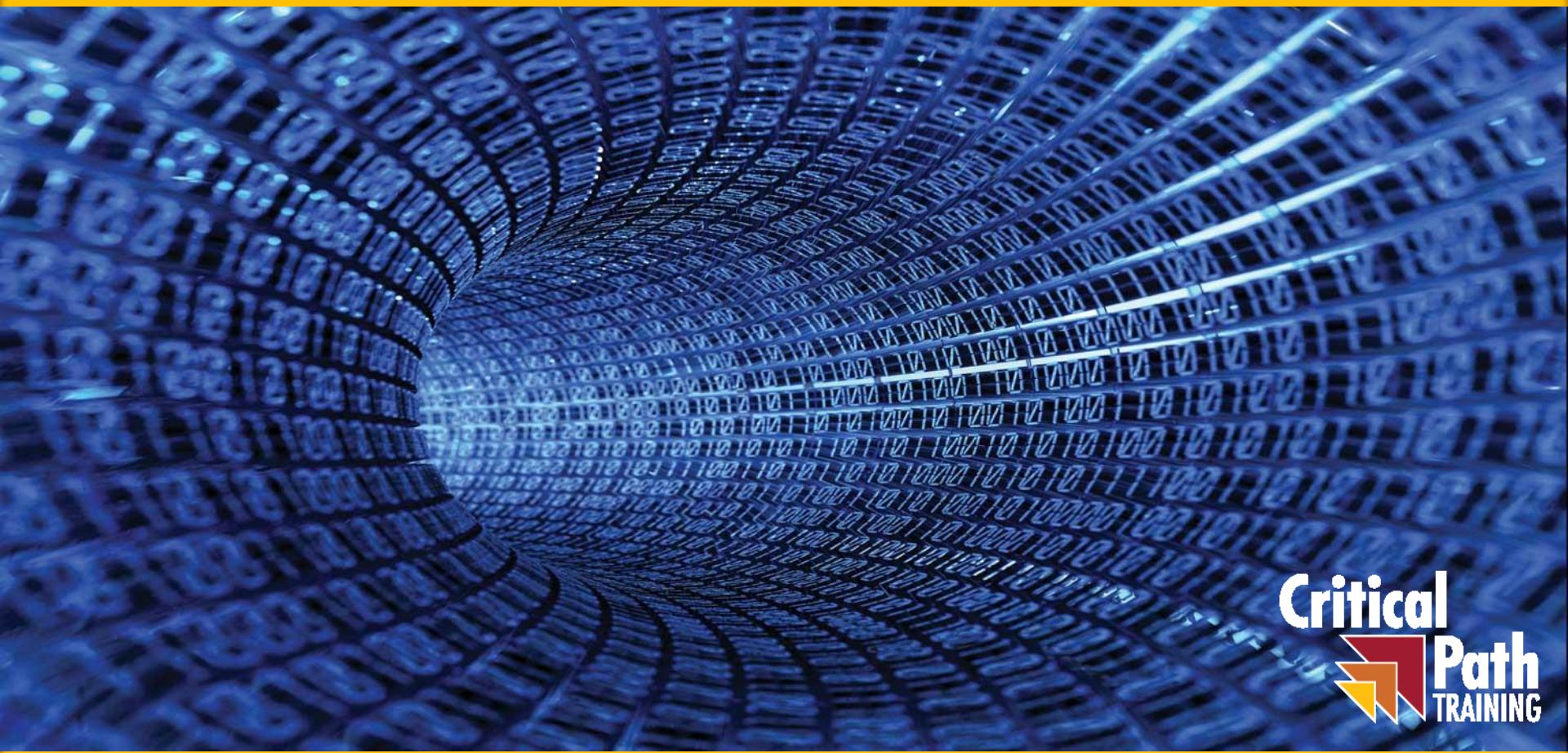


# Introduction to PowerApps and Flow



Critical  
Path  
TRAINING

# Agenda

- Microsoft Flow
- PowerApps
- Common Data Model



# PowerApps Studio Interface

The screenshot illustrates the PowerApps Studio interface with several numbered callouts highlighting specific features:

- 1** **Properties Panel**: Located on the left side, showing a tree view of the screen elements and their properties.
- 2** **List View**: A grid view of budget items with columns for category, budgeted amount, spent amount, and remaining amount.
- 3** **Screen Properties**: A panel on the right showing properties for the current screen, including **OnVisible**, **OnHidden**, and **OnStart**.
- 4** **Color Picker**: A color picker in the ribbon bar set to `RGB(255, 255, 255, 1)`.
- 5** **Color Swatch**: A small color swatch in the ribbon bar.
- 6** **Ribbon Bar**: The top navigation bar with tabs like File, Home, Insert, View, Action, and Screen.
- 7** **Tab Selection**: A red circle highlighting the "Insert" tab in the ribbon bar.

**Budget Tracker Screen Content:**

**All budgets**

| Category               | Budgeted | Spent   | Remaining |
|------------------------|----------|---------|-----------|
| Team events at Contoso | \$10,000 | \$1,950 | \$8,050   |
| Spring customer visits | \$5,000  | \$3,175 | \$1,825   |
| Meta Conference XVI    | \$10,000 | \$8,732 | \$1,268   |
| Litware team event     | \$5,000  | \$1,175 | \$3,825   |

**Total spent** \$1,950

**Attached receipts**

No receipts

**Expenses**

| Expense Type | Amount  |
|--------------|---------|
| Cake         | \$300   |
| Movie        | \$500   |
| Extra        | \$1,000 |
| Cleaning     | \$150   |

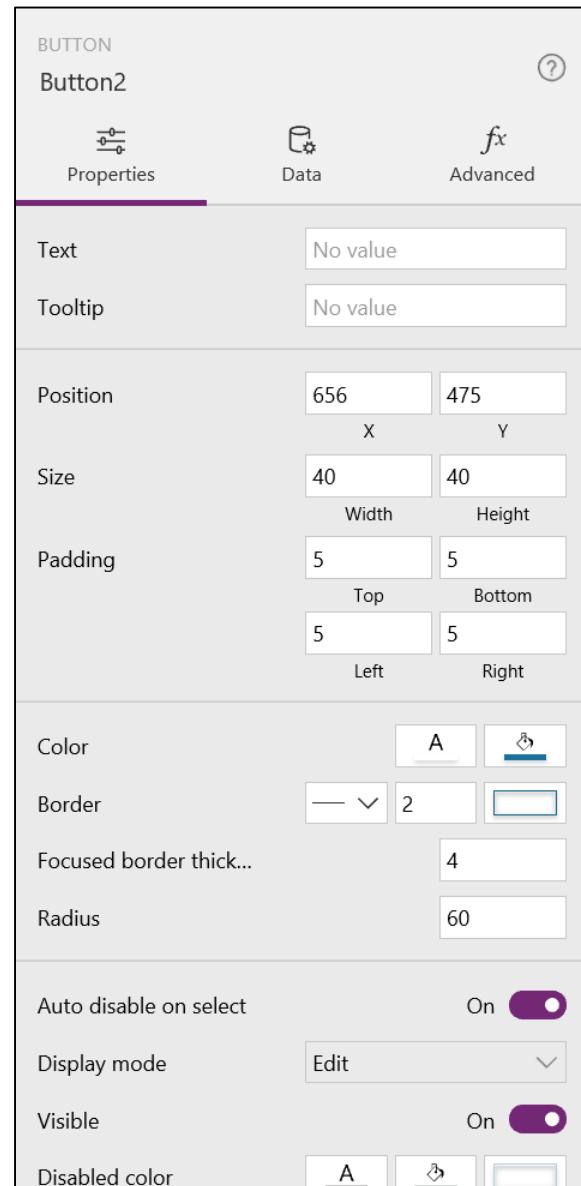
# Panels

- **Properties Panel**
  - Will expose most common properties of any control you can configure visually without writing an expression
- **Data Panel**
  - Will surface configuring data sources and binding fields to Gallery and Form controls
- **Advanced Panel**
  - Used for more advanced customizations for controls



# Properties Panel

- Configure controls without using a formula
  - Varies based on control
- Configure properties such as:
  - Text
  - Tooltip text
  - Position
  - Look & Feel
    - Size
    - Padding
    - Colors
    - Borders



# Data Panel

- Will list existing data sources
- Also provides ability to add new data sources

The screenshot shows the Data Panel interface. At the top, there are three tabs: Properties, Data (which is selected), and Advanced. Below the tabs, the title "Data sources" is followed by a help icon and a "Add data source" button with a plus sign. A list of five data sources is displayed, each with a blue cloud icon, the source name, the owner's email, the provider, and a three-dot ellipsis menu icon.

| Source            | Owner                      | Provider              | Actions |
|-------------------|----------------------------|-----------------------|---------|
| Budgets           | zoe@ck2017.onmicrosoft.com | OneDrive for Business | ...     |
| Expenses          | zoe@ck2017.onmicrosoft.com | OneDrive for Business | ...     |
| Categories        | zoe@ck2017.onmicrosoft.com | OneDrive for Business | ...     |
| ExpenseByCategory | zoe@ck2017.onmicrosoft.com | OneDrive for Business | ...     |
| Receipts          | zoe@ck2017.onmicrosoft.com | OneDrive for Business | ...     |

# Advanced Panel

- Configure more advanced customizations for controls

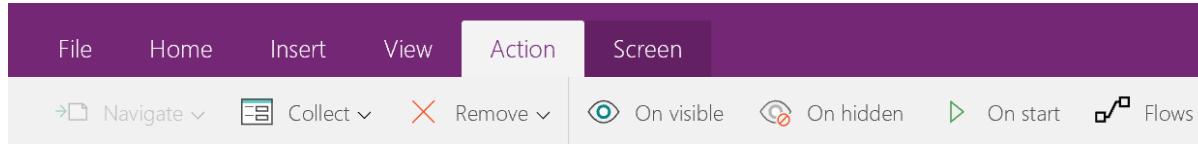
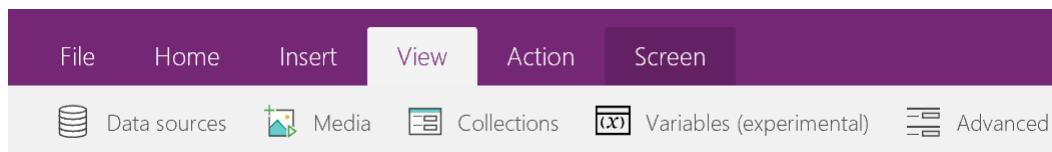
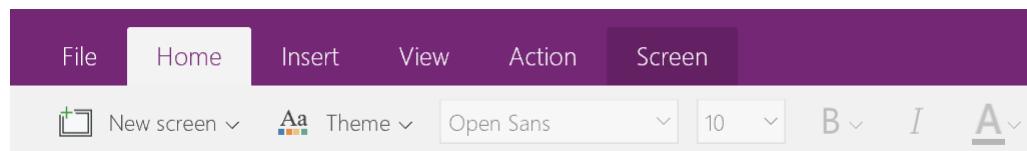
The screenshot shows the 'Advanced' tab selected in the top navigation bar of a configuration interface. The interface is divided into three main sections: ACTION, DATA, and DESIGN.

- ACTION:** Contains the code:

```
OnSelect
Navigate
(AddExpenseScreen, ScreenTransition.Fade,
{SpendRecord:SpendRecord, Record:Defaults(Expenses)});ResetForm
(FormNewExpense);NewForm
(FormNewExpense);Clear
(ReceiptsCollect)
```
- DATA:** Contains the field "Tooltip" with the value "...".
- DESIGN:** Contains the fields "AutoDisableOnSelect" set to "true" and "Color" set to "RGBA(255, 255, 255, 1)".

# Ribbon

- Set of context-sensitive tabs which will display controls you can use to build your app:
  - Home tab
  - Insert tab
  - View tab
  - Action tab



# PowerApps from a Template

- Create an app automatically based on a template for a specific scenario
  - Pre-built layouts and colour schemes for different app screens you can directly use or clone
  - Templates have predefined screens, features, and sample data
- Customize app made from template
  - Change layout, screens, features, and delete sample data



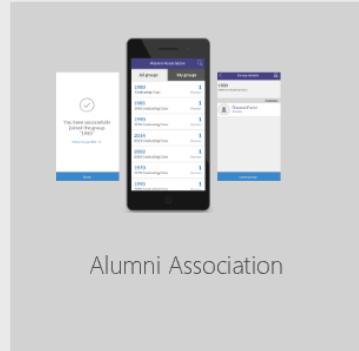
# Learning from Template Apps

- Learn how controls are being configured for common actions such as:
  - Submit data from a form by clicking on a button
  - Transition from one app screen to the other
  - Show a list of items from my data, etc.
  - Data flows in and out of your app
  - Wire up your data source to your app
  - Camera or GPS are being integrated into your app



# Phone Layout Templates

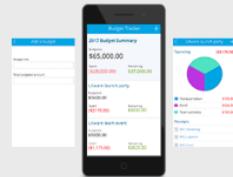
## Templates



Alumni Association



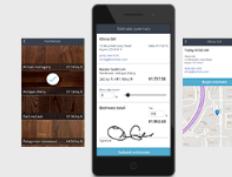
Asset Checkout



Budget Tracker



Case Management



Cost Estimator



Employee Engagement Survey



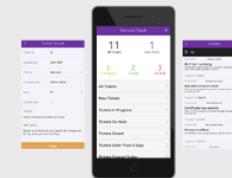
Health Plan Selector



Invoice Management



Quick Tips



Service Desk

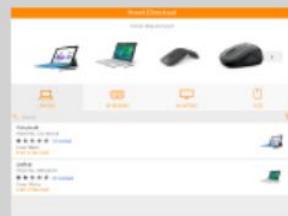


Site Inspection

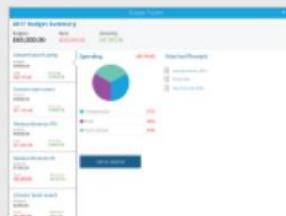


# Tablet Layout Templates

## Templates



Asset Checkout



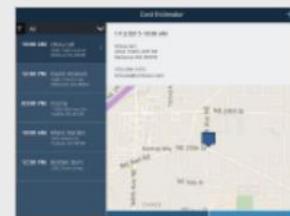
Budget Tracker



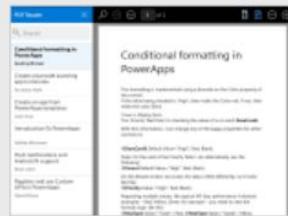
Case Management



Contest Registration



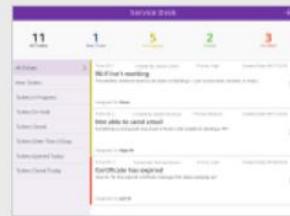
Cost Estimator



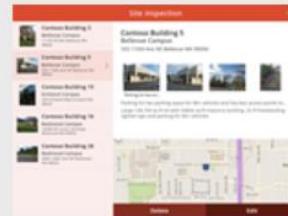
PDF Reader



Product Showcase



Service Desk

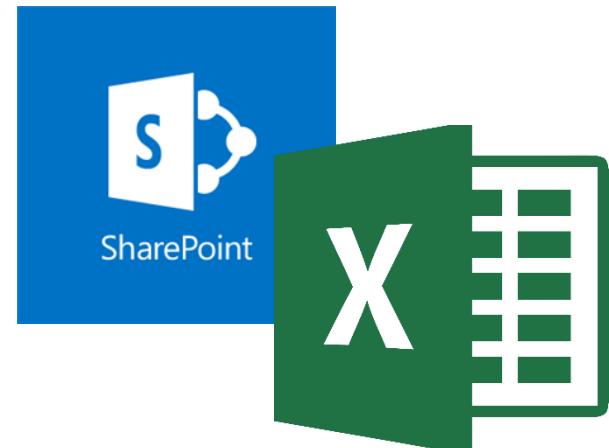


Site Inspection



# Create App from Data Source

- In PowerApps, you can automatically generate an app based on a data source which includes:
  - SharePoint List
    - Create directly from SharePoint list
    - Or create a connection to the SharePoint list
  - Entity (entities will be discussed further in Module 4)
  - Excel data



# SharePoint List

- Generate an app automatically based on data in a SharePoint list
  - Users can manage items in app for SharePoint list
  - Libraries are NOT supported
- By default, the app will have 3 screens:
  - **BrowseScreen1** - browse through all records in the list
  - **DetailsScreen1** - view all fields for a specific record
  - **EditScreen1** - create or edit a record
- Customize these screens based on your needs
- Not all columns are supported



# Supported/Unsupported Columns

| Column type            | Support                  | Default cards                               |
|------------------------|--------------------------|---|
| Single line of text    | Yes                      | View text                                   |
| Multiple lines of text | Yes                      | View text                                   |
| Choice                 | Yes (single values only) | View lookup                                 |
| Number                 | Yes                      | View percentage<br>View rating<br>View text |
| Currency               | Yes                      | View percentage<br>View rating<br>View text |
| Date and Time          | Yes                      | View text                                   |
| Lookup                 | Yes (single values only) | View lookup<br>Edit lookup                  |
| Boolean (Yes/No)       | Yes                      | View text<br>View toggle                    |



# Supported/Unsupported Columns

| Column type      | Support                  | Default cards              |
|------------------|--------------------------|----------------------------|
| Person or Group  | Yes (single values only) | View lookup<br>Edit lookup |
| Hyperlink        | Yes                      | View URL<br>View text      |
| Picture          | Yes (read-only)          | View image<br>View text    |
| Calculated       | Yes (read-only)          |                            |
| Task Outcome     | No                       |                            |
| External data    | No                       |                            |
| Managed Metadata | Yes (read-only)          |                            |
| Rating           | No                       |                            |



# Excel Data

- You can use an Excel file that has table data.
- Excel files must be uploaded to a cloud service before you can create the PowerApp:
  - Box
  - Dropbox
  - FTP
  - Google Drive
  - OneDrive
  - OneDrive for Business



# Creating Custom PowerApps

- You can create your own app from scratch using the Blank app template:
  - Using any one of a variety of data sources and then adding more sources later.
  - This approach is much more time-intensive than generating an app automatically.
  - Experienced app makers can build the best app for their needs.
  - Specify the appearance and behavior of each UI element so that you can optimize the result for your exact goals and workflow.



# Key Points

- PowerApps can be created in PowerApps Studio
- You can create apps automatically from:
  - Existing templates
  - Data sources such as a SharePoint custom list or Excel file
- Microsoft does continue to add more templates which they announce on their blog site.
- Not all SharePoint columns are supported in PowerApps



# App Name & Icon

- Name, icon, and description settings your app
- **File tab > App Settings (PowerApps Studio)**

The screenshot displays the 'App settings' interface in PowerApps Studio, divided into two main sections: 'PowerApps Studio for web' and 'PowerApps Studio for Windows'.

**PowerApps Studio for web:**

- App name:** IT Service Desk Demo - Tablet
- Icon:** Preview shows a blue square with a white gear icon.
- Background color:** A 4x4 grid of color swatches.
- Icons:** A grid of icons including a checkmark, star, magnifying glass, arrows, and other symbols, with a 'Browse' button.
- App name:** Service Desk - Tablet
- Description:** A text input field asking to describe what people can do with the app.

**PowerApps Studio for Windows:**

- Icon:** Preview shows a purple square with a white circular arrow icon.
- Background color:** A 4x4 grid of color swatches.
- Icons:** A grid of icons including a checkmark, star, magnifying glass, arrows, and other symbols, with a 'Browse' button.
- Description:** A text input field asking to describe what people can do with the app, with a link to 'Learn about Dynamics 365'.

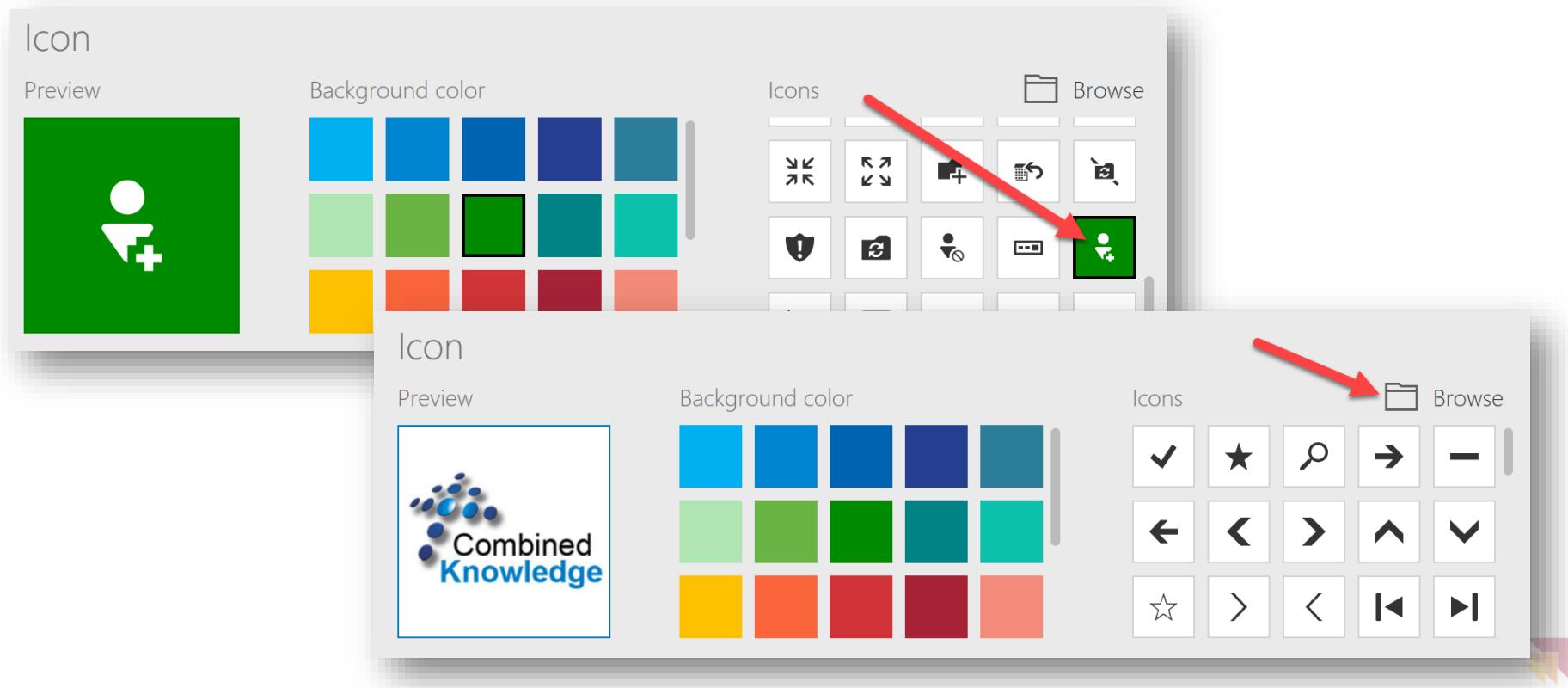
**App settings sidebar:**

- App name + icon:** Change the name or icon for the app
- Screen size + orientation:** Change the aspect ratio for your app



# App Background & Icon

- Use pre-defined icons & change background color or use your own image
  - For best results, use square icon (i.e. 300x300)



# Screen Size & Orientation

## Phone layout

### App settings

- App name + icon  
Change the name or icon for the app
- Screen size + orientation  
Change the aspect ratio for your app



#### Orientation

- Landscape
- Portrait

#### Advanced settings

##### Lock aspect ratio

Locking this automatically maintains the ratio between height and width to prevent distortion.

- On

##### Lock orientation

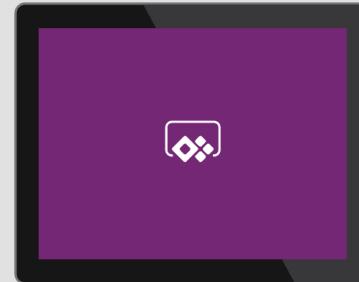
Locking orientation keeps the screen in its current orientation, even when the device is rotated.

- On

## Tablet layout

### Screen size + orientation

Choose the screen size and orientation that your users will most likely be using.



1024 x 768

#### Orientation

- Landscape
- Portrait

#### Size

- 16:9 (Default)
- 3:2 (Surface Pro 3)
- 16:10 (Widescreen)
- 4:3 (iPad)

#### Advanced settings

##### Lock aspect ratio

Locking this automatically maintains the ratio between height and width to prevent distortion.

