at some of the amazing things partners have built on the platform. You can find the showcase here <https://powerapps.microsoft.com/en-us/partner-showcase/>

Next Steps

Congratulations, you've made it to the end! We hope you found some helpful information and keep this paper around for future reference. To remind you of where we started our journey at the beginning of the document, here are the actions you should consider taking in the first year to get started on the right track!

First 30 days	<p>Focus on gaining a good understanding of what you have in place already from both a licensing and application point of view. Take time to learn your organization's goals for the platform and how you can help them succeed. Here is your checklist:</p> <ul style="list-style-type: none">- Check to see what users have already built- Get familiar with the Admin Portals- Create some default Data Protection Policies- Setup a flow to notify you when users create new connections
First 6 Months	<p>Develop and implement your plan for environments, security and data protection policies. Put in place how you want teams to work with you to deploy new applications and establish an ongoing cadence for updates.</p>
After 6 months	<p>Use monitoring and insights to look for any needed adjustments in your environments. Look for ways you can improve your ALM processes by leveraging automation such as PowerShell and Microsoft Flow for common admin tasks</p>