



Track – Power Apps

Introduction to Power Apps Component Framework



Modern Workplace Conference Paris 2021

19 janvier 2021 15:00 UTC+1

INTRODUCTION AU POWER APPS COMPONENT FRAMEWORK

Allan De Castro 

Power Platform Consultant @Avanade





Exakis Nelite
MagellanPartners

MARTELLO

JALIOS

avanade

AZEO
talents & technology

BitTitan

Powell Software

Aerow
smart contents



Microsoft

Modern Workplace
Conference Paris 2021
online

#MWCP21

18,19,20 Jan 2021

Thanks to
our sponsors !

Merci à nos sponsors

Suivez-nous tout au long de l'année !

Follow us all year round!



<https://aos.community>

<https://modern-workplace.pro>

<https://twitter.com/aOSComm>



<https://twitter.com/mwcparis>

#MWCP21

<https://www.linkedin.com/company/ams-community>



<https://www.linkedin.com/company/mwcp>

<https://www.facebook.com/aOSCommunity> (FR)

<https://www.facebook.com/aosComm> (EN)



<https://www.facebook.com/modernworkplaceconferenceparis>





Allan De Castro

Power Platform Consultant **@AvanadeFrance!**

Passionate about Dynamics 365 CE and the Power Platform.



[Linkedin.com/in/allandecastro/](https://www.linkedin.com/in/allandecastro/)



[Twitter.com/decastroallan](https://twitter.com/decastroallan)



www.blog.allandecastro.com



Activities & timeline

Here is the schedule of this Power Apps Component Framework session! Hopefully, you will like it...

What is the Power Apps Component Framework?

Live demonstration

Getting started with PCF (a.k.a. Goodbye Web Resources)

QnA

10 Minutes

- History
- Definition
- Availability
- Types of components

15 Minutes

- Community Stuff
- Deployment and configuration
- PhoneNumberByCountry
- VATNumberValidator
- BusinessProcessFlowViewer

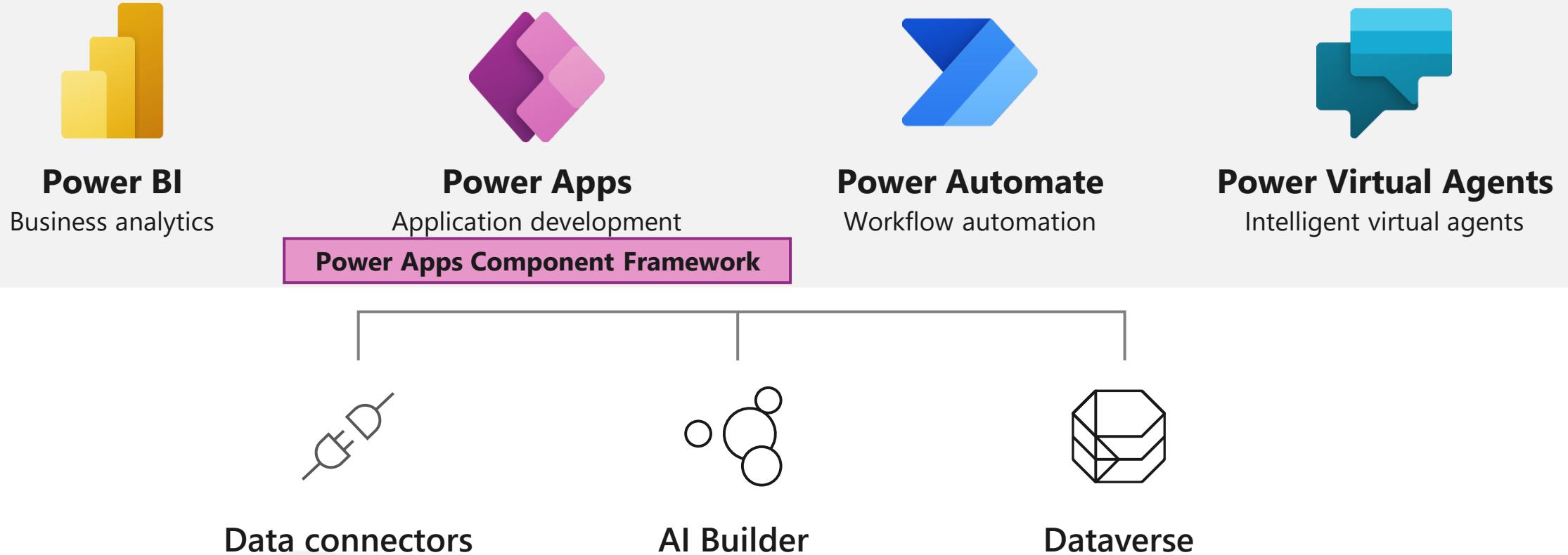
20 Minutes

- Technologies
- Framework
- First steps
- Tools needed
- Power Apps CLI
- XrmToolBox Plugin
- Architecture
- Lifecycle

What is the Power Apps Component Framework?