- On

##### Lock orientation

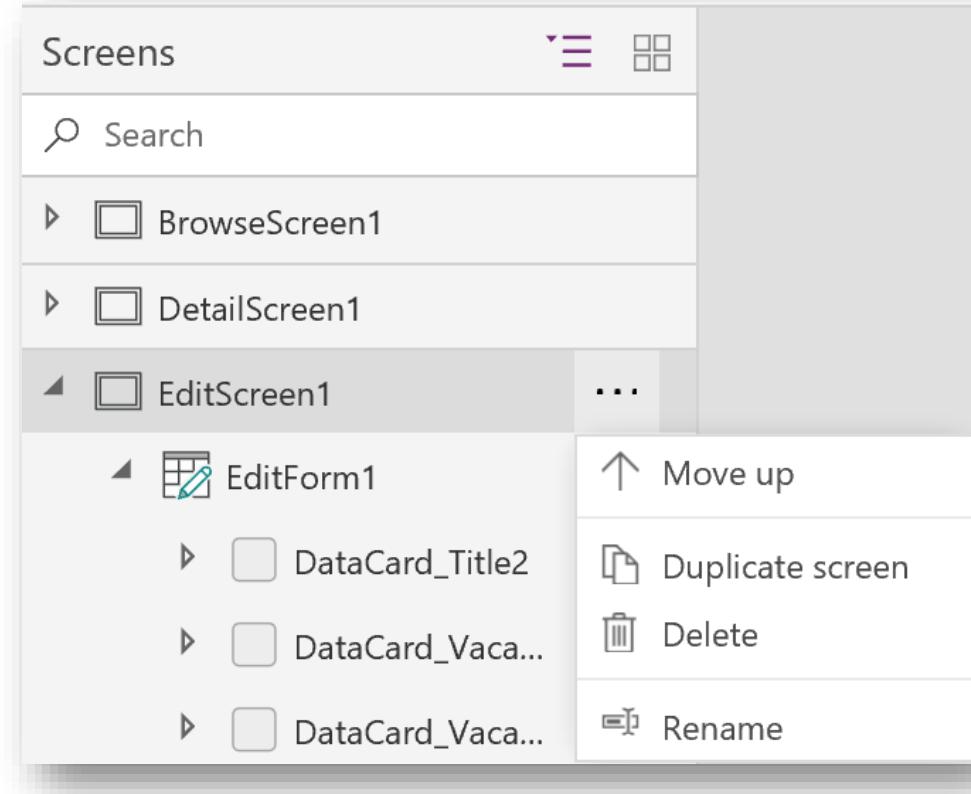
Locking orientation keeps the screen in its current orientation, even when the device is rotated.

- On



# Customizing Screens

- Move (reorder)
- Duplicate
- Add screens
- Delete screens
- Rename screens



# Screen Layout Settings

The screenshot shows the Power BI desktop application interface with the ribbon menu at the top. The 'Screen' tab is selected. A context menu is open over a visual element, specifically the 'Image position' dropdown. The 'Image position' dropdown menu is displayed, listing five options: Fill, Fit, Stretch, Tile, and Center. The 'Fit' option is currently selected. To the right of the ribbon, a color palette is visible, showing a grid of theme colors and a preview color bar.

File Home Insert View Action Screen

Background image Image position Fill

Add an image file = fx

Basel\_Add

Basel\_ArrowsUpDown

Basel\_Cancel

File Home Insert View Action Screen

Background image Image position Fill

ImagePosition

Fill

Fit

Stretch

Tile

Center

Color

Theme colors

Fill

No fill



# Form Layouts

**Vacation Request**

\* Vacation Start Date  
12/31/2001

\* Vacation End Date  
12/31/2001

\* Time Off Type

Comments

Card : PTO

PTO



CARD

DataCard\_PTO2

Properties Data Advanced

Form customization

Vacation Request  
zoe@ck2017.onmicrosoft.com  
SharePoint

Refresh

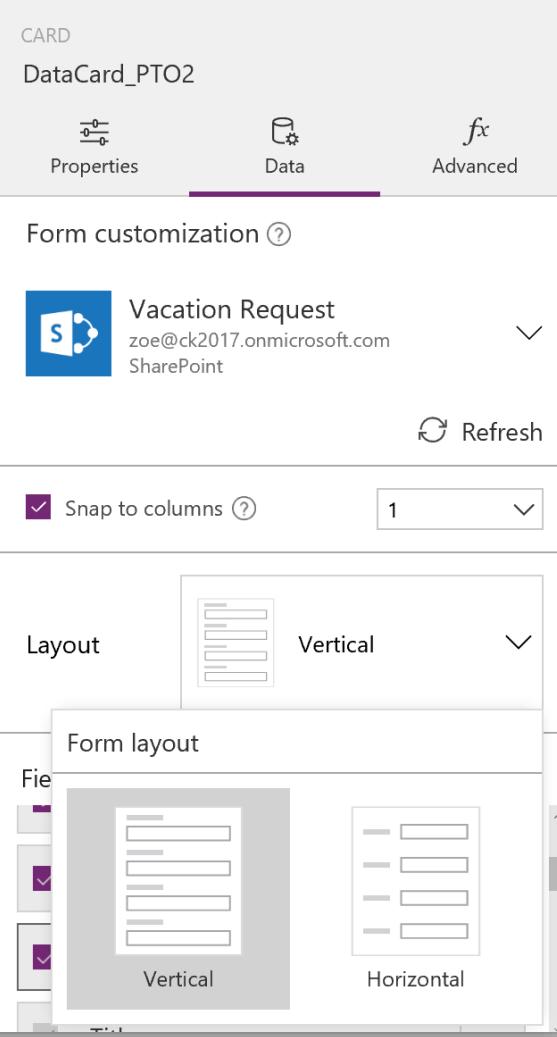
Snap to columns 1

Layout Vertical

Form layout

Vertical

Horizontal





# Controls

- You can add a variety of UI elements to your app which are called controls.
  - Controls have properties that can be configured to change the appearance and behavior of each control
- Properties can be configured:
  - Directly from the toolbar
  - In the **Properties** tab
  - Or in the formula bar



# Adding Controls

- **Text:** Label, Text input, HTML Text, Pen input
- **Controls:** Button, Dropdown, Date picker, List box, Checkbox, Radio, Toggle, Slider, Rating, Timer
- **Gallery:** Vertical, Horizontal, Flexible height, Blank vertical, Blank horizontal, Blank flexible height
- **Data table**
- **Forms:** Edit, Display, Entity form
- **Media:** Image, Camera, Barcode, Video, Audio, Microphone, Add picture
- **Charts:** Column chart, Line chart, Pie chart
- **Icons**



# Modifying Control Properties

The screenshot shows a "Vacation Request" form with the following fields:

- Title
- Vacation Start Date
- Vacation End Date
- Time Off Type
  - PTO (selected, indicated by a checked toggle switch)
  - Request Status
- Modified

The "Time Off Type" section has a tooltip: "Card : PTO".

The "Properties" panel on the right is open for the "DataCard\_PTO1" control, showing the following settings:

| Setting      | Value          | Unit |
|--------------|----------------|------|
| Position X   | 0              |      |
| Position Y   | 320            |      |
| Size Width   | 640            |      |
| Size Height  | 103            |      |
| Color        | (Color swatch) |      |
| Border       | 0              |      |
| Display mode | View           |      |
| Visible      | On             |      |
| Width fit    | Off            |      |



# Card Controls and Data Cards

- **Card controls**
  - Building blocks of the **Edit** and **Display** form controls
  - Form represents entire record
  - **Card** represents a single field of the record
- Interact with **cards** in right-hand pane after you select a form control in design workspace
  - In the pane:
    - Choose which fields to show
    - Change order of fields
    - Change how to show each field



# Data Cards

The screenshot shows the Microsoft Power Apps interface for customizing a Data Card. On the left, a preview of the Data Card is displayed with a red border around the main content area. The card has a title "Vacation Request" and contains fields for Title, Vacation Start Date, Vacation End Date, Time Off Type, Comments, PTO, and Request Status. A red arrow points from the top right towards the Fields section on the right.

**CARD**  
DataCard\_Title2

Properties Data Advanced

Form customization

Vacation Request  
zoe@ckpowerapps.onmicrosoft.com  
SharePoint

Refresh

Snap to columns 1

Layout Vertical

Fields Add custom field

Title abc ...

View text View phone View email

Edit text Edit multi- Allowed

m1 DataCard\_Title2 60%

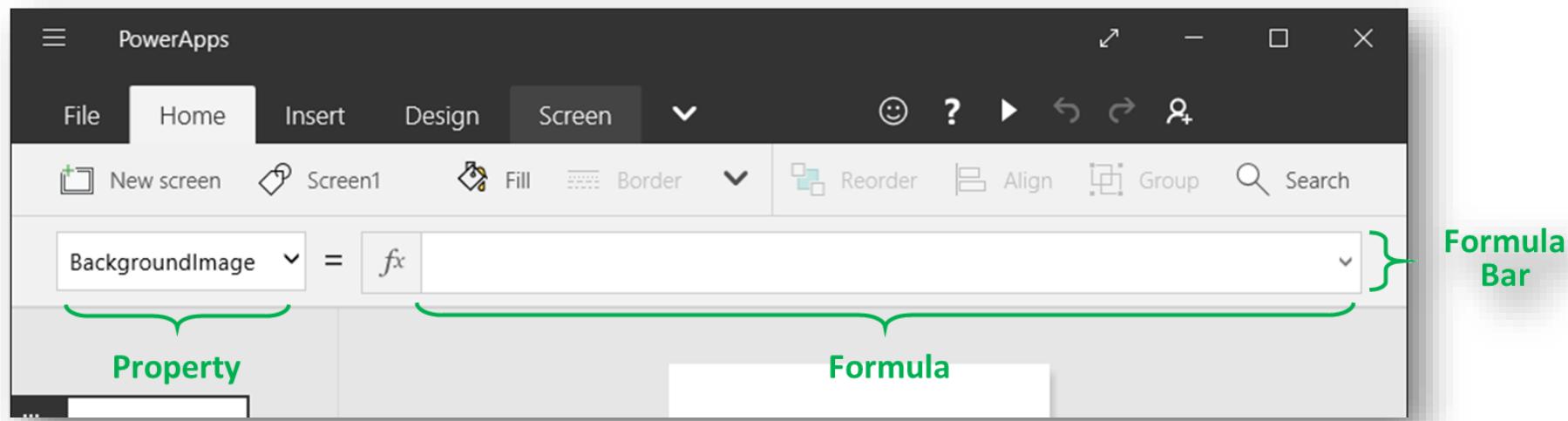
# Working with the Formula Bar

- **Excel Formulas**
  - Build formulas that populate cells
  - Create tables and charts
- **PowerApps Formulas**
  - Build formulas similar to Excel as you configure controls instead of cells
  - Build formulas that apply specifically to apps instead of spreadsheets



# Working with the Formula Bar

- Formula sits on top of the screen and has two parts:
  - **Property list:** Each control and screen has a set of properties. Use this list to select a specific property.
  - **Formula:** The formula to be calculated for this property, made up of values, operators, and functions.



# Working with the Formula Bar

- Formula **Text** of a **Label** control
  - All strings must be in double quotes “**string text**”

The screenshot shows the PowerApps formula bar interface. The top navigation bar includes File, Home, Insert, View, Action, and a tab labeled "Label". Below the navigation bar are various controls for styling: Overflow (Overflow dropdown, Auto height button), Line height (1.2), Padding, Vertical align, Visible (eye icon), and Border (border dropdown). The main formula bar has a "Text" dropdown set to "Vacation Request" and an "fx" button. A red bracket on the left points to the "Text" dropdown with the label "Property 'Text'". A red bracket on the right points to the formula with the label "Formula: string 'Vacation Request'".

Property "Text"

Formula: string "Vacation Request"

Screens

Vacation Request

\* Title

Time off Request

\* Vacation Start Date

9/4/2017

\* Vacation End Date

9/10/2017

LABEL

LblAppName3

Properties

Data

Advanced

Text

Vacation Request

Auto height

Off

Line height

1.2

# Formula Examples (Label Text)

Sum of numbers

The screenshot shows the PowerApps formula editor interface. At the top, there is a text input field containing "Text" and an equals sign (=). To the right of the equals sign is a formula bar with "fx" and "Sum(10,15)". Below this, a "Screens" section displays a card with a "Vacation Request" header. Inside the card, there is a list item for a "SharePoint conference in Stockholm" dated 8/22/2017 at 4:45 PM. To the right of the card is a large number "25" enclosed in a rounded rectangular container with four circular handles.

Return now (date and time)

The screenshot shows the PowerApps formula editor interface. At the top, there is a formula bar with "fx" and "Now()". Below this, a "Screens" section displays a card with a "Vacation Request" header. To the right of the card is a date and time "8/24/2017 1:15 AM" enclosed in a rounded rectangular container with four circular handles.

Add 5 days to today's date

The screenshot shows the PowerApps formula editor interface. At the top, there is a formula bar with "fx" and "DateAdd(Today(), 5, Days)". Below this, a "Screens" section displays a card with a "Vacation Request" header. To the right of the card is a date "8/29/2017" enclosed in a rounded rectangular container with four circular handles.

Return today's date

The screenshot shows the PowerApps formula editor interface. At the top, there is a formula bar with "fx" and "Today()". Below this, a "Screens" section displays a card with a "Vacation Request" header. To the right of the card is a date "8/24/2017" enclosed in a rounded rectangular container with four circular handles.

<https://powerapps.microsoft.com/en-us/tutorials/formula-reference/>



# Formula Errors

The screenshot shows the PowerApps formula editor interface. A formula is being typed into a text box:

```
Text = Sum(1,2,3|
```

A tooltip for the `Sum` function is displayed above the formula bar:

Sum(number, number, number, ...)  
number: A numeric value for this sum operation.

A yellow exclamation point icon is located in the bottom-left corner of the formula bar area, with a green arrow pointing towards it from the text "Yellow exclamation points indicate a problem until the formula is complete (closing parenthesis added)".

A green curly brace on the right side of the formula bar indicates that the formula is "Partial, missing an ending paren".

Yellow exclamation points indicate a problem until the formula is complete (closing parenthesis added).

Formula: Partial, missing an ending paren



# Change Value Based on Input

| Book1 - Excel |        |           |
|---------------|--------|-----------|
|               | $f\!x$ | = A1 + A2 |
|               | A      | B         |
| 1             | 49     |           |
| 2             | 64     |           |
| 3             | 113    |           |
| 4             |        |           |

A3's formula depends on the values of cells A1 and A2

The screenshot shows a software interface with a formula bar at the top. The formula bar displays "Text" followed by an equals sign and a formula:  $fx = \text{TextInput1} + \text{TextInput2}$ . Below the formula bar is a "Screen1" window. Inside the window, there are two text input fields: one containing the value "49" labeled "← TextInput1" and another containing the value "65" labeled "← TextInput2". At the bottom of the window, there is a horizontal text box containing the value "114" with a label "← TextBox1" pointing to it.



# Form Interactivity & Navigation

- Create buttons and set the navigation to go back and forth between screens

The screenshot displays a mobile application interface with a navigation bar at the top containing icons for New screen, 27, B, I, U, ab, A, Fill, and a three-dot menu. Below the navigation bar is a search bar with the placeholder "OnSelect" and an equals sign followed by an fx icon and the code `ResetForm(EditForm1);Back()`. To the left of the search bar is a "Screens" section with a search input and a list of screens: BrowseScreen1, DetailScreen1, and EditScreen1. The "EditScreen1" item is currently selected. In the center of the screen is a "Vacation Request" form with two fields: "Vacation Start Date" (12/31/2001) and "Vacation End Date" (12/31/2001). At the bottom of the screen is another search bar with the placeholder "OnSelect" and an equals sign followed by an fx icon and the code `Navigate(BrowseScreen1, None)`. The overall theme is blue and white.



# Data Sources & Connectors

- **Data Sources**
  - External information stored in cloud services
  - Examples: Excel workbooks, SharePoint lists, SQL tables
- **Connectors**
  - A proxy or wrapper around an API which allows the underlying service to talk to PowerApps and Flow



SharePoint



Office 365 Users



Salesforce  
PREMIUM



Dropbox



Excel



SQL Server



Twitter



OneDrive for Business



Dynamics 365



OneDrive



# SQL Database

- **Two connection options for SQL:**
  - Cloud (Azure SQL Database)
  - On-premises SQL Server
- **Generate app automatically**
  - From SQL Database connection
- **Build app from scratch**
  - Add new connection to new app
- **Update an existing app**
  - Add new connection to existing app

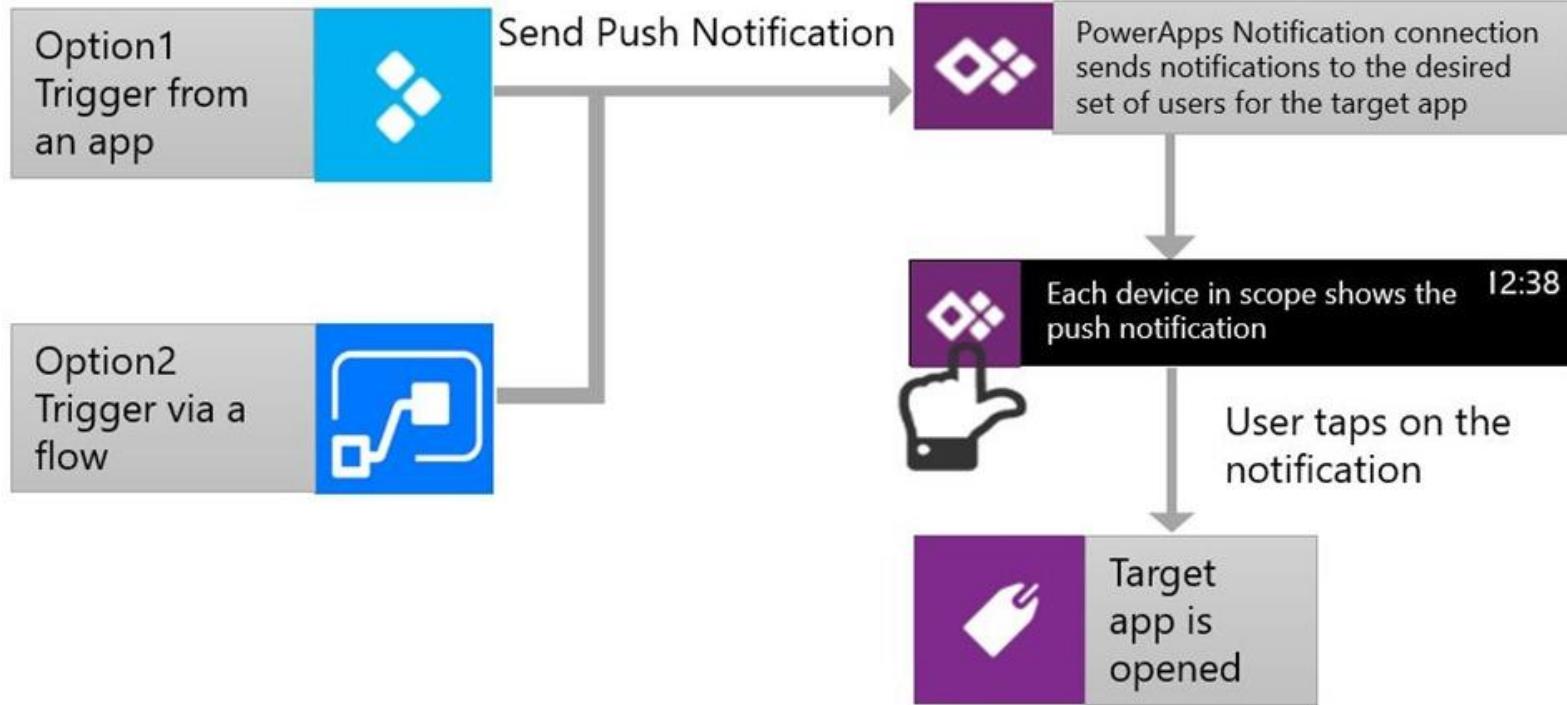


# Components of a Connector

- Components offers a set of operations:
  - **Actions**
    - Changed directed by a user such as an action to lookup, write, or delete data
  - **Triggers**
    - Can notify your app when a specific event occurs
  - Two types of triggers:
    - **Polling triggers** – These triggers call your service at a specified frequency to check for new data.
    - **Push triggers** – Provides ability to send various notifications that directly targets your Apps. Trigger push notification directly from an App or from a Flow.



# Push Notification Examples

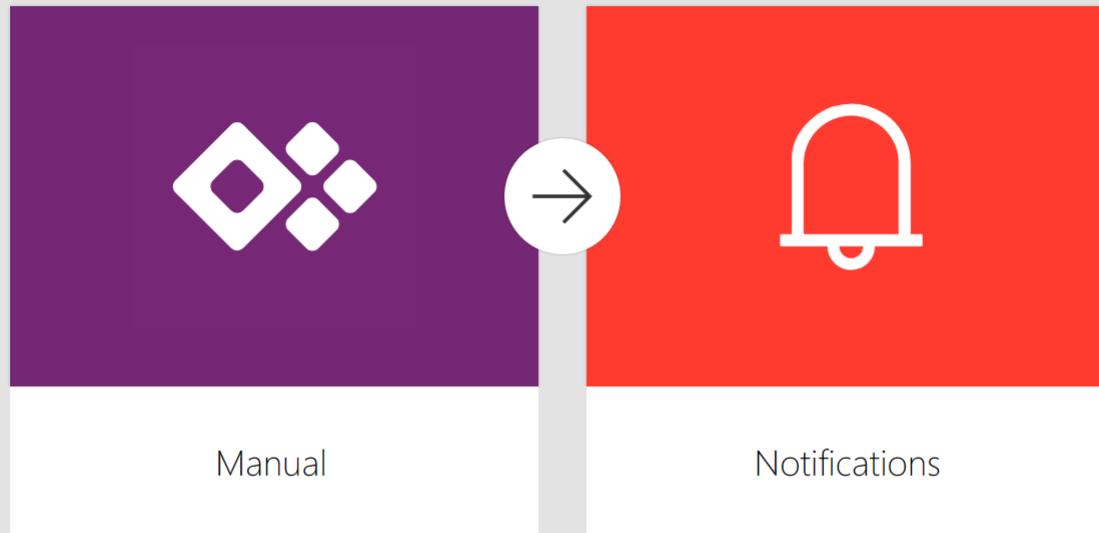


*Triggering and opening Push notifications with PowerApps.*



# Push Notification Examples

Send a push notification triggered by PowerApps



This flow sends a push notification to your Flow App through PowerApps, when triggered with a button tap. You will need an installation of the Flow App for this template.

[Use this template](#)



# Understanding Delegation

- **Delegation** is where you can delegate the processing of data to a data source before the data is pulled in an app
  - Key to building efficient apps
  - Minimizes the amount of data that needs to be brought into the app
- PowerApps includes a powerful set of functions to filter, sort, and shape tables of data
  - Equivalent to writing database queries



# Types of Delegate Functions

- **Filter functions**
  - *Filter*, *Search*, and *LookUp* can be delegated
- **Sorting functions**
  - *Sort* and *SortByColumns* can be delegated
- **Aggregate functions**
  - *Sum*, *Average*, *Min*, and *Max* can be delegated
  - Not all data sources support this delegation

<https://powerapps.microsoft.com/en-us/tutorials/delegation-list/>



# Small and Large Data Sets

- **Large data sets (over 500 records)**
  - Requires using data sources and formulas that can be delegated
  - Only way to keep your app performing well and ensures users can access all the information they need
- **Small data sets (less than 500 records)**
  - Can use any data source and formula
  - Processing can be done locally if the formula cannot be delegated



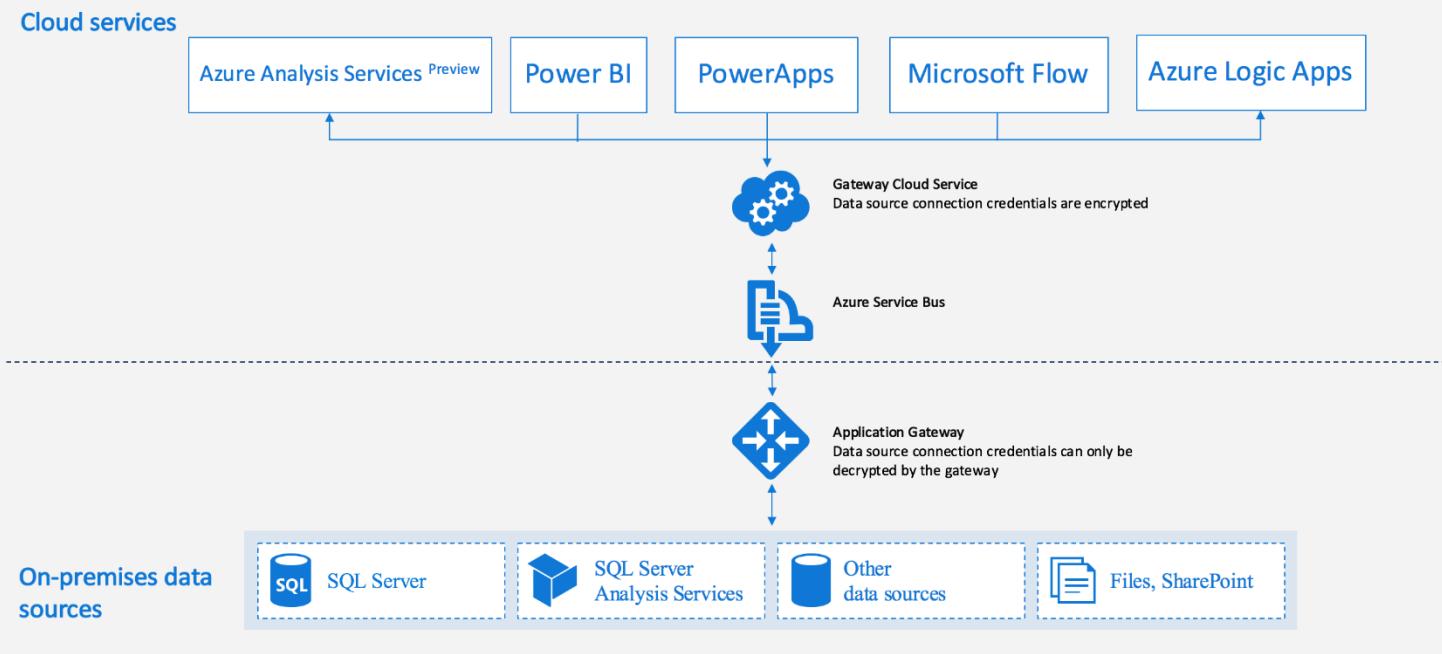
# On-premises Gateways

- Install an on-premises data gateway to transfer data quickly and securely between PowerApps and a data source that's not in the cloud
- With a gateway, you can connect to on-premises data over these connections:
  - SharePoint
  - SQL Server
  - Oracle
  - Informix
  - Filesystem
  - DB2



# On-premises data gateway

## One gateway for multiple cloud services and experiences



# Install a Gateway

The screenshot shows the Microsoft PowerApps portal interface. On the left, there's a dark sidebar with the following items:

- Home
- Apps
- Connections
- Flows
- Gateways** (this item is highlighted with a yellow oval)
- Notifications
- Common Data Service

The main content area has a light gray background. At the top, it says "Gateways" with a help icon. Below that, a message reads: "You don't have a gateway installed. To connect to your on-premises data, you can install one now." Underneath this message is a purple button labeled "Install a gateway". To the right of the message is a white rectangular box containing a purple cloud icon with an upward arrow and the text "On-Premises Data Gateway". Below this box, there's descriptive text: "Mobilize legacy on-premises systems into great apps that work across devices – without having to migrate any data." A "Learn more" link is also present. At the bottom of the white box is a black "Download" button.



# Install a Gateway

## On-premises data gateway installation

Reminder before you install.

⚠ The gateway works best when it is installed on a computer that is always on and not asleep.

The gateway will perform more slowly on a wireless network.

Next

Cancel

## On-premises data gateway installation

Getting ready to install the on-premises data gateway.

Install to

C:\Program Files\On-premises data gateway

I accept the [terms of use](#) and [privacy statement](#)

Install

Close



# Install a Gateway



Almost done.

Installation was successful!

Enter email to identify the account to use with this gateway.

Next, you need to sign in to register your gateway.



Sign in to your account X

Microsoft Azure

Work or school account

[Can't access your account?](#)

© 2017 Microsoft  
[Terms of use](#) [Privacy & Cookies](#)



# Install a Gateway

On-premises data gateway

You are signed in as zoe@ck2017.onmicrosoft.com and are ready to register the gateway.

New on-premises data gateway name  
MyOnPremGateway

Recovery key (8 character minimum)  
••••••••

(i) This key is needed to restore the gateway and can't be changed. Record it in a safe place.

Confirm recovery key  
••••••••|

We'll use this region to connect the gateway to cloud services: West US [Change Region](#)

<< Back    Configure    Cancel

On-premises data gateway

Status

Service Settings

Diagnostics

Network

**Logic Apps**  
West US    [Create a gateway resource in Azure](#)

**PowerApps and Flow**  
Default environment    (✓) Ready

**Power BI**  
Default environment    (✓) Ready

Close



# Install a Gateway

More information about on-premises gateways and installation can be found at:

<https://powerapps.microsoft.com/en-us/tutorials/gateway-reference/>



# Common Data Service

- **Common Data Service (CDS) is:**
  - A Microsoft Azure–based business application platform
  - Enables you to easily build and extend applications with their business data
  - Includes a Data Model
- **One of the biggest benefits is:**
  - It does all the heavy lifting of bringing your data together in one place for your users and applications



# 3 Pillars of Common Data Service

- **Be the focal point for your data**
  - Bring your data together into one place
- **Be a great place to build and manage apps**
  - Help app creators and developers of all skill levels to easily create apps with the data in the Common Data Service
- **It works with the rest of Microsoft**
  - Provide out-of-the-box integrations with technologies across Microsoft including Flow, Power BI, and Office



# Common Data Model Entities

- **Entities**
  - Standard set of entities comes with a common data model in the Common Data Service
  - Covers a range of common business needs
  - Grouped into categories



# Standard Entities

| Functional group                  | Description   |
|-----------------------------------|---|
| Customer Service                  | The Customer Service entities manage issues from your customers, including tracking, escalation, and documentation.   |
| Foundation                        | The Foundation entities contain information that is relevant to nearly every other entity group. This group contains entities such as Address and Currency.                                     |
| People, Organizations, and Groups | These entities encompass a rich set of people and organizations that you might interact with, including employees, contractors, donors, volunteers, fans, alumni, and families.                 |
| Purchasing                        | The Purchasing entities let you create purchasing solutions.  |
| Sales                             | The Sales entities let you create end-to-end sales solutions, from tracking leads and opportunities, to following through with contacts, accepting and delivering orders, and sending invoices. |



# Fields and Data Types

- **Default Fields**
  - Each entity contains a set of default fields
  - These fields cannot be changed or deleted
  - Some fields like *ContactID* are specific to an entity and files like *Created on date time* are common to all entities
- You can extend standard entities by adding fields:
  - Click **Add field** and specify the new field's properties



# Custom Entity

- **Create a Custom Entity**
  - When extending a standard entity is not enough
- **Data Types**
  - Fields in an entity have a data type
  - More than a dozen data types are available
- **Some representative types:**
  - *Basic types*: Text and Number
  - *More complex types*: Email and Phone
  - *Special types*: Lookup (for creating relationships) and Picklist (to hold a fixed set of values for a field)



# Working with Entities

Fields • Key • Relationships • Field groups • Data

- **Fields:** see fields and data types and add fields
- **Key:** the field that identifies each row in an entity (i.e. Contact ID for the Contact entity)
- **Relationships:** connections between related entities (i.e. Product and Product category)
- **Field groups:** used to control various behaviors (i.e. which fields to automatically show when you create an app screen in PowerApps)
- **Data:** browse sample data and your own data after it's imported



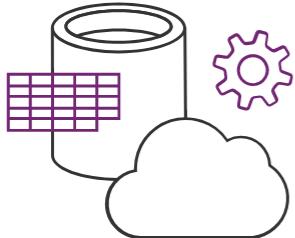
# Working with Entities

- **Open in Excel:** With the [PowerApps Office Add-in](#) installed you can explore and edit your data in Excel
- **Import data:** bring in data from Excel and CSV files
- **Export data:** export data to an Excel file
- **Export template:** export the structure of an entity to an Excel file so you can populate the file and import it back into the entity
- **Settings and Delete:** are not available for standard entities



# CDS Entities

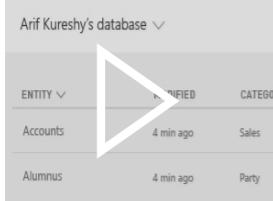
Entities 



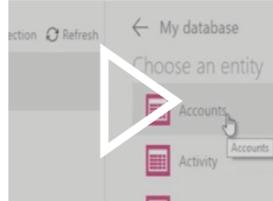
Create your own environment and database where you can build apps using the Common Data Service.

 Create database

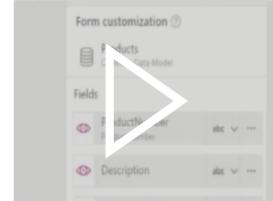
See how the Common Data Service works.



Working with entities



Creating an app



Create an app from scratch



# Building Flows

- **Use Microsoft Flow to create logic that performs one or more tasks when an event occurs in an app**
  - For example, configure a button:
    - *Creates an item in a SharePoint list*
    - *Sends an email*
    - *Meeting request is sent*
    - *A file is added to the cloud*
- **You can configure any control in an app to start the flow**
  - Continues to run even if you close PowerApps
- **Create Flows from scratch or from a template**

My flows

Team flows



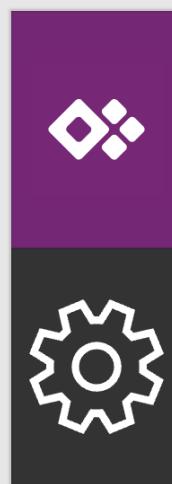
Create from blank



Create from template



# PowerApps Flow Templates



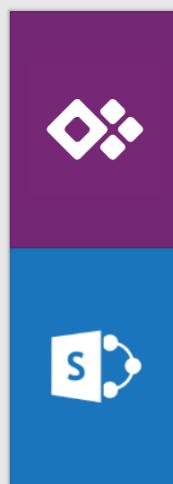
PowerApps button

By Microsoft  
Used 14425 times



Send approval  
email and follow up  
via email

By Microsoft  
Used 3240 times



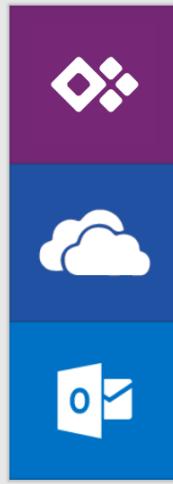
Upload a file to  
SharePoint from  
PowerApps

By Microsoft  
Used 1189 times



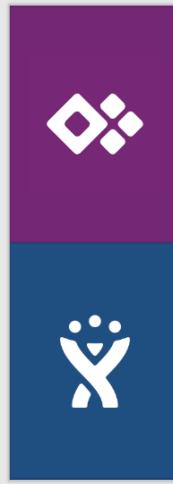
Add an item to  
SharePoint and  
send an email

By Microsoft  
Used 362 times



Request approval  
for new files in  
OneDrive for  
Business

By Microsoft  
Used 261 times



Create a new issue  
in JIRA from a  
PowerApp

By Carlos Javier Puerta  
Sanchez  
Used 72 times



# PowerApps and Flow

The screenshot shows the PowerApps interface. The left sidebar has icons for Home, Apps, Connections, Flows (which is selected), and Gateways. The main area has tabs for 'My flows' (selected) and 'Team flows'. A button to 'Create from blank' and another to 'Create from template' are at the top right. Below, it says 'The flows listed here start from an App. Click here to see all of your flows.' A table lists one flow:

| Name                                       | Last modified | Status | Action Buttons |
|--|---------------|--------|----------------|
| Upload a file to SharePoint from PowerApps | 1 day ago     | On     |                |

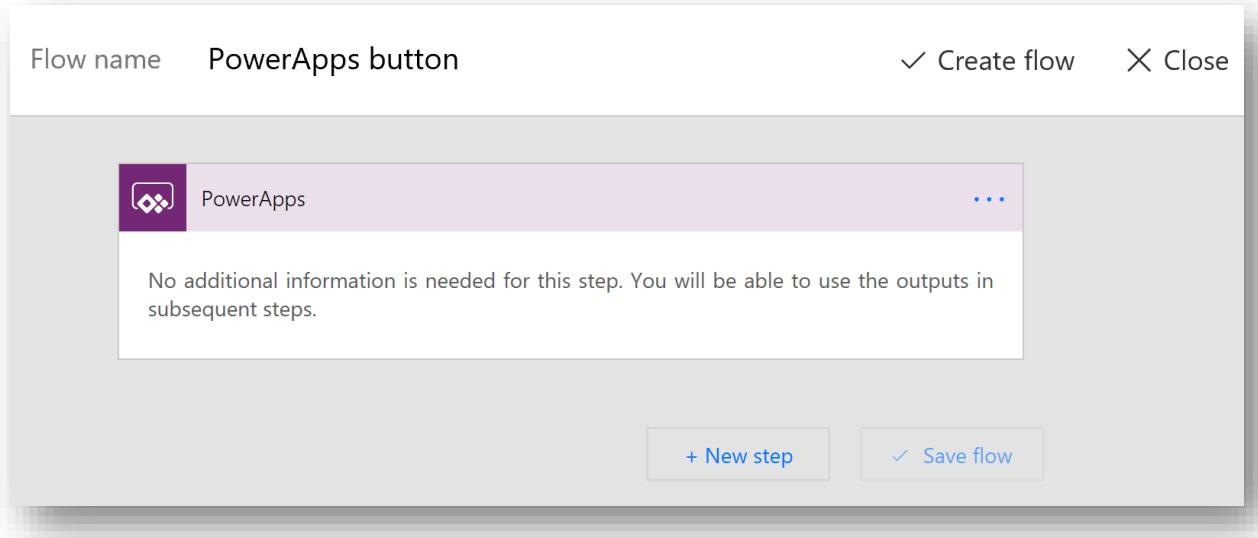
The screenshot shows the Flow interface. The left sidebar has icons for Flow (selected), My flows, Approvals, Templates, Connectors, and Learn. The main area has tabs for 'My flows' (selected) and 'Team flows'. A button to 'Create from blank' and another to 'Import' are at the top right. Below, it says 'The flows listed here start from an App. Click here to see all of your flows.' A table lists two flows:

| Name                                       | Last modified | Status | Action Buttons |
|--|---------------|--------|----------------|
| Start approval when a new item is added    | 1 day ago     | On     |                |
| Upload a file to SharePoint from PowerApps | 1 day ago     | On     |                |

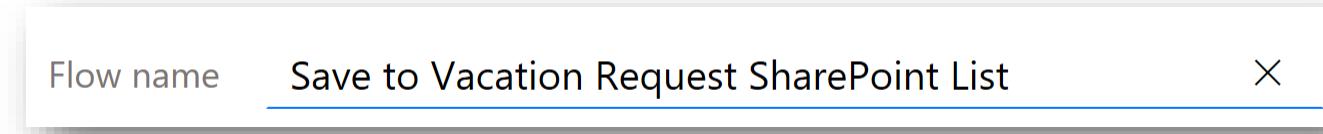


# PowerApps Portal

Create from blank <https://web.powerapps.com>



Rename Flow by clicking on Flow name text



# Flows Portal

Create from blank <https://flows.powerapps.com>

Start with one of these popular triggers



When a new email arrives



When an item is created



When a feed item is published



When a file is created



When a new tweet is posted



When a file is created



When a file is created



When a record is created

Or



Search hundreds of connectors and triggers



# Flow Visual Designer

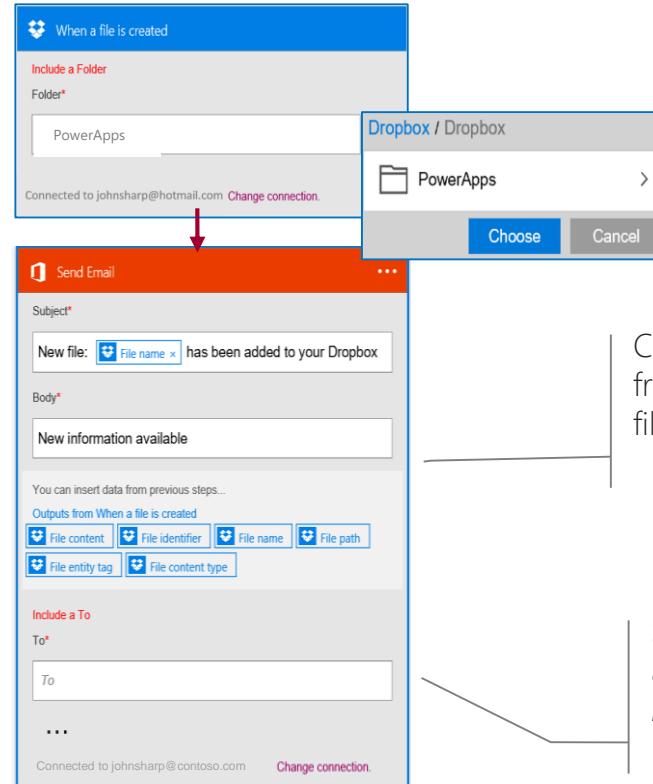
Sending an Exchange email when a new file is added in Dropbox

Authenticate to your Dropbox account.

Select folder to monitor

Authenticate to your O365 account.

Build email *Subject* and *Body*



Choose to include properties from previous step (Dropbox file) to improve relevance

Specify *Send To* email address and additional properties (optional): *From*, *CC*, *BCC* and *Importance*



# Building Blocks of Flow

- Flows whether created from a template or from scratch:
  - Will contain **building blocks** that work together in certain ways (much like a flowchart)
- Building blocks of Flow:
  - **Services** - sources and destinations of data in a flow
  - **Triggers** - events that start a flow
  - **Actions** - tasks accomplished by the flow
  - **Conditions** - allow for branching if/then logic in a flow
  - **Loops** - for iterating over actions more than once



# Triggers

- Triggers are events that initiate a Flow
- First entry in a Flow
- Linked to Connections
  - Services
  - Data sources

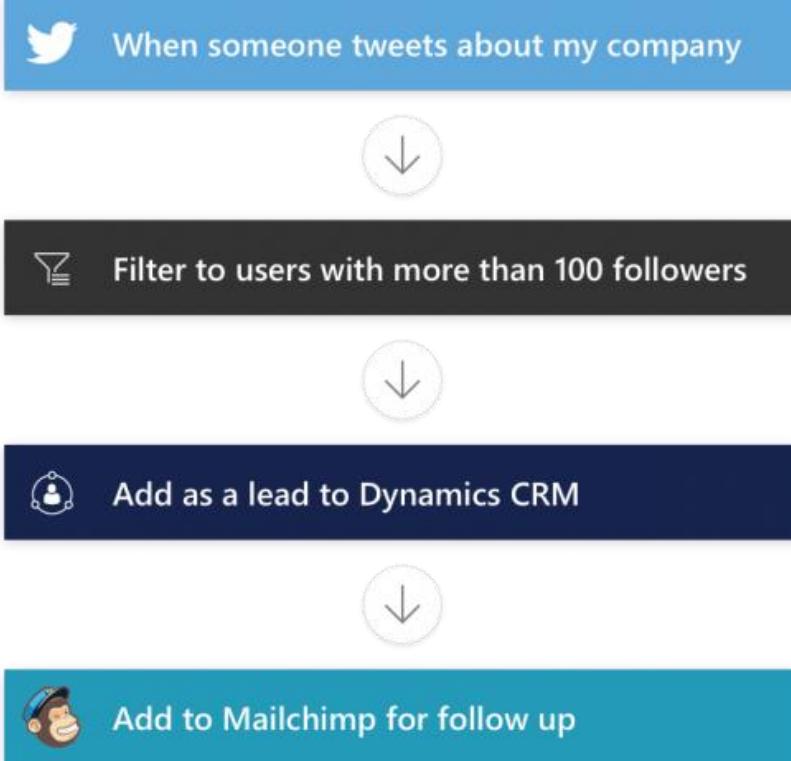


# Conditions, Actions and Loops

- **Conditions**
  - A **condition** is essentially your “**If Statement**”.
    - If this happens do this or if that happens do something else.
    - Specify that a flow performs one or more tasks only if a particular condition is true.
- **Actions**
  - An **action** is something that occurs as a result of the workflow.
  - Add multiple actions and advanced options to a flow.
- **Loops**
  - Add sequences or instances where actions need to be repeated based on a condition

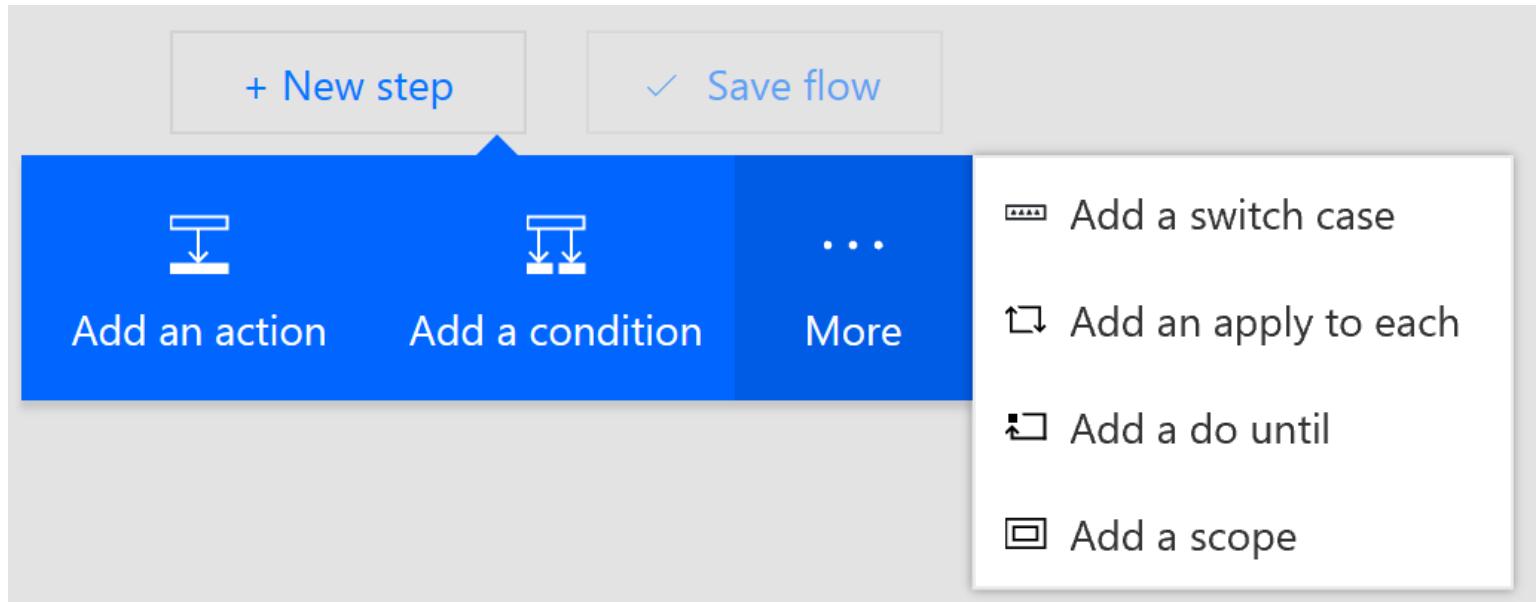


# Looping Flow Example



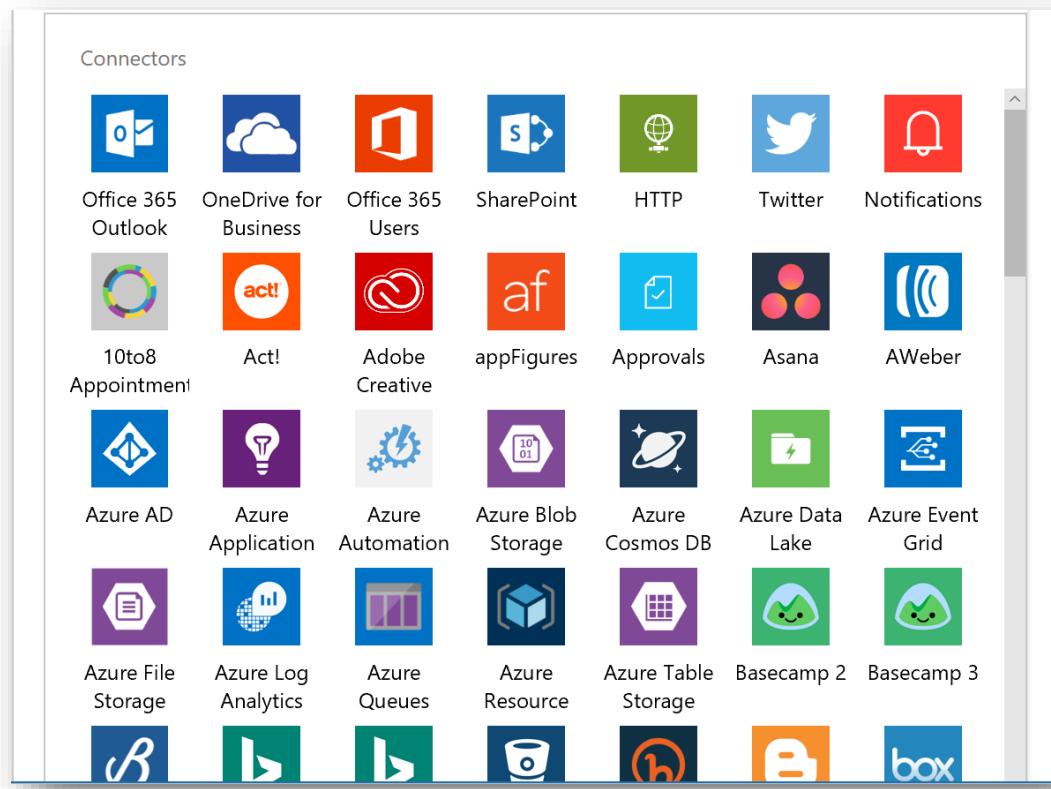
# Creating New Steps

- Action, Condition, More...



# Connectors

- Connect to services
- Surfaces Actions, Triggers
- Can manage ahead of time or when adding an action



# Actions

PowerApps

Choose an action

Search all connectors and actions

Connectors

- Office 365 Outlook
- OneDrive for Business
- Office 365 Users
- SharePoint
- HTTP
- Twitter
- Notifications

Actions (264)

- Office 365 Outlook Create contact
- Office 365 Outlook Create event (V1)
- Office 365 Outlook Create event (V2)
- Office 365 Outlook Send an email
- Office 365 Outlook Send approval email
- OneDrive for Business Create file

SharePoint

← Search all actions

Triggers (6) Actions (21)

- SharePoint Create file
- SharePoint Create item
- SharePoint Get entity values
- SharePoint Get file properties
- SharePoint Get files (properties only)
- SharePoint Get items
- SharePoint List folder
- SharePoint Copy file
- SharePoint Delete file

# Advanced Options

- Customize a flow by adding one or more advanced options and multiple actions for the same trigger.
  - For example, add an advanced option that sends an email message as high priority.
  - In addition to sending mail when an item is added to a SharePoint list, create a file in Dropbox that contains the same information.



# Recurring Flows

The screenshot shows the Microsoft Flow interface with a search bar at the top containing the text "recurrence". Below the search bar, there are sections for "Connectors", "Triggers (1)", and "Actions (2)".

**Connectors:** A list of connectors including Office 365, Schedule, and Outlook, with a "See more" link.

**Triggers (1):** A single trigger named "Schedule - Recurrence" is listed.

**Actions (2):** No actions are currently listed.

A detailed view of the "Schedule - Recurrence" trigger is shown in a modal window. The modal has a title "Recurrence" and contains the following configuration fields:

- \* Frequency: Day
- \* Interval: 1
- Time zone: (dropdown menu)
- Start time: Example: 2017-03-24T15:00:00Z

At the bottom of the modal, there is a link "Hide advanced options ^".