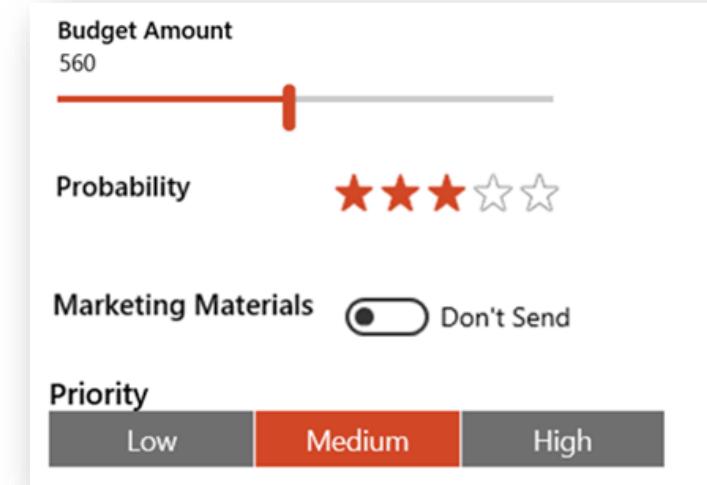
What is Power Apps?



History

First Components introduced in 2016:

Set of optimized mobile controls

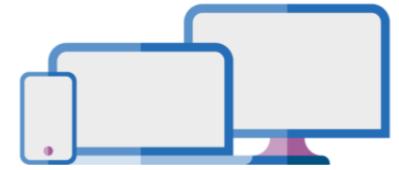


Unified Interface -> Components for all

At that time we were talking about Custom Control using an exclusive Microsoft framework