# Delayed Flows

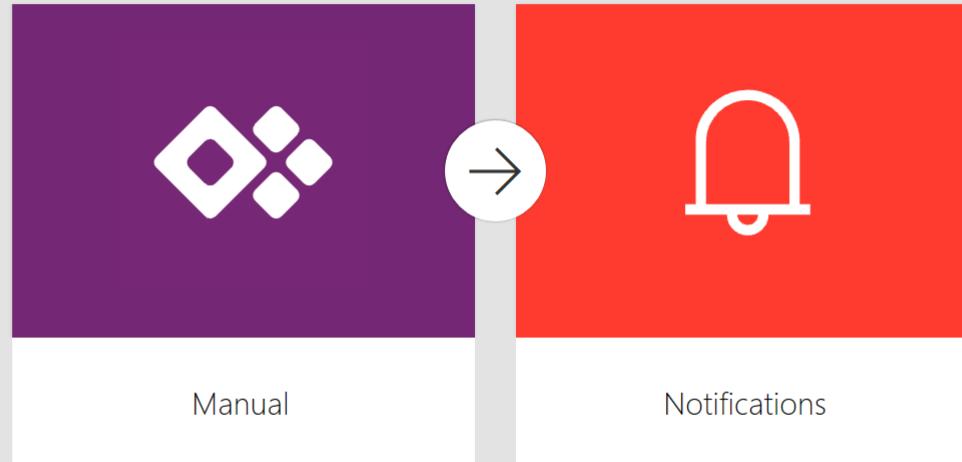
The screenshot shows a flow editor interface with three main components:

- Triggers (1) Actions (2)**: A sidebar showing two actions:
  - Schedule - Delay**: Represented by a blue square icon with a white alarm clock.
  - Schedule - Delay until**: Represented by a blue square icon with a white circular arrow.
- Delay**: A detailed configuration dialog for the first action.
  - \* Count**: A text input field with placeholder text "Specify the count of unit to delay".
  - \* Unit**: A dropdown menu currently set to "Minute". Other options include Day, Hour, Minute, and Second. There is also a "Enter custom value" field.
- Delay until**: A configuration dialog for the second action.
  - \* Timestamp**: A text input field with placeholder text "Example: 2016-07-11T14:45Z".



# Push Notifications

Send a push notification triggered by PowerApps



This flow sends a push notification to your Flow App through PowerApps, when triggered with a button tap. You will need an installation of the Flow App for this template.



# Notifications Connectors

Connectors

See more

Notifications PowerApps Notification

Triggers (0) Actions (3)

Notifications Send me a mobile notification

Notifications Send me an email notification

PowerApps Notification SendPushNotification

Send me a mobile notification

\* Text  
Create a notification message

Link  
Include a link in the notification

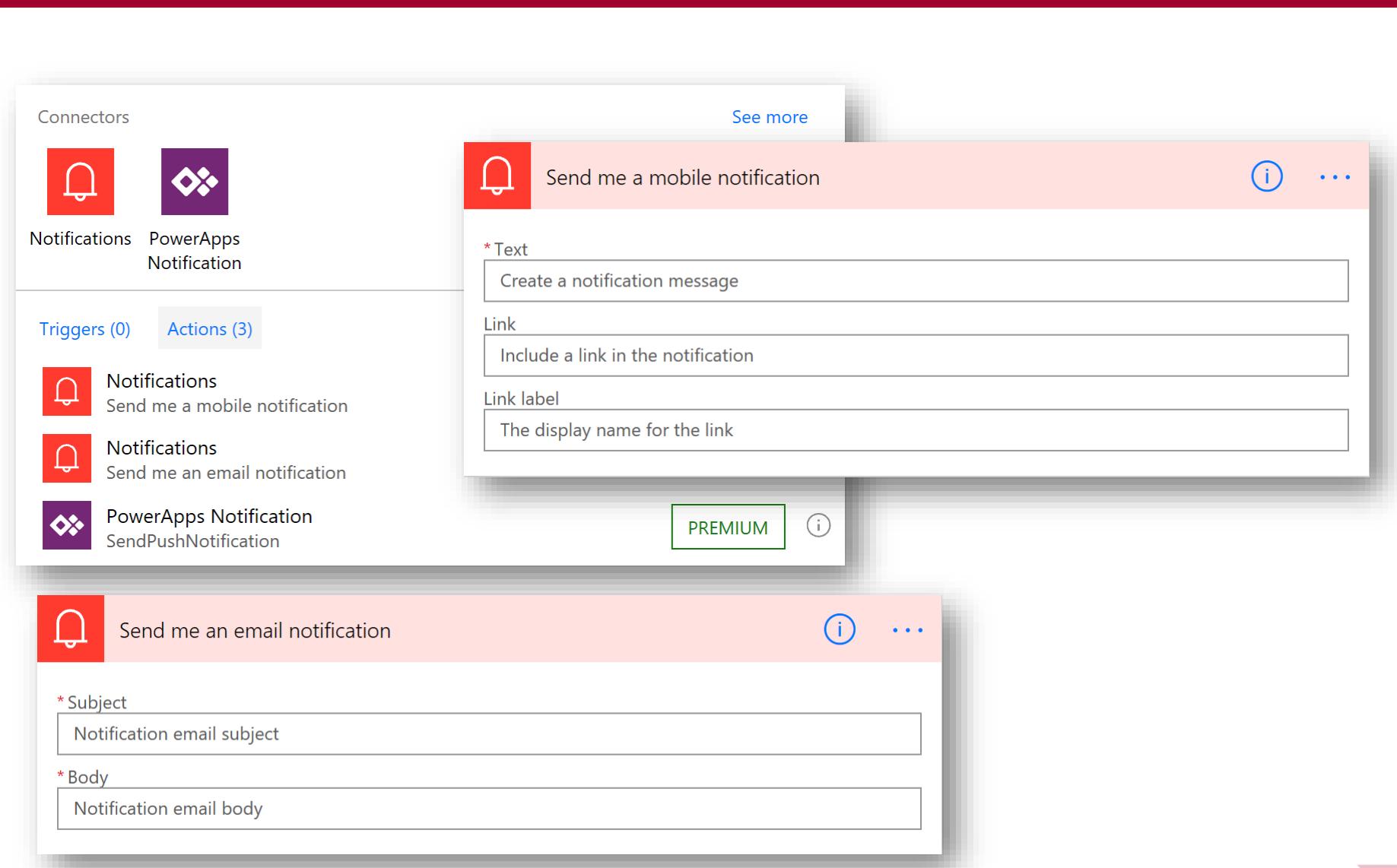
Link label  
The display name for the link

PREMIUM

Send me an email notification

\* Subject  
Notification email subject

\* Body  
Notification email body



# Flow Mobile App

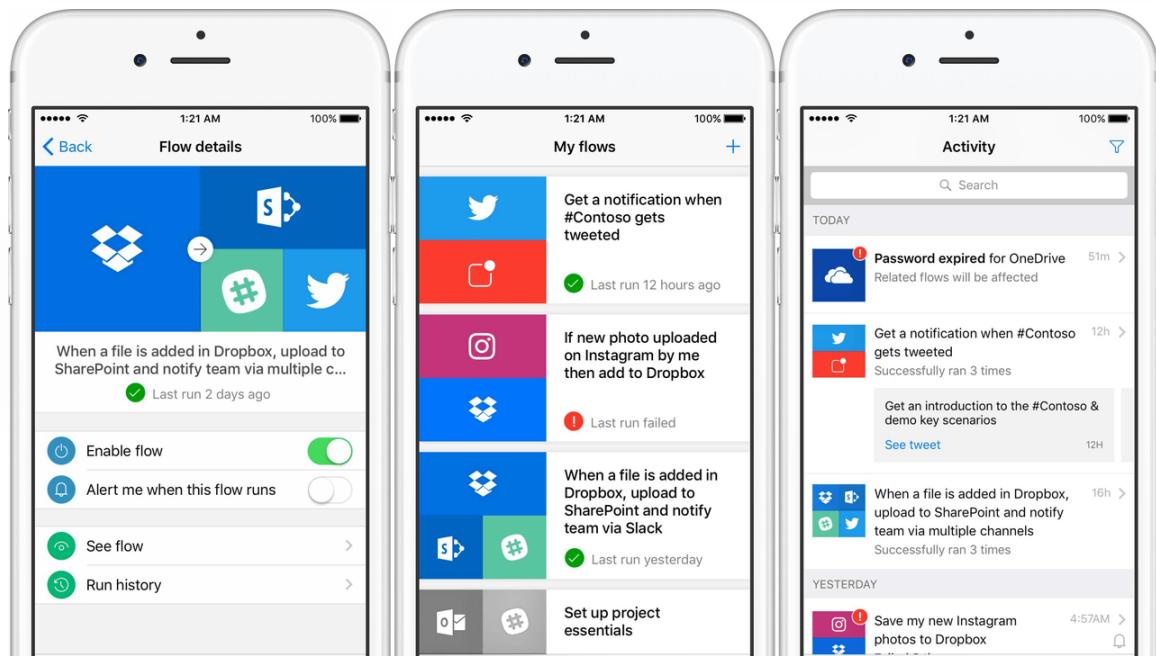
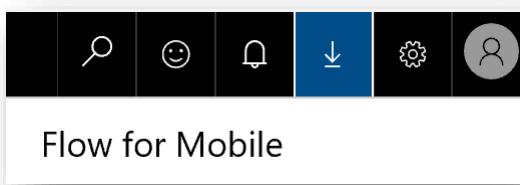
- Available for:
  - iOS
  - Android
  - Windows

Create automated workflows between your favorite apps and services to get notifications, synchronize files, collect data, and more — all from the palm of your hand

 Download from Apple Store

 Download from Google Store

 Download from Windows Store



# Application Lifecycle Manage (ALM)

- **Managing Power Apps**
  - Sharing PowerApps
  - PowerApp Versioning
  - Microsoft AppSource
- **Running PowerApps**
  - PowerApps Mobile
  - Web Browser
  - SharePoint List
- **PowerApps Admin Center and Flow Admin Center**
  - Environments
  - Data Loss Prevention Policies



# Sharing PowerApps

- **Apps can be shared with:**
  - Users or groups
  - Entire organization
- **There are 3 types of sharing options:**
  - Can Use
    - Users/groups can run the app but cannot share
  - Can Use and Share
    - Users/groups can run the app and share with others
  - Can Edit
    - Users/groups can run the app, customize it, and share the new version of the app



# PowerApps Versioning

- For apps saved to the cloud, PowerApps keeps a history of changes:
  - Accessible via <https://web.powerapps.com>
  - Restore, delete, and publish versions

Details • Share • Versions • Settings

PowerApps keeps a version history of your apps.

| VERSION | MODIFIED              | MODIFIED BY       | POWERAPPS RELEASE | PUBLISHED | COMMENT              |
|---------|-----------------------|-------------------|-------------------|-----------|----------------------|
| 3       | 8/11/2017 11:30:26 PM | MOD Administrator | 2.0.676           |           | Publish this version |
| 2       | 8/11/2017 11:24:14 PM | MOD Administrator | 2.0.676           |           | Restore  Delete      |
| 1       | 8/11/2017 1:04:45 AM  | MOD Administrator | 2.0.676           | Live      | Restore  Delete      |



# Microsoft AppSource

- **What is Microsoft AppSource?**
  - A destination to help business users find, try, and use line-of-business Software as a Service (SaaS) apps  
<https://appsource.microsoft.com>
  - Find top business solutions for your industry from Microsoft and select partners—including exclusive apps



Get the right app, right now

Find top business solutions for your industry from Microsoft and select partners—including exclusive apps.



Build on what you have

Add new capabilities to your existing business applications. From Dynamics to Power BI, you'll find solutions to enhance what you already use.



Start with confidence

Try any app for free before you buy it, then work with a trusted Microsoft partner to implement your new solution.



# Microsoft AppSource

- Get solutions tailored to your industry that work with the products you already use such as:
  - Dynamics 365, Office 365, Power BI, and various Cloud solutions

|  |  |   |  |   |
|--|--|---|--|---|
| <br><b>Google Analytics</b><br>By Microsoft<br>Power BI<br><br>Visualize data and gain insights from sites tracked by Google Analytics directly in Power BI | <br><b>Microsoft Dynamics NAV</b><br>By Microsoft<br>Power BI<br><br>Immediately gain access to business performance insights | <br><b>Microsoft Azure Enterprise</b><br>By Microsoft<br>Power BI<br><br>Analyze Azure consumption for your enterprise enrollment in Power BI | <br><b>Salesforce</b><br>By Microsoft<br>Power BI<br><br>Explore and monitor sales metrics for yourself and your team | <br><b>Bing</b><br>By Microsoft<br>Power BI<br><br>On-demand analytics of Internet search activity |
| Free   | Free   | Free  | Free   | Free  |
| <a href="#">Get it now</a>   | <a href="#">Get it now</a>   | <a href="#">Get it now</a>  | <a href="#">Get it now</a>   | <a href="#">Get it now</a>  |



# Microsoft AppSource

- Easily search for apps and filter based on category, industry and product

The screenshot shows the Microsoft AppSource search interface. At the top is a search bar with the placeholder "Search Microsoft AppSource" and a magnifying glass icon. Below the search bar are three sections for refining results:

- Refine by category**: A list of 15 categories with checkboxes:
  - Analytics
  - Artificial intelligence
  - Collaboration
  - Customer service
  - Finance
  - Human resources
  - IT + administration
  - Marketing
  - Operations + supply chain
  - Productivity
  - Sales
- Refine by industry**: A list of 11 industries with checkboxes:
  - Agriculture
  - Distribution
  - Financial services
  - Government
  - Healthcare + life sciences
  - Manufacturing
  - Professional services
  - Retail + consumer goods
  - Education
- Refine by product**: A list of four products with checkboxes and icons:
  - Cloud solutions (blue square icon)
  - Dynamics 365 (blue square icon with white arrow)
  - Power BI (yellow square icon with chart)
  - Office 365 (red square icon with white 'O')



# Microsoft AppSource

## Products

Office 365 x

[Reset filters](#)

### Apps

[Web apps](#)

[Cortana Intelligence](#)

### Add-ins

[Dynamics 365](#)

[Power BI](#)

[Dynamics NAV](#)

### Office 365 >

- Excel
- OneNote
- Outlook
- PowerPoint
- Project
- SharePoint
- Word

## Best apps of the year for Office 365



### Boomerang

By Baydin Inc.  
Outlook

The ultimate email productivity tool. Send later, track responses, schedule meetings, and more!

Free

[Get it now](#)



### officeatwork Template Chooser

By officeatwork 365  
Excel | PowerPoint | Project | Word

An easy way to manage, discover and use your templates in Office.

Additional purchase may be required

[Get it now](#)



### Pickit Free Images

By Pickit  
OneNote | PowerPoint | Word

Free HD images, icons & illustrations to bring your docs & presentations to life in a click.

Free

[Get it now](#)



### Smartsheet for Outlook

By Smartsheet.com, Inc.  
Outlook

Organize, track, and collaborate on projects or processes in real time right from your inbox.

Additional purchase may be required

[Get it now](#)

## Categories

- Analytics
- Artificial intelligence
- Collaboration
- Customer service
- Finance
- Human resources
- IT + administration

## Editor's Pick for Office 365



### Yesware Email Tracking

By Yesware Inc.



### Salesforce Lightning for Outlook



### DocuSign for Word

By DocuSign, Inc.



### iWRITER 365 - Templates for Word



# PowerApps Admin Center

- **PowerApps using in an organization**
  - Need to have established boundaries and policies
- **PowerApps Admin Center provides admins (*global tenant admins & environment admins*) the ability to:**
  - Create and manage environments
  - Databases
  - Data policies



# Flow Admin Center

- Place for global tenant & environment admins to manage:
  - **Data Loss Prevention (DLP) Policies**
  - **Data Integration**
    - Projects
    - Connection Sets
  - **Tenant**
    - User licenses
    - Quota

<https://admin.flow.microsoft.com>



# Managing Environments

- An **Environment** is a space to store, manage, and share your organization's:
  - Business data, PowerApps, and Flows
- Also serve as **containers** to separate apps that have:
  - Different roles
  - Security requirements
  - Target audiences
- How you choose to leverage environments depends on your organization and apps you are trying to build



# Environment Examples

- **Single environment:** Build apps in a single environment
- **Separate environments:** Create separate environments that group Test and Production versions of your apps
- **Separate environments by teams:** Create separate environments corresponding to specific teams or departments in your company.
  - Each containing the relevant data and apps for each audience
- **Separate environments by branches:** Create separate environments for different global branches of your organization



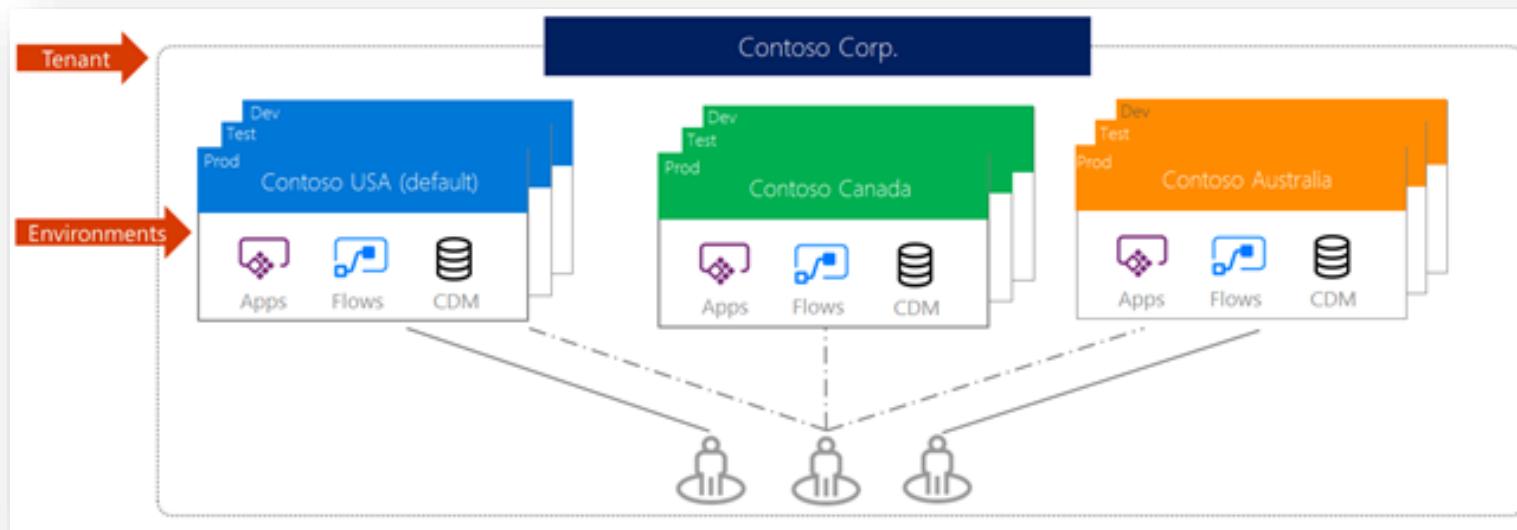
# Apps in Environments

- App in an environment is permitted to connect to data sources deployed in same environment which includes:
  - Connections, Gateways, Flows, and Common Data Service databases
- For example, in a scenario with two environments for Test and Production (named *Test* and *Prod*):
  - You create a Common Data Service database in each of the environments
    - An app in the ‘Test’ environment will only be permitted to connect to the ‘Test’ database
    - It cannot connect to the ‘Prod’ database and vice-versa



# CDS in Environments

- Every environment can have zero or one Common Data Service databases (provides storage for your apps)
  - The ability to create a CDS database for your environment depends on your purchased PowerApps license and your permission within the environment.



# Environment Roles and Permissions

- **Environment Admin role** - can perform all administrative actions on an environment including the following:
  - Add or remove a user or group from either Environment Admin or Environment Maker role
  - Provision Common Data Service database for environment
  - View and manage all resources created in environment
  - Set Data Loss Prevention policies
- **Environment Maker role** - can create new resources within environment including apps, connections, custom APIs, gateways, and flows using Microsoft Flow



# Data Loss Prevention Policies

- **Data is critical in an organization's success**
  - It needs to be readily available for decision-making
  - Needs to be protected so it's not shared with audiences that should not have access to the data
- **Data Loss Prevention (DLP) Policies**
  - PowerApps provides a way to protect your data by providing the ability to create and enforce policies
  - You can define which consumer services/connectors specific business data may be shared with



# Data Loss Prevention Policies

- **Benefits of a DLP policy**
  - Ensures data is managed in a uniform manner across the organization.
  - Prevents important business data from being accidentally published to services such as social media sites.
- **Example:** Prevent business data stored in SharePoint from being automatically published to a Twitter feed.
  - Create a DLP policy that blocks SharePoint data from being used as the source for tweets.



# Microsoft Flow roadmap - highlights



## H2 Calendar 2017

- Export/import Flows across environments
- “In-the-box” review Flows for SharePoint Online
- Use Flow for Content Publishing approvals for SharePoint Online Publishing sites
- Utilize people, choice, attachment and multi-values in your Flows for SharePoint Online
- In-context Flow support in Dynamics 365, OneDrive and Teams
- Usage analytics for Flows
- Flow activity included in Office 365 audit logs
- UK deployment



## Calendar 2018

- Use a HTML rich text editor to build emails
- Guides for debugging actions and conditions
- Government cloud deployment
- ALM – Sandbox environments
- ALM – Move or copy across environments
- Support for GDPR compliance
- Leverage device signals (e.g. Location) with Flows on mobile devices
- Include Flows in AppSource
- Azure functions integration

Roadmap is for general information purposes only and subject to change



# PowerApps roadmap - highlights



## H2 Calendar 2017

- Create and use embedded PowerApps as custom list forms in SharePoint Online
- Utilize attachments, multi-select fields, and images in your SharePoint Online driven PowerApps
- Easily embed PowerApps in SharePoint Online pages with web parts and within Power BI reports
- Use the Rules explorer for easier expression authoring
- Tenant-wide admin resource management, app usage analytics, and package export/import across environments
- Easily connect to Azure App Services & APIM



## Calendar 2018

- Use server-side business logic for validation rules and easy defaulting
- Create model-driven apps
- Support for GDPR compliance
- Government cloud deployment
- SQL - Support for stored procs and views
- Enhanced application lifecycle management with sandbox environments



# Control Input and Output Properties

No 2-way binding in PowerApps – one property to set and the other to fetch

## Input and Output Properties

- Input only – Value based on a formula, can not use in other formulas
  - Does not show up in an IntelliSense
  - `TextInputBox.Default` (initial value)
- Input/Output – Value based on a formula, can use in other formulas
  - Shows up everywhere
  - `TextInputBox.BackgroundColor`
- Output only – No formula, can only use in other formulas
  - Does not show up in formula bar or advanced pane
  - `TextInputBox.Text` (value with user input)

Text input

Text input



# Primitive Data Types

- Number – **123.456**
- Text – “**Hello World**”
- Boolean – **True** or **False**
- DateTime – **1/12/2017 12:00 PM**
- Date – **1/12/2017**
- All can hold Blank (Null) values
  - Test with **IsBlank** function
  - Create with **Blank** function



# Compound Data Types

- Record – { FirstName: “Fred”, LastName: “Jones” }
- Table
  - Table( { FirstName: “Jane”, LastName: “Doe” },  
{ FirstName: “John”, LastName: “Smith” } )
  - [ 123, 456, 789 ] (one column named “Value”)
- Records and Tables can be nested
  - Table can contain records of tables of tables of records ...
  - Record can contain tables of records of tables of records ...
- There are no arrays, use a Table instead

## Table Functions

- Tables are a value
  - Passed as arguments
  - Returned from functions
  - Just like a number or text string
- Each of the following functions returns a table
  - Filtering: **Filter, Search**
  - Sorting: **Sort, SortByColumns**
  - Shaping: **AddColumns, DropColumns, RenameColumns, ShowColumns**
  - Grouping: **GroupBy, Ungroup**



# State Variables

- Collections
  - App scope
  - Tables
  - Primary functions: **Collect**, **Clear**, **ClearCollect**
  - Can be stored on the local device with **SaveData** & **LoadData**
  - Can be used as a global variable, but can be awkward to use as it is a table
- Context variables
  - Local screen scope
  - Primitive, record, or table
  - Primary functions: **UpdateContext**, **Navigate**
- All viewable from the File menu “Collections” and “Variables” tabs
- *hint*



# Demo

- XX

```
ClearCollect( DataStructures, Table( { Name: { First: "John", Last: "Doe" }, Eyes: "Blue", Height: "5' 10\"", Addresses: Table( {  
    Type: "Home", Street: "1 Clover Ln", City: "Austin", State: "TX" }, { Type: "Work", Street: "2 Main St", City: "Dallas", State: "TX" } )^  
  ),  
  { Name: { First: "Jane", Last: "Doe" }, Eyes: "Hazel" } ) );  
UpdateContext( { VarText: "Hello World", VarNumber: 1234, VarRecord: { FirstName: "Sue", LastName: "Smith" }, VarTable: [ 1, 2, 3, 4 ] } )
```

I



# Events and State Changes

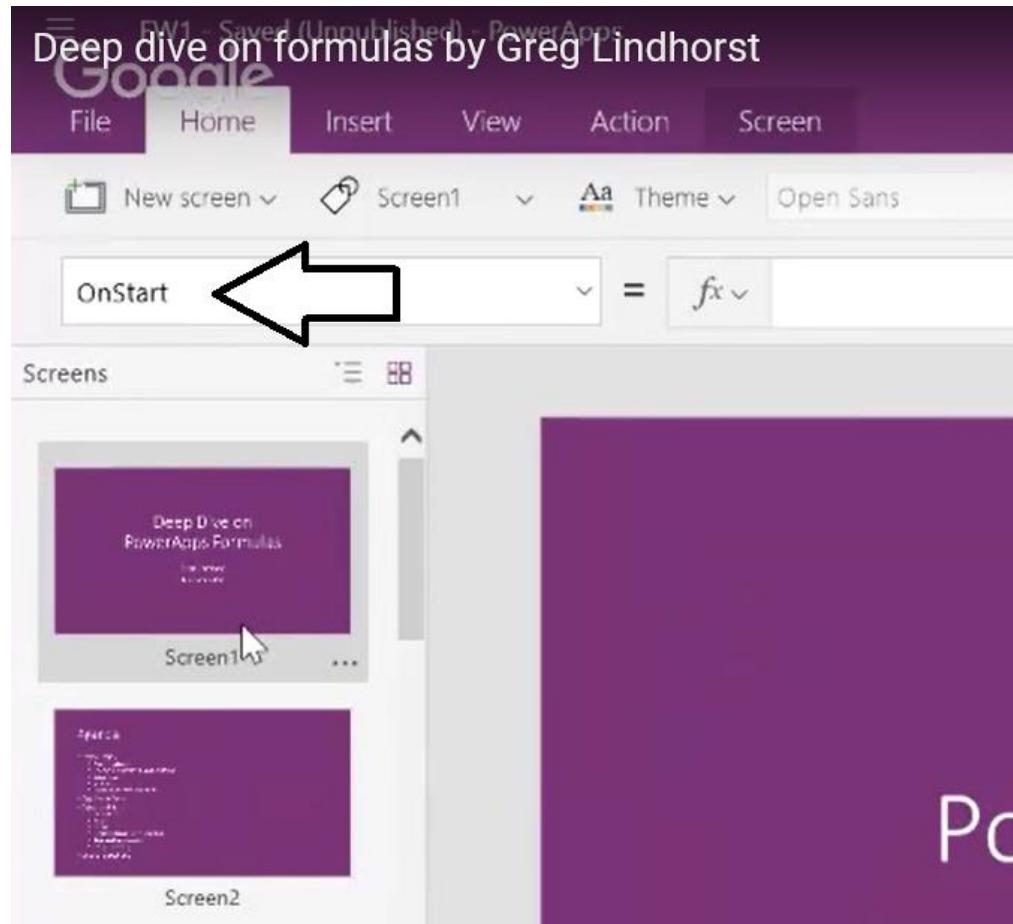
- Not Excel
  - Excel doesn't have Buttons, but apps do
  - Excel can't change state (the values of cells), but apps do
- Some formulas define imperative logic
  - OnSelect
  - OnVisible
  - OnStart
  - ...
- Can involve multiple steps with chaining operator ; (or :: depending on locale)
- Used to take actions
  - Set a context variable's value
  - Add to a collection
  - Change screens
  - Submit data to a server
  - Call a connector API
  - ...