Power Platform, renamed into Power Apps Component Framework

~~Custom Control~~ → PCF → Power Apps Component Framework



Definition

Model Driven App, Canvas Apps & Portal

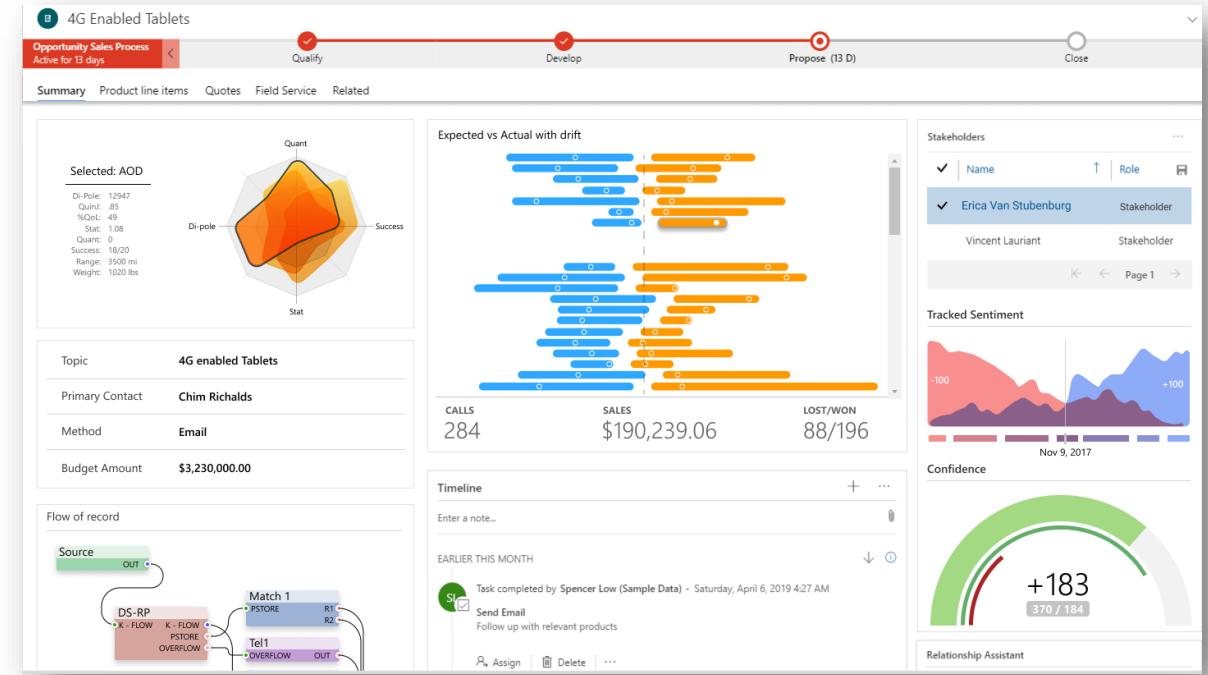
1st and 3rd party developers

Professionally developed controls with well defined contracts, lifecycle and interaction model

Improves performance

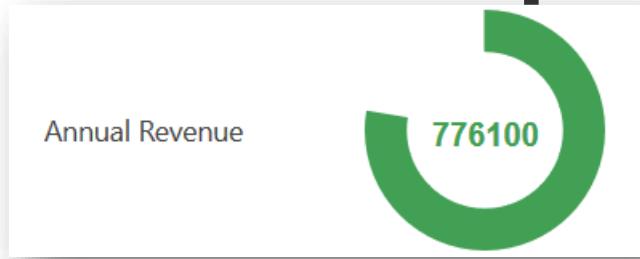
Create your own user experience

Modern Tooling (CLI)

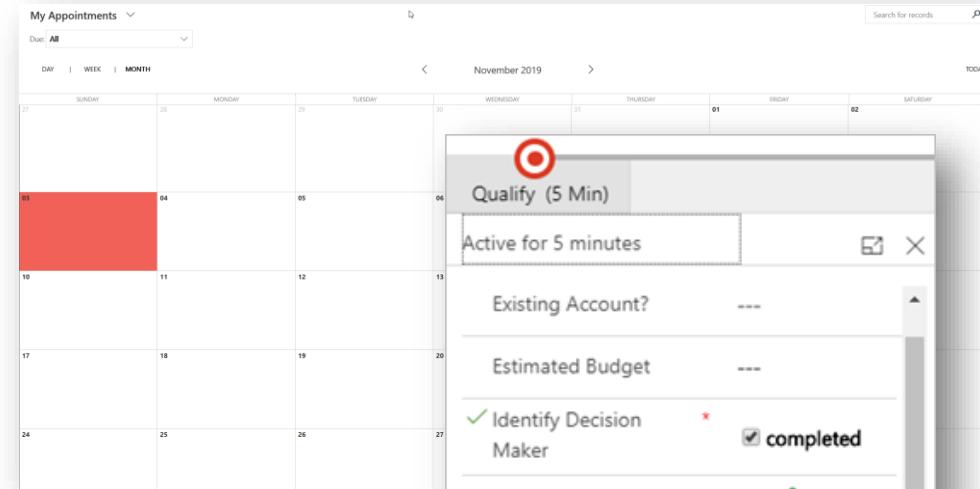


Out-of-the-box component

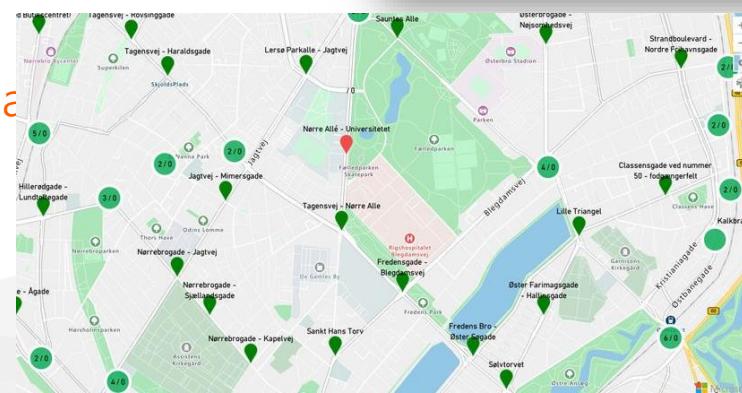
Field on the form



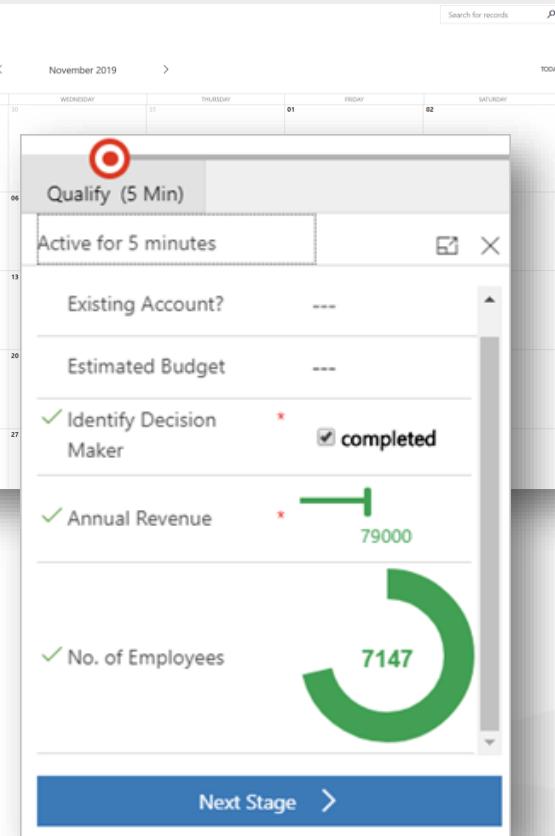
DataSet (Entity Grid/SubGrid)



Business Process step



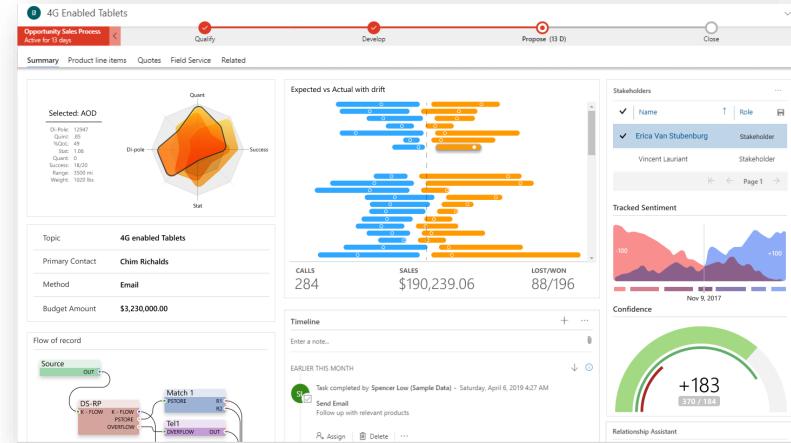
Display/Use external content (LinkedIn, API, Azure Ma



Availability

Public Preview, April 23, 2019

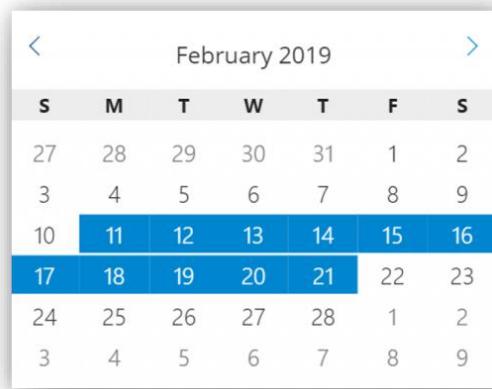
The PowerApps component framework and the PowerApps CLI is now available for public preview



General Availability, October 1, 2019

Announcing the general availability of the PowerApps component framework for model-driven applications and PowerApps CLI

Two Types of Components



Custom Canvas Compositions : Low-code composition using expressions and connectors (Canvas App).