Demo



# App OnStart – Only on First Form



# Same in PowerApps as in Excel

|              |                 |             |           |           |               |               |                |
|--------------|-----------------|-------------|-----------|-----------|---------------|---------------|----------------|
| Abs          | Collect         | DateValue   | ForAll    | LookUp    | Proper        | Set           | TimeValue      |
| Acceleration | Color           | Day         | GroupBy   | Lower     | Radians       | ShowColumns   | TimeZoneOffset |
| Acos         | ColorFade       | Defaults    | HashTags  | Max       | Rand          | Shuffle       | Today          |
| Acot         | ColorValue      | Degrees     | Hour      | Mid       | Refresh       | Sin           | Trim           |
| AddColumns   | Compass         | Disable     | If        | Min       | Remove        | Sort          | TrimEnds       |
| And          | Concat          | Distinct    | IsBlank   | Minute    | Removelf      | SortByColumns | Ungroup        |
| App          | Concatenate     | Download    | IsEmpty   | Mod       | RenameColumns | Split         | Update         |
| Asin         | Connection      | DropColumns | IsMatch   | Month     | Replace       | Sqrt          | UpdateContext  |
| Atan         | Count           | EditForm    | IsNumeric | Navigate  | Reset         | StartsWith    | UpdateIf       |
| Atan2        | Cos             | Enable      | IsToday   | NewForm   | ResetForm     | StdevP        | Upper          |
| Average      | Cot             | EndsWith    | Language  | Not       | Revert        | Substitute    | User           |
| Back         | CountA          | Errors      | Last      | Now       | RGBA          | SubmitForm    | Validate       |
| Blank        | CountIf         | EncodeUrl   | LastN     | Or        | Right         | Sum           | Value          |
| Calendar     | CountRows       | Exit        | Launch    | Param     | Round         | Switch        | VarP           |
| Char         | DataSourcesInfo | Exp         | Left      | Parent    | RoundDown     | Table         | ViewForm       |
| Clear        | Date            | Filter      | Len       | Patch     | RoundUp       | Tan           | Weekday        |
| ClearCollect | DateAdd         | Find        | Ln        | Pi        | SaveData      | Text          |                |
| Clock        | DateDiff        | First       | LoadData  | PlainText | Search        | ThisItem      |                |
| Coalesce     | DateTimeValue   | FirstN      | Location  | Power     | Second        | Time          |                |

Excel  
Formula Language

DAX - Data Analysis  
Expression Language

PowerApps  
Formula Language



# Imperative Functions

|              |                |             |           |           |               |               |                |
|--------------|----------------|-------------|-----------|-----------|---------------|---------------|----------------|
| Abs          | Collect        | DateValue   | ForAll    | LookUp    | Proper        | Set           | TimeValue      |
| Acceleration | Color          | Day         | GroupBy   | Lower     | Radians       | ShowColumns   | TimeZoneOffset |
| Acos         | ColorFade      | Defaults    | HashTags  | Max       | Rand          | Shuffle       | Today          |
| Acot         | ColorValue     | Degrees     | Hour      | Mid       | Refresh       | Sin           | Trim           |
| AddColumns   | Compass        | Disable     | If        | Min       | Remove        | Sort          | TrimEnds       |
| And          | Concat         | Distinct    | IsBlank   | Minute    | RemoveIf      | SortByColumns | Ungroup        |
| App          | Concatenate    | Download    | IsEmpty   | Mod       | RenameColumns | Split         | Update         |
| Asin         | Connection     | DropColumns | IsMatch   | Month     | Replace       | Sqrt          | UpdateContext  |
| Atan         | Count          | EditForm    | IsNumeric | Navigate  | Reset         | StartsWith    | UpdateIf       |
| Atan2        | Cos            | Enable      | IsToday   | NewForm   | ResetForm     | StdevP        | Upper          |
| Average      | Cot            | EndsWith    | Language  | Not       | Revert        | Substitute    | User           |
| Back         | CountA         | Errors      | Last      | Now       | RGBA          | SubmitForm    | Validate       |
| Blank        | CountIf        | EncodeUrl   | LastN     | Or        | Right         | Sum           | Value          |
| Calendar     | CountRows      | Exit        | Launch    | Param     | Round         | Switch        | VarP           |
| Char         | DataSourceInfo | Exp         | Left      | Parent    | RoundDown     | Table         | ViewForm       |
| Clear        | Date           | Filter      | Len       | Patch     | RoundUp       | Tan           | Weekday        |
| ClearCollect | DateAdd        | Find        | Ln        | Pi        | SaveData      | Text          |                |
| Clock        | DateDiff       | First       | LoadData  | PlainText | Search        | ThisItem      |                |
| Coalesce     | DateTimeValue  | FirstN      | Location  | Power     | Second        | Time          |                |



# What is Flow?

- Service for automating workflows across other services
  - Designed by Microsoft for business users more than developers
- What can you do with Flow?
  - Get notifications
  - Copy files
  - Collect data
  - Automate approvals



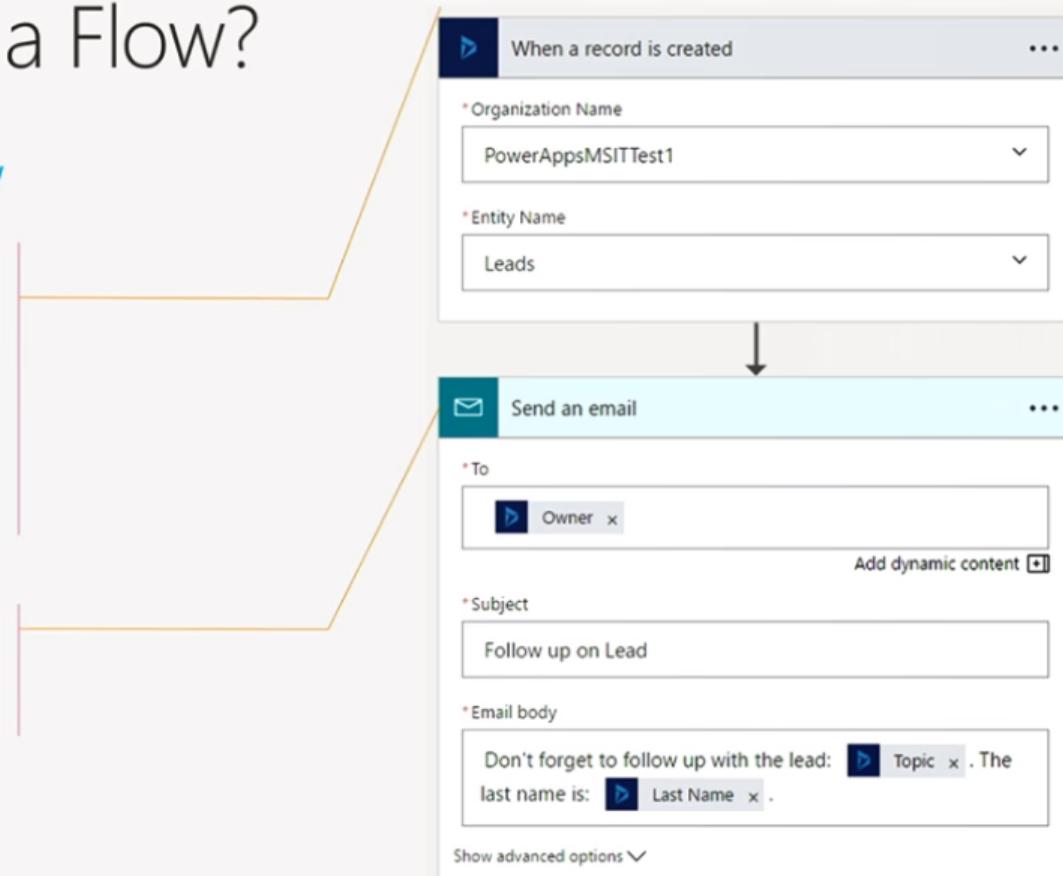
# What makes up a Flow?

## Example: Notification Flow

Trigger – the event that kicks off the flow:

- manually
- on a schedule
- on an event in the cloud

Actions – what the flow does  
*Can use data from the trigger*



# Connect to cloud and on premises data

Over 170 out of the box connectors with new connectors every week

Use Custom Connectors to connect to any custom REST endpoint

Connect to on-premises data using gateways



SharePoint Online



Salesforce



Dynamics CRM Online



Google Drive



Dropbox



Excel



OneDrive



Slack



Twitter



Office 365



Microsoft Azure



Custom APIs



# Templates help users to get started

|   |  |  |  |   |   |   |   |
|---|--|--|--|---|---|---|---|
| <p>Create a flow from blank</p>    | <p> Copy Dynamics CRM Accounts to Common Data Service customer organizations</p> <p>By Microsoft<br/>Used 227 times</p> | <p> Copy Dynamics CRM Accounts to customers in Dynamics 365 for Operations</p> <p>By Microsoft<br/>Used 187 times</p> | <p> When an object is created in Dynamics CRM, create a list item</p> <p>By Microsoft<br/>Used 184 times</p>    | <p> Create Dynamics Leads based on Tweets with a button tap</p> <p>By Microsoft<br/>Used 31 times</p>   | <p> Add a Dynamics Case after email approval</p> <p>By Microsoft<br/>Used 21 times</p>                 | <p> Copy new Dynamics CRM Cases to the Common Data Service</p> <p>By Microsoft<br/>Used 28 times</p> | <p> Create a Wunderlist Task for New CRM Leads</p> <p>By James Turner<br/>Used 25 times</p>    |
| <p> When a new opportunity is created post to Yammer</p> <p>By Richard Bosse<br/>Used 122 times</p>     | <p> Create Dynamics CRM Leads from an Excel table</p> <p>By Microsoft<br/>Used 121 times</p>                            | <p> Create Dynamics Leads based on Tweets</p> <p>By Microsoft<br/>Used 118 times</p>                                  | <p> Create a new record in Dynamics CRM when a new list item is added</p> <p>By Microsoft<br/>Used 95 times</p> | <p> Send an mail from your SMTP server on new opportunities</p> <p>By Kerem Israr<br/>Used 28 times</p> | <p> Copy new Dynamics CRM Leads to the Common Data Service</p> <p>By Microsoft<br/>Used 25 times</p>   | <p> Attach OneDrive files about Dynamics Contacts as notes</p> <p>By Microsoft</p>                   | <p> Create CRM Account for new Azure Queue Messages</p> <p>By Marius Pedersen</p>              |
| <p> Copy new Dynamics CRM Contacts to the Common Data Service</p> <p>By Microsoft<br/>Used 31 times</p> | <p> Create a SharePoint item based on a Dynamics CRM record</p> <p>By Henry Schrein<br/>Used 79 times</p>               | <p> Create Dynamics notes for Emails</p> <p>By Microsoft<br/>Used 81 times</p>  | <p> Move rows from and Excel table to Dynamics AX Online Vendors</p> <p>By Microsoft<br/>Used 58 times</p>      | <p> Send a Push Notification for new Sales Orders</p> <p>By Isaac Dongbatu</p>                          | <p> Copy Teams from the Common Data Service to Operations Customer Groups</p> <p>By Michael Barron</p> | <p> Copy Salesforce Accounts to Dynamics AX Online Customers</p> <p>By Microsoft</p>                 | <p> When a new lead is created in Dynamics CRM, add it to a Smartsheet</p> <p>By Microsoft</p> |
| <p> Copy from Dynamics CRM to the Common Data Service and Operations</p> <p>By Microsoft</p>          | <p> CRM Opportunity Win to Create a New Project</p> <p>By Microsoft</p>   | <p> Create Dynamics CRM Leads from a Google Sheet</p> <p>By Microsoft</p>   | <p> On approval create a Dynamics CRM entity for items in a SharePoint list</p> <p>By Microsoft</p>           | <p> Create a case in Dynamics from Freshdesk</p> <p>By Microsoft</p>                                  | <p> Low stock warning</p> <p>By Microsoft</p>  | <p> Social Media Promotions</p> <p>By Microsoft</p>  | <p> Add notes to a Dynamics CRM contact</p> <p>By Microsoft</p>                              |



# Build flows from scratch

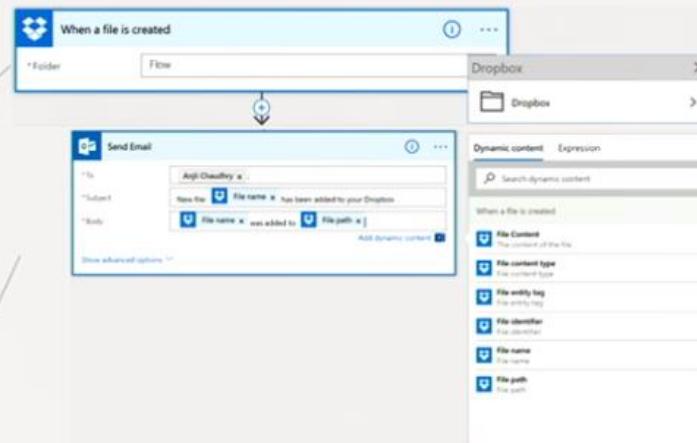
Sending an Exchange email when a new file is added in Dropbox

Authenticate to your Dropbox account.

Select folder to monitor

Authenticate to your O365 account.

Build email *Subject* and *Body*



Specify *Send To* email address and additional properties (optional): *From*, *CC*, *BCC* and *Importance*

Choose to include properties from previous step (Dropbox file) to improve relevance



# Manage your flow

## Flow Details

- View and edit title and description
- Turn flow on and off
- Clone it
- View run history

The screenshot shows the Microsoft Flow management interface. At the top, there's a navigation bar with 'Flows', 'Approvals', 'Templates', 'Connectors', 'Learn', a search bar 'Search templates ...', and a 'More' button. On the far right, there's a 'This flow is' toggle switch set to 'On'. Below the navigation bar, the main content area displays a flow card for a flow named '2 Run sentiment analysis on tweets and push results to a Power BI dataset'. The card includes icons for Twitter, a green circle, and a yellow bar chart, along with a link to 'See analytics'. To the right of the card, there's a 'CONNECTIONS' section listing 'Test Analysis' (anjiic@microsoft.com), 'anjiic@microsoft.com' (twice), and 'anjili' (anjiic@microsoft.com). Further down, the 'RUN HISTORY' section shows ten recent runs, all of which have 'Succeeded'. The last run was 37 seconds ago. The 'OWNERS' section shows 'Anjili Chaudhry' (anjiic@microsoft.com) as the owner, with an option to 'Add another owner'.

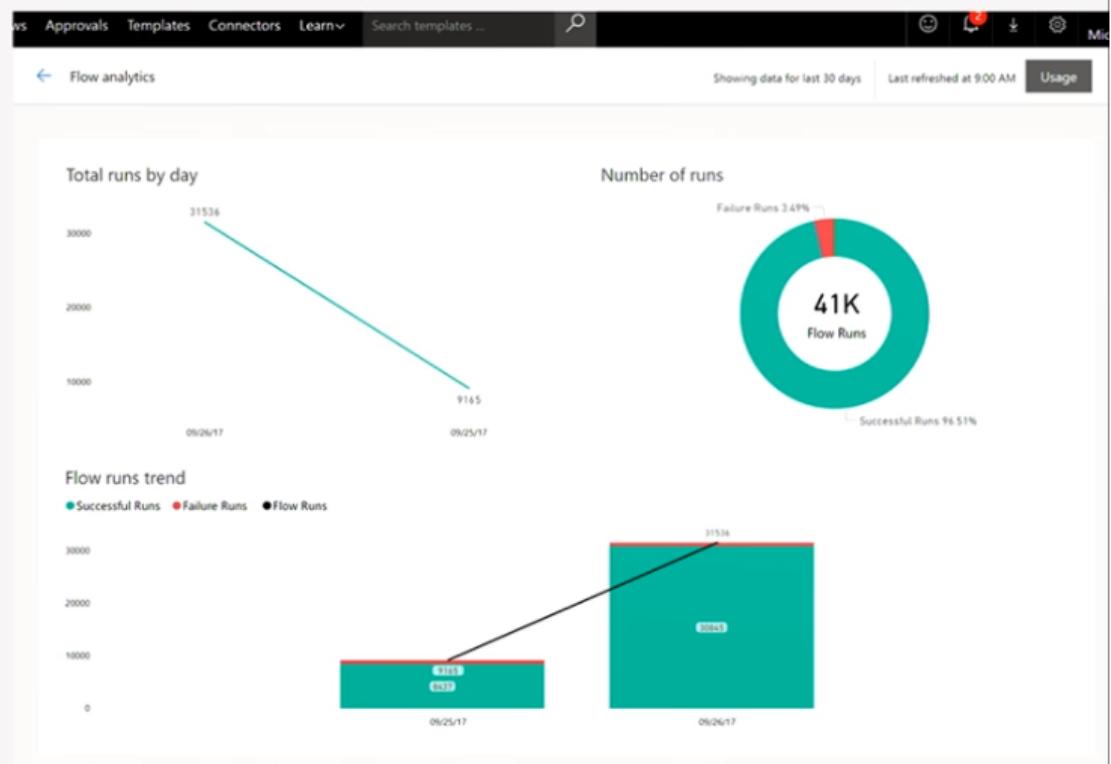
# Manage your flow

## Flow Details

- View and edit title and description
- Turn flow on and off
- Clone it
- View run history

## Flow Analytics

- See number of runs and trends



# Collaborate with your co-workers

## Co-authoring

- Many authors can edit a flow, not just a single owner
- If the original author leaves the company the flow continues to function

The screenshot shows the 'Owners' section of a Microsoft Flow configuration page. At the top, there is a navigation bar with links for 'My flows', 'Approvals', 'Templates', 'Connectors', 'Learn', and a search bar. Below the navigation is a header with the text 'When an item is created -> Start an approval|Condition,Update item,Send em...'. On the right side of the header are several small icons: a smiley face, a gear, a red circle with a number '1', a downward arrow, and a gear.

**Owners**

Adding another owner allows others to edit, update and delete this flow. All owners can also access the run history and add or remove other owners.

[Learn more](#)

**Add another owner**

Sunay

Sunay Vaishnav  
sunayv@microsoft.com

Anjali Chaudhry  
anjlc@microsoft.com

Jon Levesque  
jolevesq@microsoft.com

**Embedded connections**

Everyone listed as an owner will have access to all these connections and will only be able to use them in this flow.

[Learn more](#)

**Embedded connections used in this flow:**

| Approvals           | anjlc@microsoft.com | X |
|---------------------|---------------------|---|
| anjlc@microsoft.com | anjlc@microsoft.com | X |
| Mail                | anjlc@microsoft.com | X |
| anjlc@microsoft.com | anjlc@microsoft.com | X |



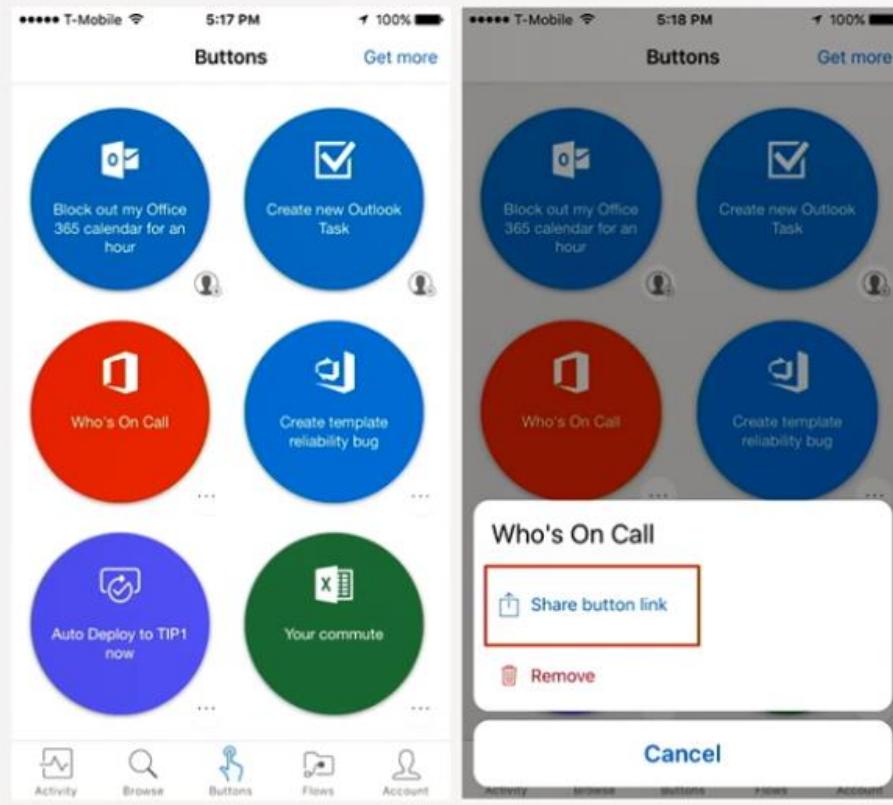
# Create Buttons and share them

## What are buttons?

- Easy way to trigger on demand flows with a push of a button!
- Buttons widget on phone and in the Flow app
- Physical buttons with Flic and Bttn

## Sharing buttons

- Make buttons available to your co-workers to automate their tasks



# Admin center

## Environment administration

- Create and manage your organization's environments
- Lifecycle management of flows
- Manage roles and role membership
- View license and quota information

## Data Loss Protection

- Control how data flows within your organization
- Set policies for classification of services

## Local agility, central control

- Allow teams control over their environment and still have centralized control

Flows and apps that combine services from your business data and Non-business data groups will be automatically blocked. Services may only appear in one data group but can be moved at any time.

### Business data only



### Non-critical, mixed data (Default)



### Non-business data



✓ Create data policy

# Build a connector to feature on Flow OR custom to your company

Connect to your own services

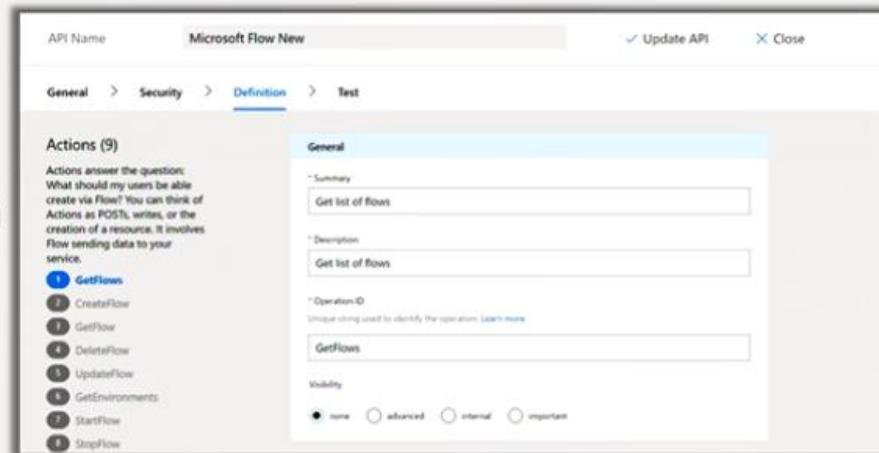
- Any REST end-point

Custom connectors within an organization

- Scoped to a company

Partner with Flow and PowerApps to build  
first class connectors for your services

- Showcase your hero scenarios using  
templates and explore co-marketing  
opportunities with Microsoft



# What's happened in the last six months?

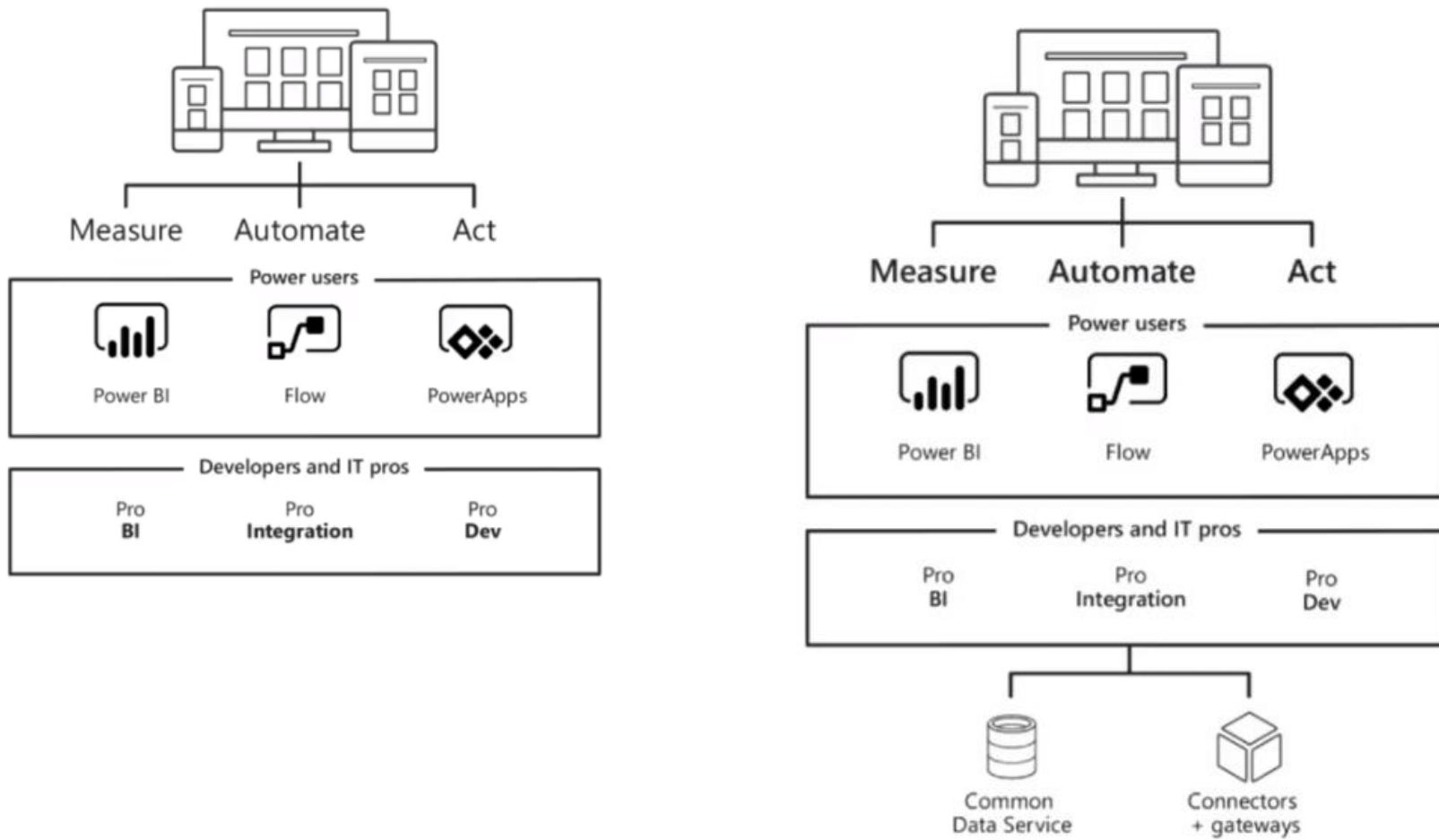
## Flow Q2 2017 Updates

- New modernized approval experiences
- Trigger flows from physical buttons
- 'Diet' designer and new getting started experiences
- Team flows general availability
- Trigger buttons from widgets on iOS or Android
- Control-flow improvements like parallels and error handling
- Manage flows, licenses, quotas in the Admin center
- SharePoint Document library integration
- Windows Phone Support
- Convert the columns for tables or lists
- Build and submit your own connectors
- New Community Plan and Office 365 F1 Plan
- Dozens of new connectors

## Flow Q3 2017 Updates

- Import and Export
- Use expressions in actions
- Grow up to Azure Logic apps
- OneDrive for Business integration
- More SharePoint column/file metadata support
- Reshare buttons and collect lists, emails or files
- Download full run history to Excel
- Compliance certifications (e.g. ISO, SOC, HIPAA etc...)
- Flow collections gallery
- Power BI analytics integration (First release only)
- Office 365 security & compliance center audit logs
- Custom connectors for on-premises APIs
- Dozens of new connectors





# Triggers

Run manually based on user action

- From dedicated Flow button
- From other apps – PowerApps, SharePoint...

Run on a schedule

- From every minute to 4am on Saturdays

Run based on an event that happens  
in a cloud or on-prem service

The screenshot shows the Microsoft Flow interface for selecting triggers. At the top, there is a search bar labeled "Search all services and triggers". Below the search bar, there are two main sections: "SERVICES" and "TRIGGERS (115)".

**SERVICES:**

- Flow button for mobile
- Office 365 Outlook
- SharePoint
- RSS
- Schedule
- Twitter
- Outlook.com
- Dropbox
- OneDrive for Business
- OneDrive

**SEE MORE**

**TRIGGERS (115):**

- appFigures - When there is a new review
- Asana - When a project is created
- Azure Queues - When there are messages

**SEE MORE**



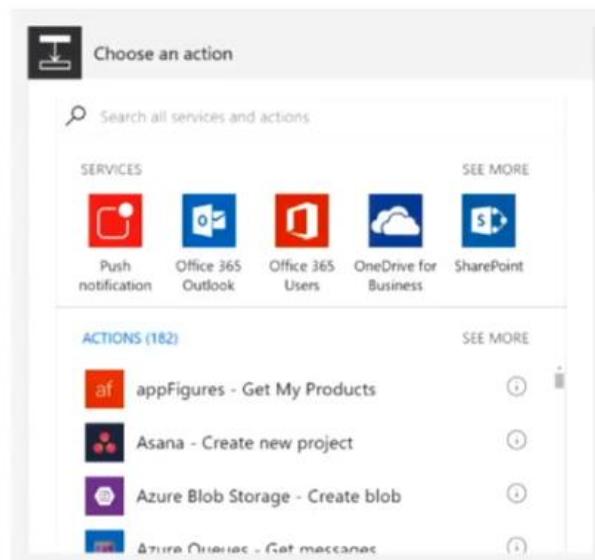
# Actions

Any number of steps that run after the trigger

Connect to any cloud service or via the on-prem data gateway

Connect to any HTTP endpoints if there's no out-of-box connector

Some actions may not even use connections – just manipulate data locally

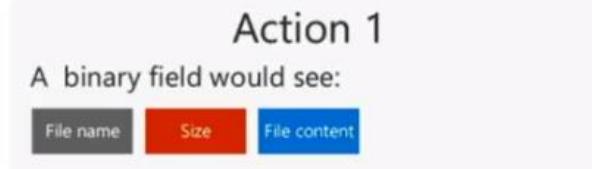
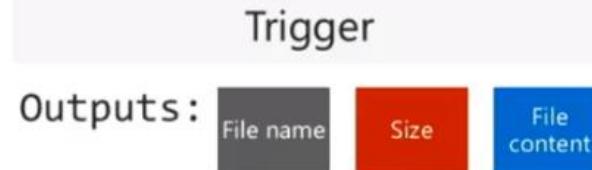


# Flowing data

Data 'flows' from each step and is available for all later steps

Use  **Add Dynamic Content** to select outputs from previous steps

Certain outputs show up based on the types of the inputs and outputs



# Using expressions

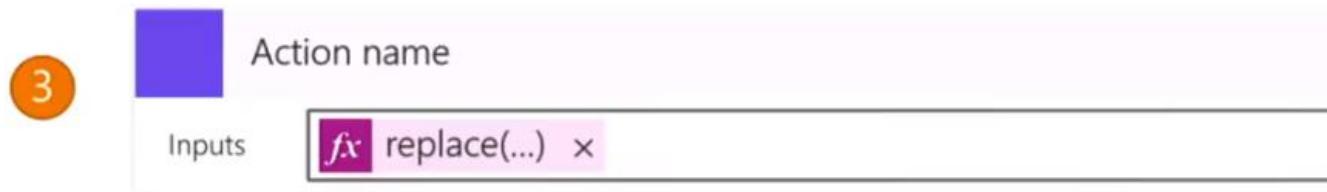
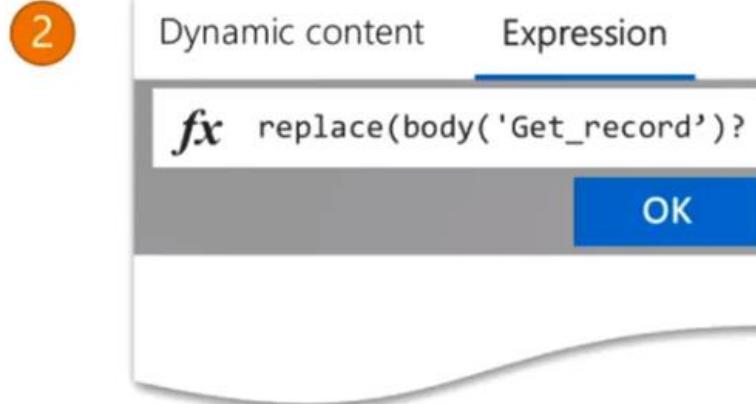
Many reasons to use expressions:

-  Convert types
-  Perform simple, inline calculations like string manipulation or arithmetic, or date/time manipulation
-  Generate data like the current time, a GUID or a random number
-  Handling optional values
-  "If" conditional statements
-  Working with lists



# How to create expressions

- 1 Add Dynamic Content



# Anatomy of an expression

myid: 123

mytitle: This item is due at <time>



This item is due at 2017-09-26T19:00:00Z

myname: Stephen

**fx** replace(body('Get\_record')?['mytitle'], '<time>', utcnow())

1. Expression name: **replace**
2. One or more parameters, separated by commas
3. Use single quotes around static text
4. Include expressions in other expressions
5. Use square brackets to get properties from objects
6. Use the ? for optional properties



# Data types and formats

## Text – aka *Strings*

- Normal – 'Example string'
- Email, URL, phone number, etc...
- Base64
- Binary content
- Data URI
- URI component

For strings use single quotes

## Non-text

- Floating-point number – 9.0
- Integer number – 137
- Boolean – true or false
- Array – a list of items
- Object – a record with key-value pairs
- XML content

Do *not* use single quotes for numbers, integers or booleans –

For example '`true`' does not equal `true`



# Handling content conversion

- Some conversion is automatic
- In some cases, a conversion expression is automatically added by the UI
- Other conversions you'll need to author an expression for directly
  - The expression is named for the *destination* type
  - For example to convert to base 64, use the `base64()` expression

`fx string(value)`

Convert the parameter to a string

`fx float(value)`

Convert the parameter argument to a floating-point number

`fx bool(value)`

Convert the parameter to a Boolean

`fx base64(value)`

Returns the base 64 representation of the input string

`fx base64ToBinary(value)`

Returns a binary representation of a base 64 encoded string

`fx base64ToString(value)`

Returns a string representation of a base 64 encoded string

`fx binary(value)`

Returns a binary representation of a value

`fx dataUriToBinary(value)`

Returns a binary representation of a data URI

`fx dataUriToString(value)`

Returns a string representation of a data URI

`fx dataUri(value)`

Returns a data URI of a value

`fx uriComponent(value)`

Returns a URI encoded representation of a value

`fx uriComponentToBinary(value)`

Returns a binary representation of a URI encoded string

`fx uriComponentToString(value)`

Returns a string representation of a URI encoded string

`fx array(value)`

Convert the input to an array



|                |                        | UI adds automatically |                        |                |                |           |                |         |                   |                   |                 |             |
|----------------|------------------------|-----------------------|------------------------|----------------|----------------|-----------|----------------|---------|-------------------|-------------------|-----------------|-------------|
| To<br>From     |                        | String                | Base 64                | Binary content | Data URI       | URI comp. | Floating-point | Integer | Bool.             | Array             | JSON Object     | XML content |
| String         | Yes                    | base64()              | binary()               | dataUri()      | uriComponent() | float()   | int()          | bool()  | split()<br>json() | json()            | xml()           |             |
| Base 64        | base64ToString()       | Yes                   | base64ToBinary()       | *              | *              | *         | *              | *       | *                 | *                 | *               | *           |
| Binary content | string()               | base64()              | Yes                    | dataUri()      | uriComponent() | *         | *              | *       | *                 | *                 | *               | *           |
| Data URI       | dataUriToString()      | *                     | dataUriToBinary()      | Yes            | *              | *         | *              | *       | *                 | *                 | *               | *           |
| URI comp.      | uriComponentToString() | *                     | uriComponentToBinary() | *              | Yes            | *         | *              | *       | *                 | *                 | *               | *           |
| Floating-point | Yes                    | base64()              | binary()               | dataUri()      | uriComponent() | Yes       | No             | No      | No                | No                | No              | No          |
| Integer        | Yes                    | base64()              | binary()               | dataUri()      | uriComponent() | Yes       | Yes            | No      | No                | No                | No              | No          |
| Bool.          | Yes                    | base64()              | binary()               | dataUri()      | uriComponent() | No        | No             | Yes     | No                | No                | No              | No          |
| Array          | join()<br>string()     | *                     | *                      | *              | *              | No        | No             | No      | Select Action     | Select or Compose | xml()           |             |
| JSON object    | string()               | *                     | *                      | *              | *              | No        | No             | No      | Select or Compose | Compose Action    | xml()           |             |
| XML content    | string()               | *                     | *                      | *              | *              | No        | No             | No      | xpath()           | xpath()           | Logic apps only |             |

\* First convert to/from a string and then use the expression required for normal strings



# Working with strings

- Combine strings together with `concat()`
- Take a *substring* – like `LEFT()` in Excel
- Replace certain parts of a string
- Convert casing with `toLower()` or `toUpper()`
- Find the location of text, or if it starts with or ends with certain text

 `concat(text_1, text_2?, ...)`

Combines any number of strings together

 `substring(text, startIndex, length)`

Returns a subset of characters from a string

 `replace(text, oldText, newText)`

Replaces a string with a given string

 `guid()`

Generates a globally unique string (GUID)

 `toLower(text)`

Converts a string to lowercase using the casing rules of the i...

 `toUpper(text)`

Converts a string to uppercase using the casing rules of the i...

 `indexOf(text, searchText)`

Returns the first index of a value within a string (case-insens...

 `lastIndexOf(text, searchText)`

Returns the last index of a value within a string (case-insensit...

 `startsWith(text, searchText)`

Checks if the string starts with a value (case-insensitive, invar...

 `endsWith(text, searchText)`

Checks if the string ends with a value (case-insensitive, invari...



# Arithmetic

Unlike Excel you cannot currently use the arithmetic operators you expect like +, -, \*, /

Instead you'll use expressions

For example to say  $X + Y * Z$

You would write `add(X,mul(Y,Z))`

**`fx min(collection or item1, item2?, ...)`**

Returns the minimum value in the input array of numbers

**`fx max(collection or item1, item2?, ...)`**

Returns the maximum value in the input array of numbers

**`fx rand(minValue, maxValue)`**

Generates a random integer within the specified range (inclu...

**`fx add(summand_1, summand_2)`**

Returns the result from adding the two numbers

**`fx sub(minuend, subtrahend)`**

Returns the result from subtracting two numbers

**`fx mul(multiplicand_1, multiplicand_2)`**

Returns the result from multiplying the two numbers

**`fx div(dividend, divisor)`**

Returns the result from dividing the two numbers

**`fx mod(dividend, divisor)`**

Returns the remainder after dividing the two numbers (mod...



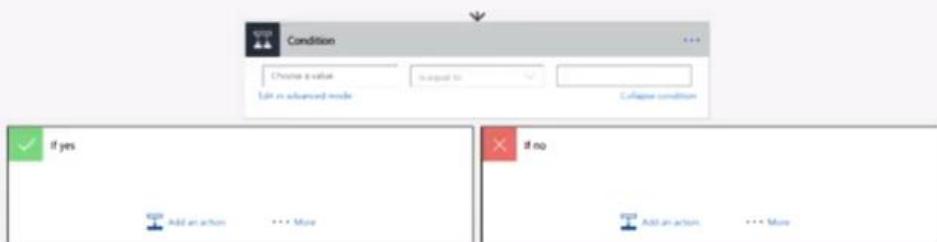
# Date and time

- `addToTime()` to add times together
- `convertTimeZone()` to handle local times
- `formatDateTime()` for displaying in emails
- `dayOfWeek()` to extract specific date segments

|  |   |
|--|---|
| <code>fx addMinutes(timestamp, minutes, format?)</code>                                  | Adds an integer number of minutes to a string timestamp pa...   |
| <code>fx addHours(timestamp, hours, format?)</code>                                      | Adds an integer number of hours to a string timestamp pass...   |
| <code>fx addDays(timestamp, days, format?)</code>  | Adds an integer number of days to a string timestamp passe...   |
| <code>fx convertTimeZone(timestamp, sourceTimeZone, destinationTimeZone, format?)</code> | Converts a string timestamp passed in from a source time zo...  |
| <code>fx convertToUtc(timestamp, sourceTimeZone, format?)</code>                         | Converts a string timestamp passed in from a source time zo...  |
| <code>fx convertFromUtc(timestamp, destinationTimeZone, format?)</code>                  | Converts a string timestamp passed in from a UTC to a target... |
| <code>fx formatDateTime(timestamp, format)</code>  | Returns a string in date format                                 |
| <code>fx startOfDay(timestamp, format)</code>  | Returns the start of the day to a string timestamp passed in    |
| <code>fx startOfMonth(timestamp, format)</code>  | Returns the start of the month of a string timestamp            |
| <code>fx startOfYear(timestamp, format)</code>   | Returns the start of the year of a string timestamp             |
| <code>fx dayOfWeek(timestamp)</code>   | Returns the day of week component of a string timestamp         |
| <code>fx dayOfMonth(timestamp)</code>  | Returns the day of month component of a string timestamp        |
| <code>fx dayOfYear(timestamp)</code>   | Returns the day of year component of a string timestamp         |
| <code>fx ticks(timestamp)</code>   | Returns the number of ticks (100 nanoseconds interval) since... |



# Expressions in conditions



1. First, build what you can in the simple mode
2. Then select **Edit in advanced mode**
3. Finally, enter the expression – use any of the Boolean expressions

NOTE: Unlike in Add Dynamic Content, you must begin with the @ character

**f<sub>b</sub>** equals(object1, object2)

Returns true if two values are equal

**f<sub>b</sub>** and(expression1, expression2)

Returns true if both parameters are true

**f<sub>b</sub>** or(expression1, expression2)

Returns true if either parameter is true

**f<sub>b</sub>** not(expression)

Returns true if the parameters are false

**f<sub>b</sub>** less(value, compareTo)

Returns true if the first argument is less than the second

**f<sub>b</sub>** lessOrEquals(value, compareTo)

Returns true if the first argument is less than or equal to th...

**f<sub>b</sub>** greater(value, compareTo)

Returns true if the first argument is greater than the second

**f<sub>b</sub>** greaterOrEquals(value, compareTo)

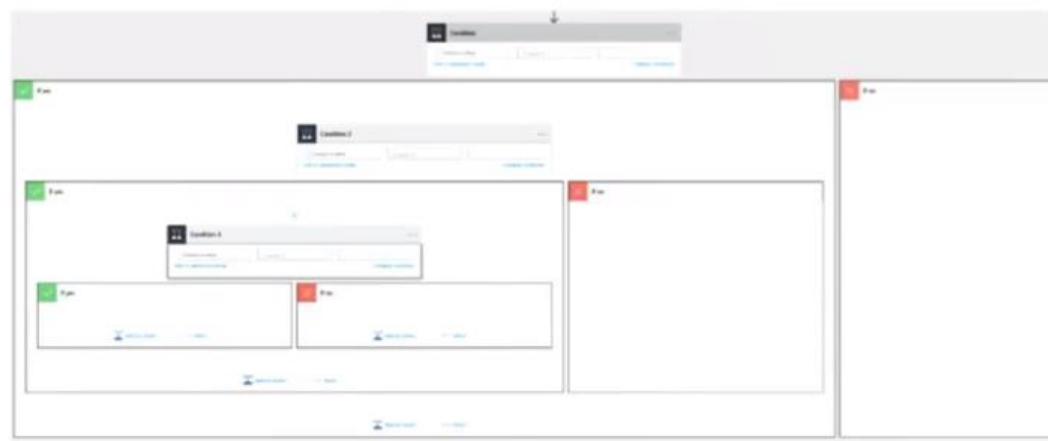
Returns true if the first argument is greater than or equal t...



# The **if()** expression

Why use the **if()** expression if you can use a condition?

Avoid multiple nested conditions just to set a field



**Caution:** The if does not 'short circuit' which can result in errors if you try to reference a missing property in one branch



## Handling optional values

**Caution:** trying to use a property that's not set – aka "null" -- will cause the flow to fail

Two steps to handle:

- 🚫 Use the "?" character after selecting the property (this happens automatically)
- ⚙️ Then, use the coalesce() function to provide a default value

```
coalesce(body('Get_record')?['content'], 'Default Value')
```



\* Site Address  
Process Simple Partners -  
<https://microsoft.sharepoint.com/teams/prospart>

\* List Name  
Project List

\* Id  
 ID x

\* Title  
 Title x

\* End Time  
 End Time x

\* Email aliases  
 Email aliases x

Start Time

Num of Days  
 Add dynamic content

Dynamic content Expression

c

- coalesce
- concat
- contains
- convertFromUtc
- convertTimeZone
- convertToUtc
- createArray
- decodeUriComponent
- encodeUriComponent
- listCallbackUrl
- uriComponent

# What are Arrays?

|       |      |                                      |
|-------|------|--------------------------------------|
| Jane  | 8.29 | ID: 123<br>Name: Jane<br>Mail: J@a.c |
| Dan   | 4.89 | ID: 888<br>Name: Bob<br>Mail: B@a.c  |
| Steve | 0.28 | ID: 456<br>Name: Dave<br>Mail: D@a.c |

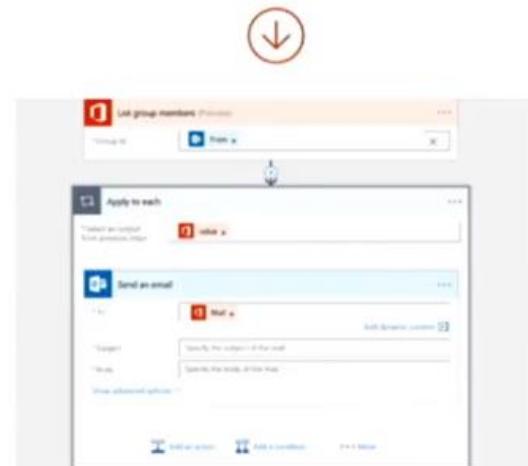
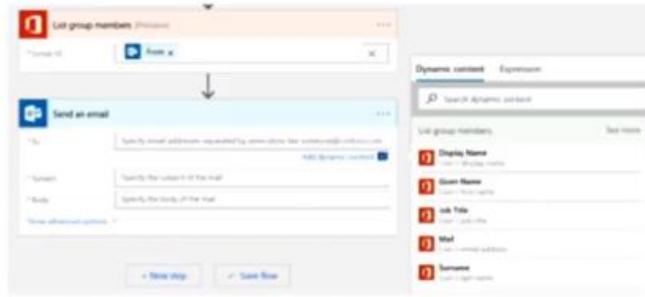
A few different classes of things you may need:

- ⚡ Repeat an action over each list item
- 📋 Get a single item from a list
- FilterWhere a list down
- ⟳ Make a list work with a different action



# Apply to each

- Automatically added when you try to use an output that's *part of a list*
- Destination step to repeats over every list item
- If you do NOT want this you can:
  - Get only one list item using an expression, or,
  - Convert the whole list to a string in the form of a list or a table



## Getting certain list items



Use the **first()** expression to get the first item from a list

Use square brackets and a number (starts at 0) to get a certain item, for example **[2]**

Use the **last()** expression to get the final item

**Note:** If the array is a list of Strings, Numbers, Booleans etc..., then the above expression can be used directly.

However, if it is an array of objects, you will need to **use square brackets** to select the specific property you need:

```
last(body('Get_rows')['value'])['mycolumn1']
body('Get_rows')['value'][2]['mycolumn1']
```



# Convert the list to a string

Join – use join to get a simple list,  
for example, if you have a list of  
email addresses\*

Create HTML/CSV Table – convert a  
list of Objects to a tabular display  
format, for example, for inclusion in  
the body

\* Note: You must have a list of strings, not a list of Objects

The image displays two separate Microsoft Power Automate flows, each consisting of three main steps: a 'Join' step, a 'Send an email' step, and a 'Create HTML table' step.