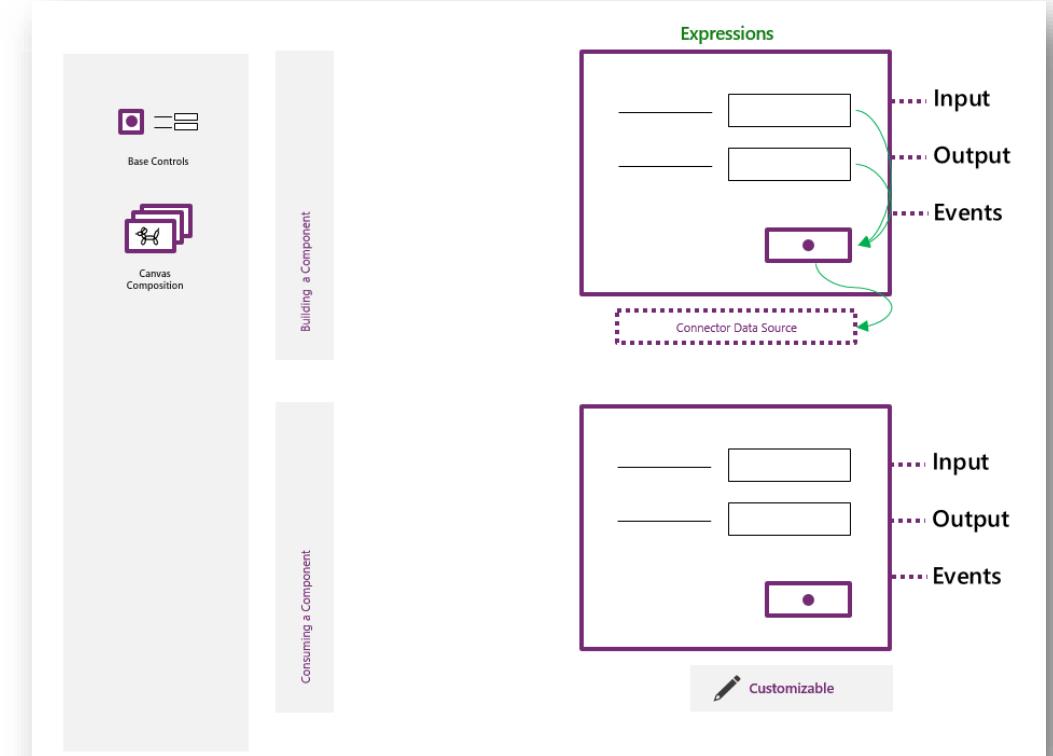
Custom Pro Components : Create more sophisticated component using code (Model Driven App, Canvas App & Portal)

Custom Canvas Compositions

Allows you to connect the **basic** commands between base control and the connectors using **expressions**.

Use of OOB controls available, enriched with expressions and connected data (using connector) to produce **reusable components**.

Input and output properties to **interact** with the component by passing data (can be used to change the visual aspect).



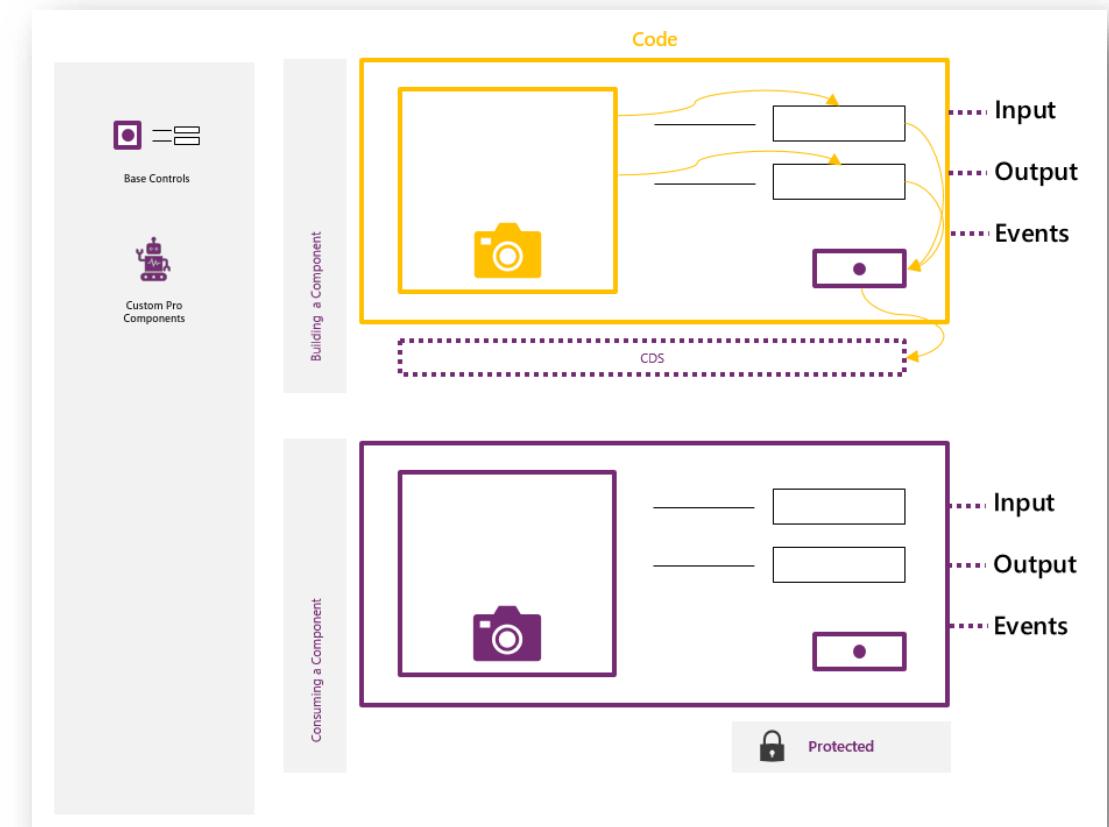
Business App Summit 2019: Microsoft PowerApps: Building and Extending Applications

Custom Pro (Dev) Compositions

Using **code** to connect base controls.

Custom Pro (Dev) Components use input and output properties and can **connect natively to Dataverse**.

Custom Pro (Dev) Components are consumed as **black box components** to Makers, and can't be **customized through maker experiences**, except for developer designated properties.



Business App Summit 2019: Microsoft PowerApps: Building and Extending Applications

Resume

An **extensibility** Framework for **professional developer** to create custom component and **user experiences** to be leveraged by **citizen developers**:



Conception

- Using full HTML, JavaScript, React.. to create custom UI.
- Access to an extensive API Surface.



Packaging

- PowerApps CLI (Managed & UnManaged).



Installation

- Rich ecosystem of reusable, accessible, and performant UI component.
- Import into Dataverse Environment



Configuration

- Configurable by Makers



Client Rendering

- Both App + Portal
- Adapts according to the configuration
- Improves performance

Examples

Products By Industry

Disposable Tableware

- Food Service
- Disposable Lids**
- Disposable Serveware
- Disposable Tableware**
- Food Service Baskets
- Heavy Machinery
- Chippers

Indie Flower

PowerFont sample PCF Component

This is a test HTML Text control

EAN13

5 901234 123457

EAN13 customized

PCF Controls

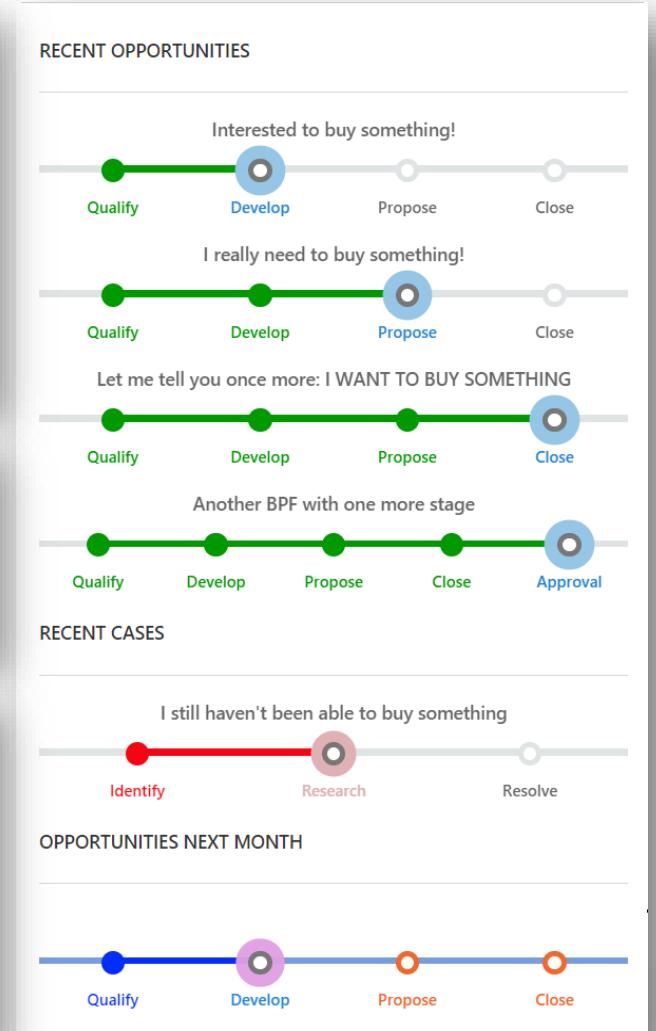
Siren

IsValidSiren Yes

France	+33 6 34 55 66 77	
USA	+44 20 8366 1177	
Algeria	+213 634 55 66 77	
Swiss	+41 44 668 18 00	
Italy	+39 365 478 9145	
Portugal	+351 22 040 4000	