**Flow 1 (Top): Join and Send an email**

- Join Step:** Set to 'From' and 'Join with' both set to 'value x'.
- Send an email Step:** Set to 'To' as 'Output x'.
  - Subject: 'Specify the subject of the mail'
  - Body: 'Specify the body of the mail'

**Flow 2 (Bottom): Create HTML table and Send an email**

- Create HTML table Step:** Set to 'From' and 'Include Headers' set to 'Yes'.
  - Columns: 'Automatic'
- Send an email Step:** Set to 'To' as 'Output x'.
  - Subject: 'Specify the subject of the mail'
  - Body: 'Specify email addresses separated by semicolons like someone@contoso.com'



# Filtering

Ideally – have the connector do the filtering for you

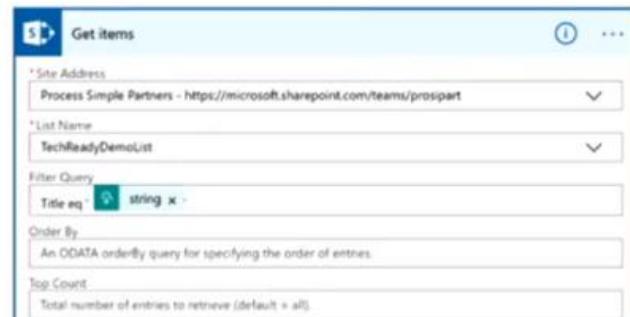
Requires use of Odata syntax and is limited. Read more [here](#).

## Built-in Flow action – Filter array

Select the Array in the From field

Can either use simple or advanced mode just like for Conditions

If you need the first N items you can use **take()** or **skip()**



# Transforming Arrays

Two input modes: fill key-value pairs, or typing directly in

 Create an array of objects



The screenshot shows the 'Select' action configuration. The 'From' section contains a single item labeled 'value x'. The 'Map' section is expanded, showing a table with two rows:

| Name  | Display Name |
|-------|--------------|
| Email | Mail         |

Below the table, there are input fields for 'Enter key' and 'Enter value', and a 'Add dynamic content' button.

Useful for passing this Array to another action

 Create an array strings, numbers, booleans etc...



The screenshot shows the 'Select' action configuration. The 'From' section contains a single item labeled 'value x'. The 'Map' section is expanded, showing a table with one row:

| Mail    |
|---------|
| value x |

Below the table, there is a 'Add dynamic content' button.

Useful for getting a simple list, for example, of email addresses



# Passing data into arrays

Can hardcode a list – or – have the list be dynamic at runtime

 Fill in each property of the object and get one item created

Send an email

|                     |  |
|---------------------|--|
| * To                | Specify email addresses separated by semicolons like someone@contoso.com |
| * Subject           | Specify the subject of the mail  |
| * Body              | Specify the body of the mail   |
| Attachments Name    | Attachment name  |
| Attachments Content | Attachment content   |
| Importance          | Importance   |

 Need to pass in an Array that exactly matches the fields

Send an email

|             |  |
|-------------|--|
| * To        | Specify email addresses separated by semicolons like someone@contoso.com |
| * Subject   | Specify the subject of the mail  |
| * Body      | Specify the body of the mail   |
| Attachments | Attachments  |
| Importance  | Importance   |



## Checking list containment or duplicates

- You can use the `contains()` expression to see if something is in an array
- Use the `union()` function with the array twice to get a set of unique items



## Building objects directly

You may need to use Compose to build an object or an Array

Use JSON syntax on the Compose action

```
{  
    "key" : "value"  
}  
  
[  
    "Bob",  
    "Jane"  
]
```



## Escaping rules

When you directly are using the expression language you may need to *escape* certain characters

Using strings with a single quote character:

```
@substring('It''s A Great Day!',1,5)
```

Using spaces in action names:

```
@body('Name_with_spaces')
```

Building an object you need to escape '@':

```
{
    "@@odata.type" : ""
}
```



## Normal action execution

Run in sequence

Flow terminates once an error (failure or timeout) is encountered

Skipped means the previous action didn't run

*Summary:* four possible states for an action:

- 👍 Succeeded
- 👎 Failed
- ✗ Timed out
- ↗ Skipped

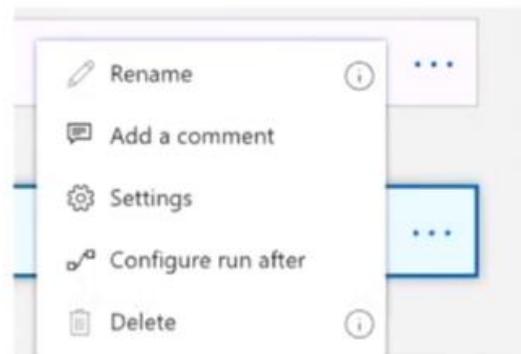


# Action settings

The “...” menu contains entry points to **Settings** and **Configure run after**

Settings lets you configure:

- Async Actions
- Timeout
- Retry policy
- Sequential
- And more!



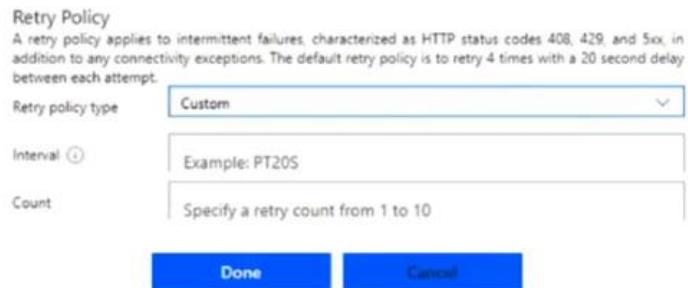
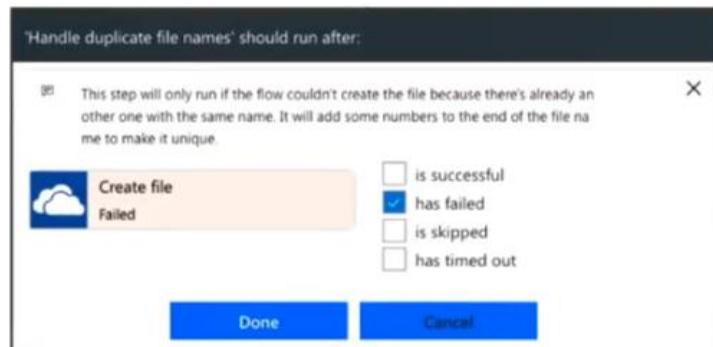
# Error handling

Select the Run After option in the action menu

Choose which error conditions,  
the arrow will turn dotted red

Use parallels for errors that  
aren't at the end of the flow

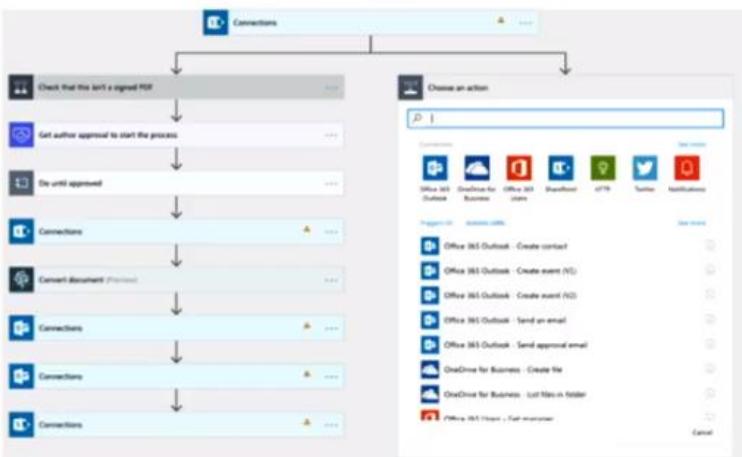
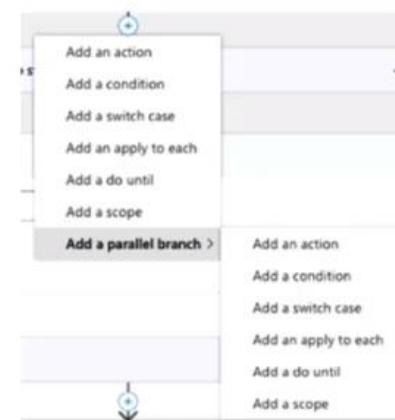
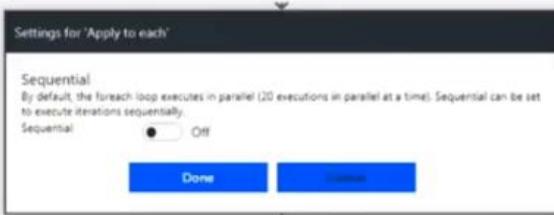
Retry policy by default handles  
transient failures



# Parallel execution

For actions select Add parallel branch from the  above the action

Inside of a foreach you can do up to 20 at a time

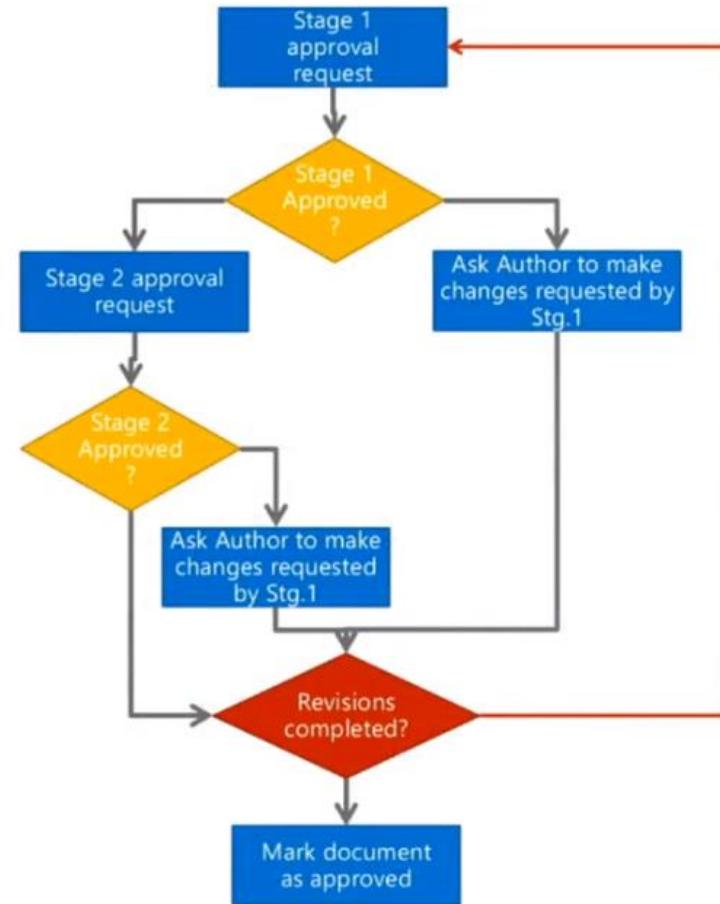


# Looping

Do until - actions that you want to repeat until a condition is met

Wait for an approval to be completed

Check out a dedicated session on approvals Friday

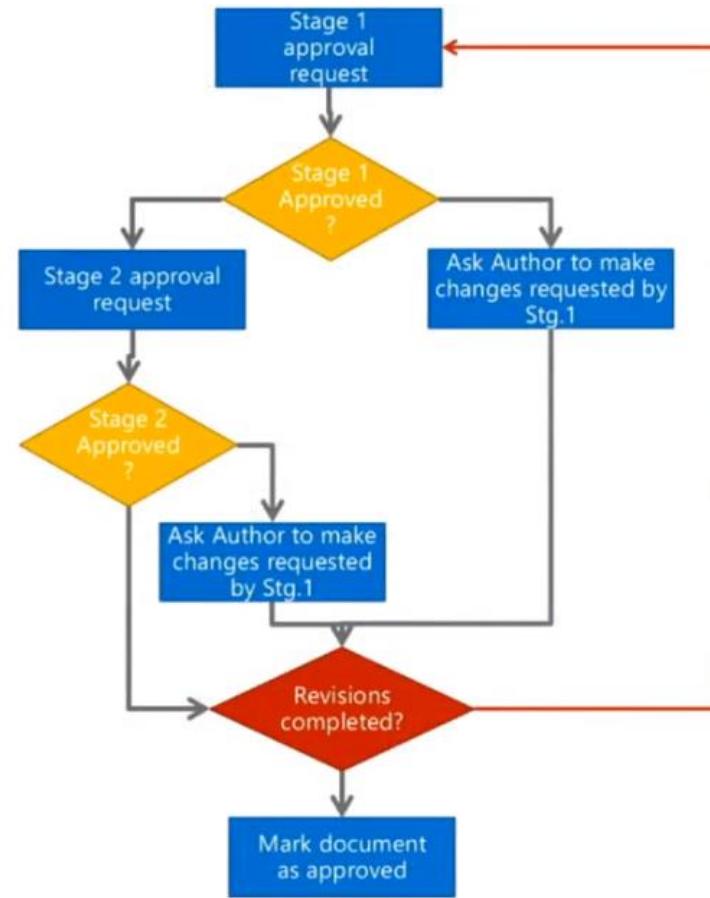


# Looping

Do until - actions that you want to repeat until a condition is met

Wait for an approval to be completed

Check out a dedicated session on approvals Friday

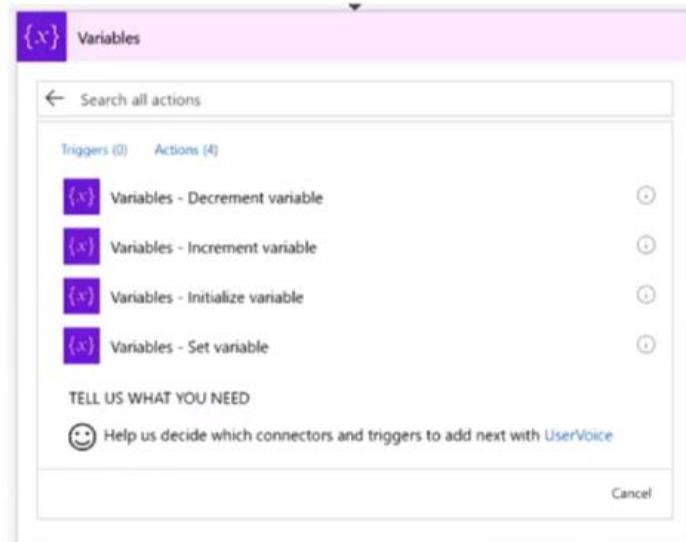


# Variables

Always define the type and initialize the variable before you use it.

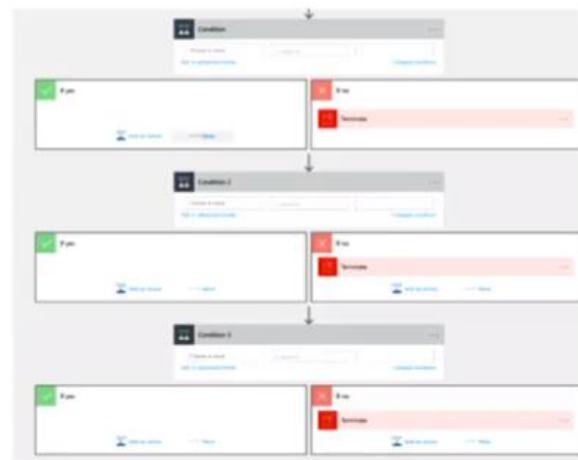
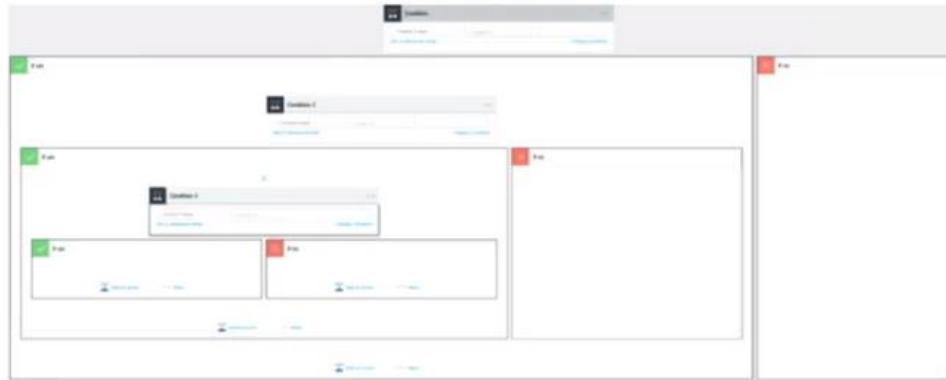
Useful for:

- Counters – you can increment or decrement variables
- Checking if something happened



# Terminate flows

You can use this action to avoid nested conditions by ending in the middle



# Asynchronous Actions

Any “trigger” can be in  
the middle of a Flow as well

For example, create a flow  
that waits for a reply to a  
certain thread in the middle

You probably want to configure  
a Time Out and handle that case



# Calling nested workflows

In the child workflow:

Request

HTTP POST URL: `https://prod-56.westus.logic.azure.com:443/workflows/419593519f75...`

Request Body JSON Schema

```
Example:
{
    "type": "object",
    "properties": {
        "address": {
            "type": "string"
        }
    },
    "required": ["address"]
}
```

Use sample payload to generate schema

Show advanced options ▾

In the parent workflow:

HTTP

\* Method: POST  
URL: `https://prod-56.westus.logic.azure.com:443/workflows/419593519f754af7a37f07ab5736bb98/triggers/manual/paths/invoke?api-version=2016-06-01&sp=%2Ftriggers%2Fmanual%2Frun&sv=1.0&sig=vIgMxpt4ArzyNbXfnZMFeZegoGx2KqW-ttAKi>3IGQ`

Headers: Enter key Enter value

Body: `{"message": "Success!"}`

Add dynamic content Show advanced options ▾

Response

\* Status Code: 200

Headers: Enter key Enter value

Body: `{"message": "Success!"}`

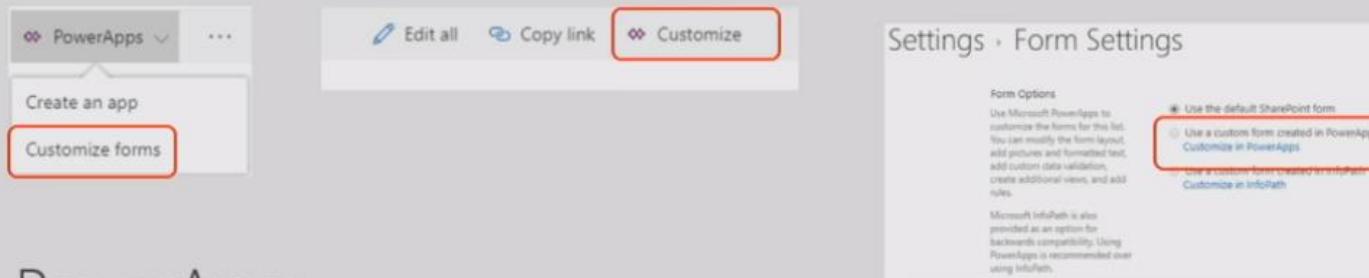
Add dynamic content



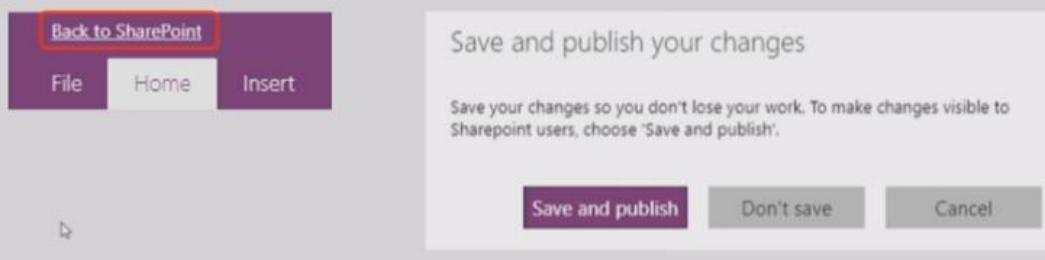
# Embedded in SharePoint

Smooth transition between platforms

- In SharePoint



- In PowerApps





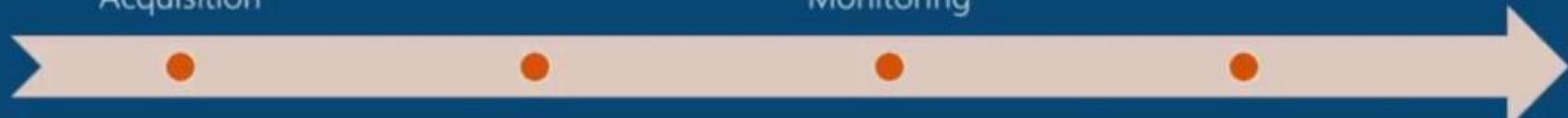
# Let's take an journey today

*Who is using these services?  
How do I manage licensing?*

*Where do I go to monitor  
usage across my org?*

Acquisition

Monitoring



Security

Application Lifecycle  
Management

*What data is accessible?  
What are environments?  
How do I control access?*

*How should I setup environments for my org?  
How do I deploy a solution for my org?  
How do I orchestrate updates?  
How do I identify and fix bugs & issues?*

- Environments
- Data policies
- Data integration
- Tenant
- User licenses
- Quotas

## Environments

Search



New environment

| NAME                    | REGION        | CREATED BY    | CREATED    |
|-------------------------|---------------|---------------|------------|
| Fabrikam Inc. (default) | United States | SYSTEM        | 05/09/2017 |
| Demo-Build              | United States | Jane Doe      | 05/09/2017 |
| Jane Doe's Environment  | United States | SYSTEM        | 05/10/2017 |
| Demo-Inspire            | United States | James Oleinik | 09/26/2017 |
| Ignite Demo UAT         | United States | James Oleinik | 09/27/2017 |
| Ignite Demo Production  | United States | James Oleinik | 09/27/2017 |

## Environment Admin

Search



Details + Users

Enter names, email addresses, or user groups

+ Add everyone in my org



Name

Email



Frank Weigel

lwuser7@powerflowbuild.onmicrosoft.com



James Oleinik

admin@powerflowbuild.onmicrosoft.com



Jane Doe

build@powerflowbuild.onmicrosoft.com



Cancel

Save



[← Ignite Demo UAT](#)

Delete Import resources Export resources

Security • Details • Resources • Database

Environment roles

User roles

Permission sets

| NAME              | DESCRIPTION  |
|-------------------|--|
| Environment Admin | An environment admin has the ability to perform all administrative actions on an en... |
| Environment Maker | An environment maker has the ability to create new resources in an environment.        |



Security • Details • Resources • Database

Environment roles

User roles

Permission sets

User roles

Search   New role

| NAME              | DESCRIPTION                                       |   |
|-------------------|---|---|
| Organization User | Default role that has access to public data.      |   |
| Database Owner    | Full access to all resources and database schema. |   |



Ignite 2017

Created: 09/27/2017 Modified: 09/27/2017

Delete

Details + Share

Enter names, email addresses, or user groups

+ Add everyone in my org

Shared with

| Name           | Email                                  | Permission |
|----------------|--|------------|
| James Oleinik  | admin@powerflowbuild.onmicrosoft.com   | Owner      |
| Charles Lamana | test2@powerflowbuild.onmicrosoft.com   | Can use    |
| Mark Overholt  | IWUser4@powerflowbuild.onmicrosoft.com | Can use    |
| Darshan Desai  | IWUser0@powerflowbuild.onmicrosoft.com | Can edit   |



 Ignite 2017  
Created: 09/27/2017 Modified: 09/27/2017

[Play](#) [Edit](#) [Delete](#) [Export \(preview\)](#)

[Details](#) • [Share](#) • [Versions](#) • [Settings](#) • [Analytics \(preview\)](#)

PowerApps keeps a version history of your apps.

| VERSION | MODIFIED               | MODIFIED BY           | POWERAPPS RELEASE | PUBLISHED         | COMMENT |  |  |
|---------|------------------------|-----------------------|-------------------|-------------------|---------|--|--|
| 7       | 9/27/2017, 2:41:03 PM  | Tenant Monkey Account | 2.0.703           | <span>Live</span> |         |  |  |
| 6       | 9/27/2017, 11:23:20 AM | Tenant Monkey Account | 2.0.703           |                   |         |  |  |
| 5       | 9/27/2017, 11:08:06 AM | Tenant Monkey Account | 2.0.703           |                   |         |  |  |
| 4       | 9/27/2017, 10:12:00 AM | Tenant Monkey Account | 2.0.703           |                   |         |  |  |
| 3       | 9/27/2017, 8:40:38 AM  | Tenant Monkey Account | 2.0.703           |                   |         |  |  |



# Application Lifecycle Management

When considering how to setup environments for your org, start simple!