Component

Azure PaaS Azure SaaS Field Services Finance and Operations IoT
 Marketing Power BI Project Service Automation Retail



Live demonstration

- Community Stuff
- Deployment and configuration
- PhoneNumberByCountry
- VATNumberValidator
- BusinessProcessFlowViewer



Examples

PCF Gallery by Guido Preite @crmanswers



The screenshot displays a collection of Power Apps controls and examples from the PCF Gallery:

- Notification Control:** A card-based interface for handling notifications, showing messages like "Received Image" and "Received Text".
- Gantt Progress Timeline:** A Gantt chart showing tasks over time.
- CONTACT:** A contact record for "Hello Bartle" with tabs for Summary, Details, and Related.
- Image To Text Control:** An example where an image of text ("WE CREATED MYTHS to unite our species") is converted into text ("WE CREATED MYTHS to unite our species").
- Text:** A control that extracts text from an image.
- PCF AppInsights:** A dashboard showing app usage statistics for 6.5K users.
- Energy Efficiency Rating Control:** Three sliders for rating energy efficiency from A to E.
- All Tasks - Kanban:** A Kanban board for tracking tasks across three columns: Not Started, In Progress, and Waiting on someone else.
- Status Reason Kanban:** A Kanban board for tracking tasks with specific status reasons.
- Immersive Reader Demo:** A demo of the Immersive Reader feature, showing text with highlighted words and definitions.
- RECENT OPPORTUNITIES:** A section showing recent opportunities with a timeline from Quality to Approval.
- Business Process Flow Viewer:** A visual representation of business process flows with stages: Quality, Develop, Propose, Close, and Approval.
- Attachment Manager:** A file manager interface showing attachments for an item.
- YouTube Player:** A player for a YouTube video titled "How to Use your data with IN 1 MINUTE!"
- Rich Text Editor with Chart & UML Designer:** A rich text editor with integrated chart and UML diagram tools.