## 1. Single environment model

Save+Publish allows testing & verifying new bits without impacting the users of your application

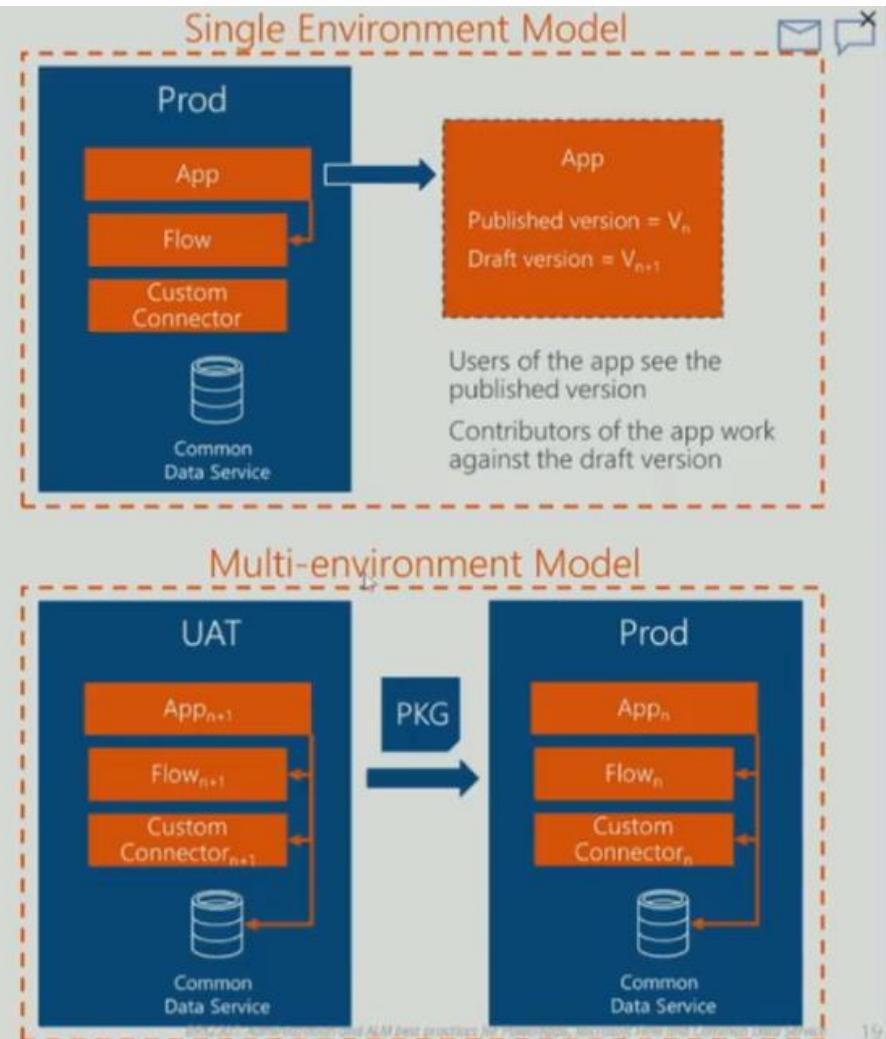
This model should work great for scenarios where you are not introducing breaking changes to your data model (either connectors or in your CDS schema)

## 2. Multi-environment model

Leverage packaging to move your solutions across environments (or tenants)

This model is required if your solution depends on custom connector or CDS where you need to introduce breaking change in the data model

Packaging requires a PowerApps Plan 2 or Microsoft Plan 2 license





PowerApps



Home



Apps



Learn



Connections



Flows



Gateways



Notifications

## Export package

### Package details

Created by James Oleinik on 09/27/2017

Name \*

Environment

Description

## Review Package Content

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

| NAME        | RESOURCE TYPE | IMPORT SETUP | ACTION  |
|-------------|---------------|--------------|---|
| ignite 2017 | App           | Update       |   |

Related resources

| NAME                      | RESOURCE TYPE                 | IMPORT SETUP         | ACTION  |
|---------------------------|-------------------------------|----------------------|---|
| Send out my meeting notes | Flow                          | Update               |   |
| jamesol@microsoft.com     | Office 365 Outlook Connection | Select during import |   |



## Import package

### Package details

Created by James Oleinik on 09/27/2017

#### Name

My pkg

#### Environment

Ignite Demo UAT

#### Description

N/A

### Review Package Content

Choose your import options.

| NAME  | RESOURCE TYPE | IMPORT SETUP  | ACTION |
|---|---------------|---------------|--------|
| <input checked="" type="checkbox"/> Ignite 2017 | App           | Create as new |        |

### Related resources

| NAME  | RESOURCE TYPE                 | IMPORT SETUP         | ACTION |
|---|-------------------------------|----------------------|--------|
| <input checked="" type="checkbox"/> Send out my meeting notes | Flow                          | Create as new        |        |
| <input checked="" type="checkbox"/> jamesol@microsoft.com     | Office 365 Outlook Connection | Select during import |        |

Import

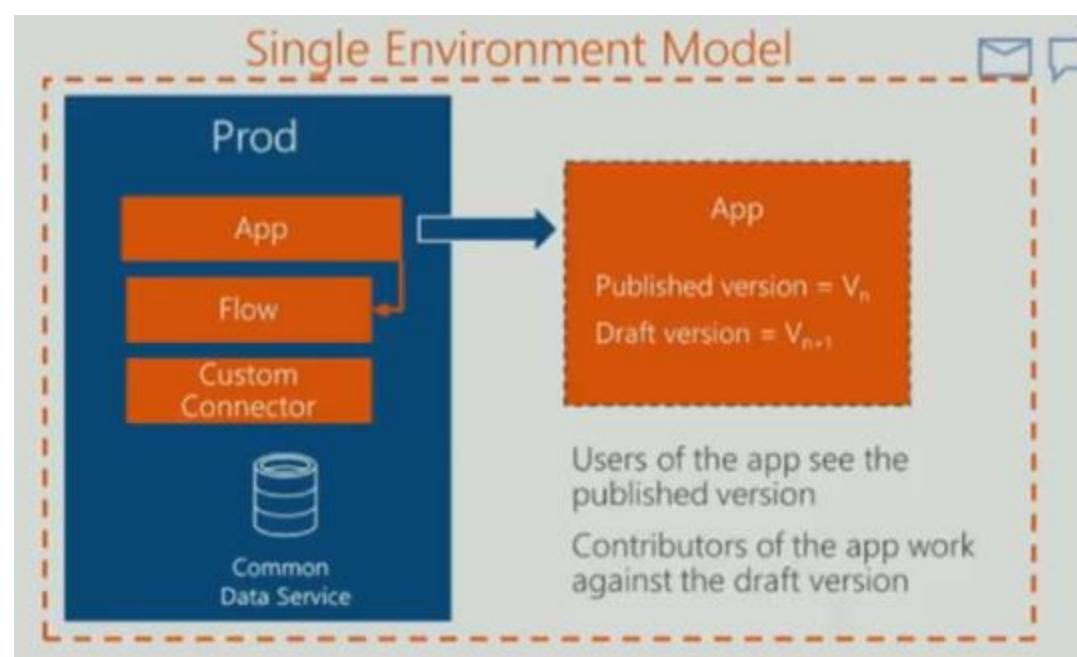
Cancel



## 1. Single environment model

Save+Publish allows testing & verifying new bits without impacting the users of your application

This model should work great for scenarios where you are not introducing breaking changes to your data model (either connectors or in your CDS schema)

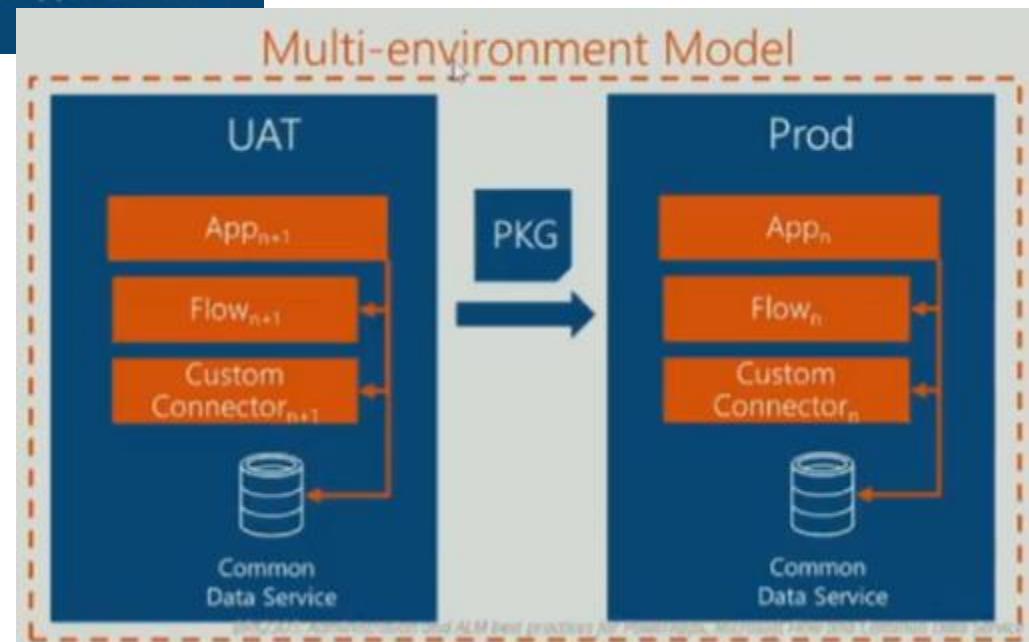


## 2. Multi-environment model

Leverage packaging to move your solutions across environments (or tenants)

This model is required if you solution depends on custom connector or CDS where you need to introduce breaking change in the data model

Packaging requires a PowerApps Plan 2 or Microsoft Plan 2 license



# Environments

## Security

PowerApps and Microsoft Flow are compliant with 13+ standards including HIPAA, EU Model Clauses, ISO & SOC

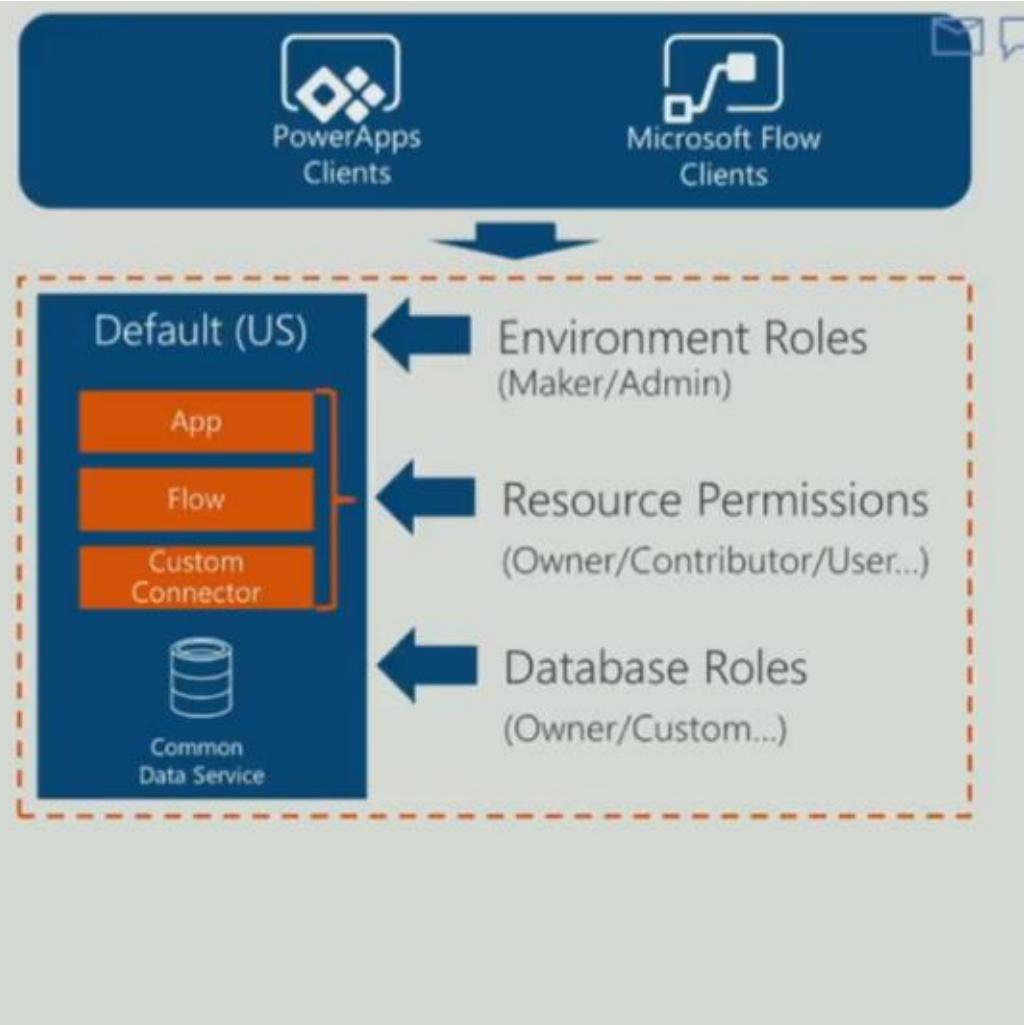
Environments serve two purposes in the business application platform:

1. Resource partitioning (by region)
2. Governance boundary

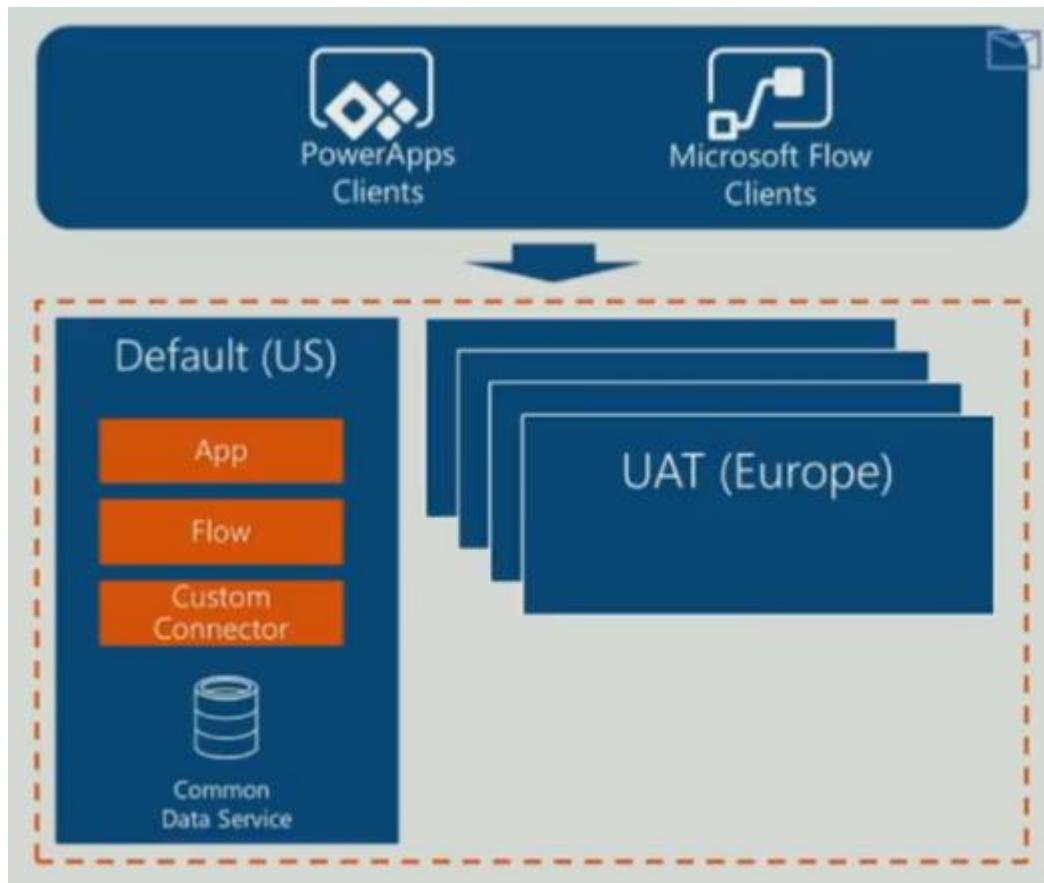
Every tenant is automatically provisioned a Default environment

Access is controlled at three levels:

1. Environment roles
2. Resource permissions for apps/flows/etc.
3. Database roles

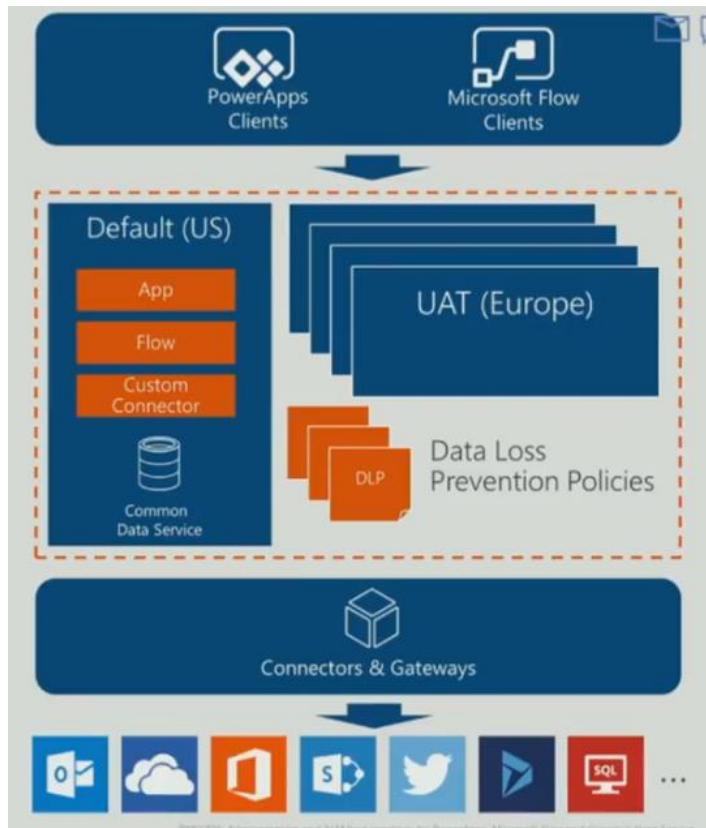


Additional environments can be provisioned by users with a PowerApps Plan 2 paid or trial license



# Policies

Admins can set DLP policies to control access to data and connectors from apps and flows



# Policy Violation

- Business Data vs Non-business Data

Error

Using these connections together conflicts with the company data loss prevention policies. Click [here](#) to learn more.

More

Ok



# Monitoring

PowerApps Admin center allows you to centrally monitor and manage your org's resources:

1. View and manage environments
2. View and manage all apps and flows within an environment
3. View and manage your CDS database

View app usage analytics from [web.powerapps.com](http://web.powerapps.com)  
(app performance and error reporting reports are coming soon)

Flow usage analytics is also coming soon

These features require a PowerApps Plan 2 or Microsoft Plan 2 license

The image shows two screenshots of the PowerApps Admin center. The top screenshot is a desktop view of the 'Environments' page, listing three environments: 'Fabrikam Inc. (default)', 'Demo-Build', and 'Jane Doe's Environment'. The bottom screenshot is a tablet view of the 'Analytics (preview)' page for the 'Org Browser (Phone)' app, showing various charts and data tables related to app usage and performance.





# Acquisition

PowerApps licenses always include an equivalent license for Flow

PowerApps and Flow plans are included within Office 365 and Dynamics 365 subscription

Individual users within an organization can directly sign-up for the following plans:

1. PowerApps Plan 2 90-day trial
2. PowerApps Community Plan
3. Flow Free

Admins can purchase and assign licenses for the stand-alone Plan 1 and Plan 2 offerings for PowerApps and Microsoft Flow

All licenses are per-user and can be managed in the Office 365 Admin Center or via PowerShell

The user licenses report for PowerApps and Microsoft Flow is available in the PowerApps Admin Center



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 F1\*



Dynamics 365 for Sales, Enterprise edition  
Dynamics 365 for Customer Service, Enterprise edition  
Dynamics 365 for Operations, Enterprise edition  
Dynamics 365 for Field Service, Enterprise edition  
Dynamics 365 for Project Service Automation, Enterprise edition  
Dynamics 365 for Team Members, Enterprise edition  
Dynamics 365 for Financials, Business edition  
Dynamics 365 for Team Members, Business edition  
Dynamics 365, Enterprise edition, Plan 1\*\*  
Dynamics 365, Enterprise edition, Plan 2 \*\*



Flow Free  
Flow Plan 1  
Flow Plan 2



PowerApps Plan 1  
PowerApps Plan 2 (+90d trial)  
PowerApps Community Plan

# Roadmap (highlights)



## H2 Calendar 2017

- Export/import Flows across environments
- "In-the-box" review Flows for SharePoint Online
- Use Flow for Content Publishing approvals for SharePoint Online Publishing sites
- Utilize people, choice, attachment and multi-values in your Flows for SharePoint Online
- In-context Flow support in Dynamics 365, OneDrive and Teams
- Usage analytics for Flows
- Flow activity included in Office 365 audit logs
- UK deployment

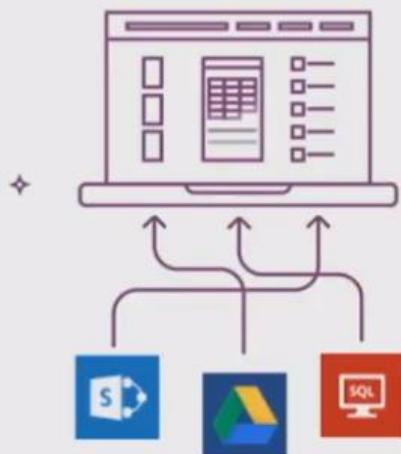


## Calendar 2018

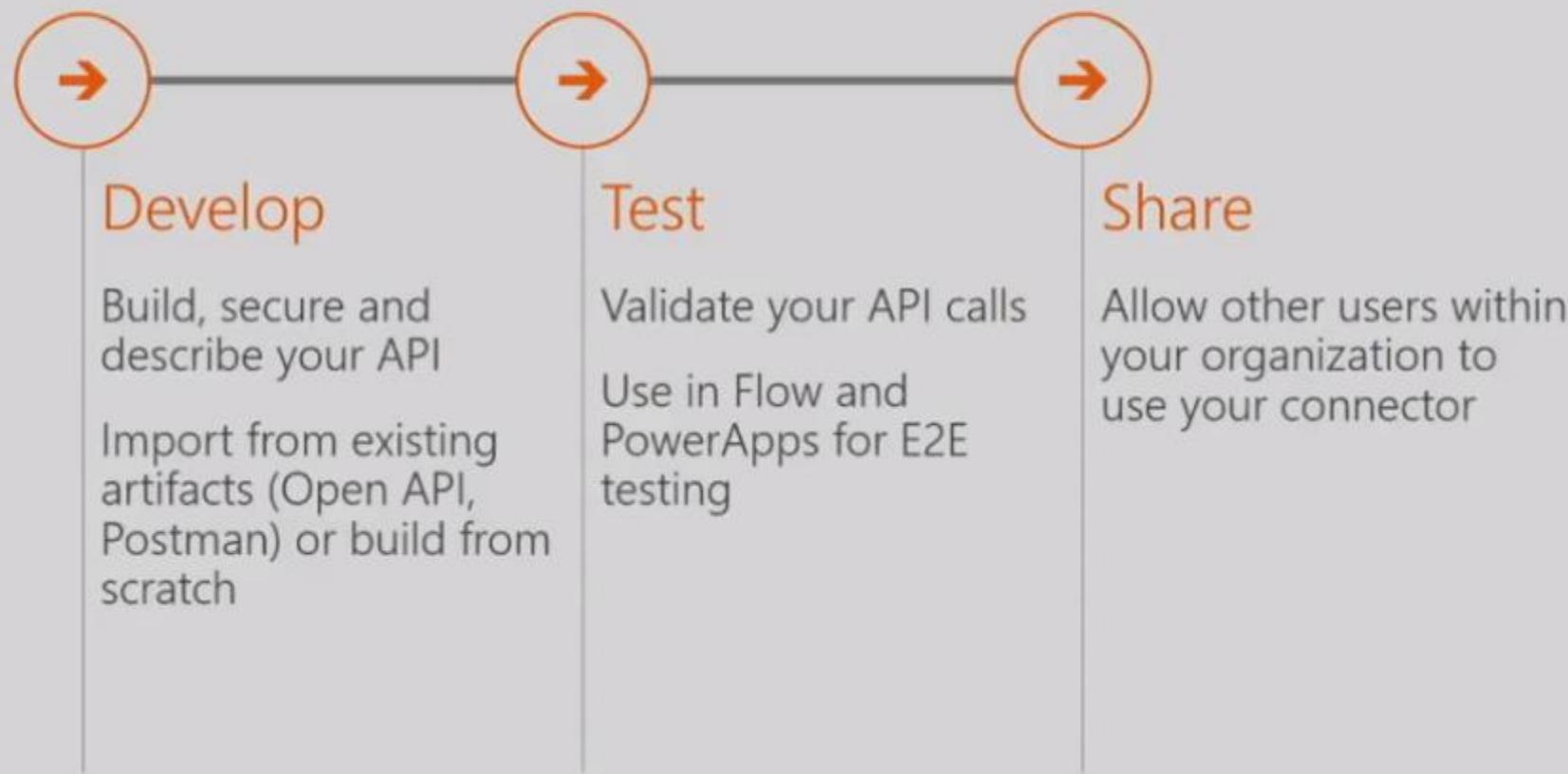
- Use a HTML rich text editor to build emails
- Guides for debugging actions and conditions
- Government cloud deployment
- ALM – Sandbox environments
- ALM – Move or copy across environments
- Support for GDPR compliance
- Leverage device signals (e.g. Location) with Flows on mobile devices
- Include Flows in AppSource
- Azure functions integration



A **connector** is a wrapper around an API that allows the underlying service to talk to **Microsoft Flow** and **PowerApps**



# Building a Custom Connector



# Why build a connector?



## Enterprise developers

Looking for connectivity to a service we don't currently support  
Need connectivity to a custom or internal service



## Partners and ISVs

Expand the number of services you integrate with  
Provide the extensibility your customers need  
Increase exposure and adoption



# Building a Custom Connector