DEMO

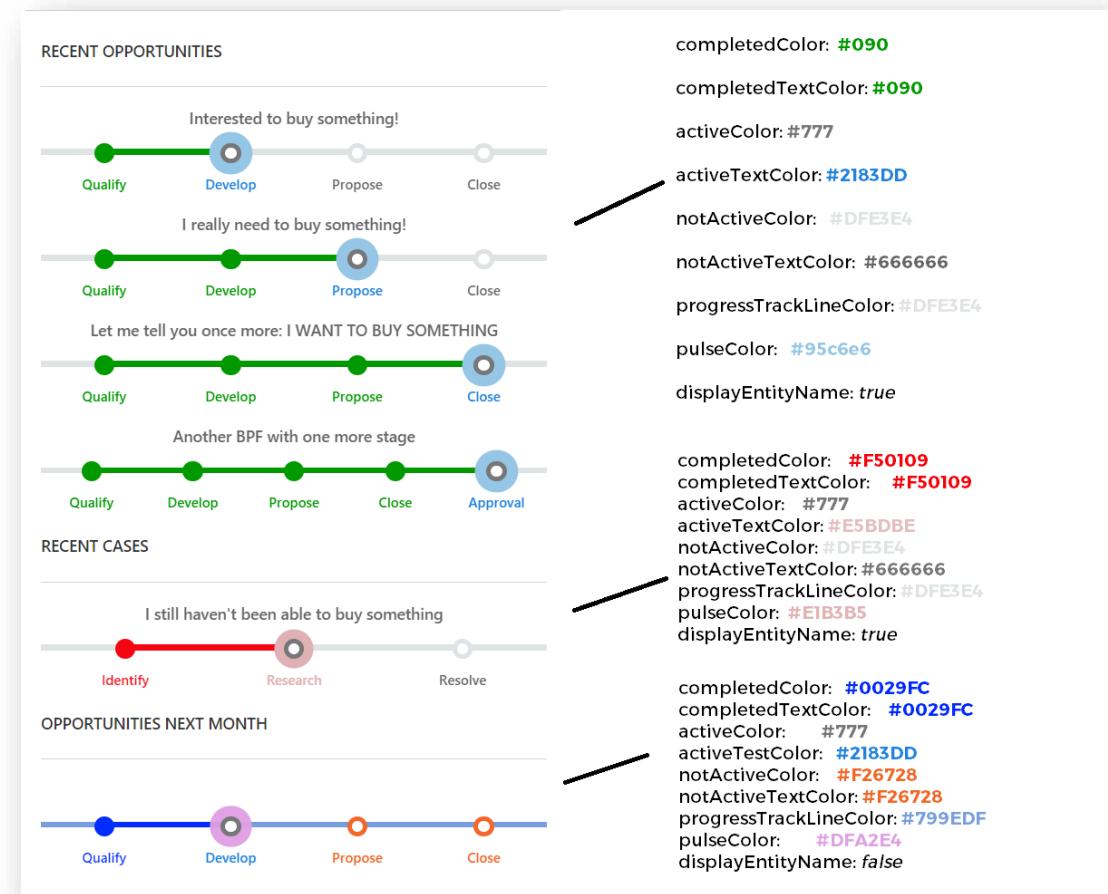
Examples

Business Process Flow Viewer

Displays BPFs with all process stages OR categories of entities from a grid/views.

Benefits:

- Improves information's visualization
- Fully configurable
- Supports any entities (OOB, custom...)
- Supports any BPFs (OOB, custom...)
- FREE!
<https://github.com/allandecastro/BusinessProcessFlowViewer>



- Home
- Recent
- Pinned

My Work

- Dashboards
- Activities

Customers

- Accounts
- Contacts

Sales

- Leads
- Opportunities
- Competitors

Collateral

- Quotes
- Orders
- Invoices
- Products
- Sales Literature

Marketing

Show Chart	New	Delete	Refresh	Email a Link	Flow	Run Report	Excel Templates	Export to Excel	Import from Excel	Quick Campaign	...
Account Name	Main Phone	Address 1: City	Primary Contact	Email (Primary Contact)							
Allan De Castro, Inc.	+33 6 24 08 51 06	---	---	---							

Search for records

Getting started with PCF (a.k.a. **Goodbye Web Resources**)

- Technologies
- Approach
- Tools & Architecture

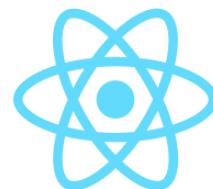


Technologies

Many **advanced technologies** to take these components to the **next level!**



TypeScript

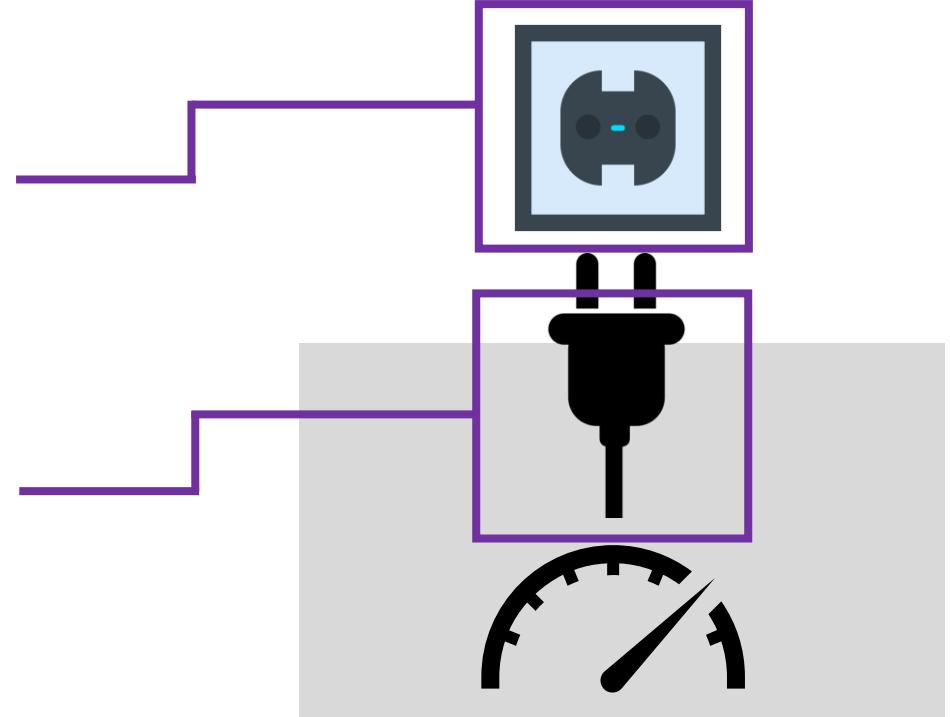


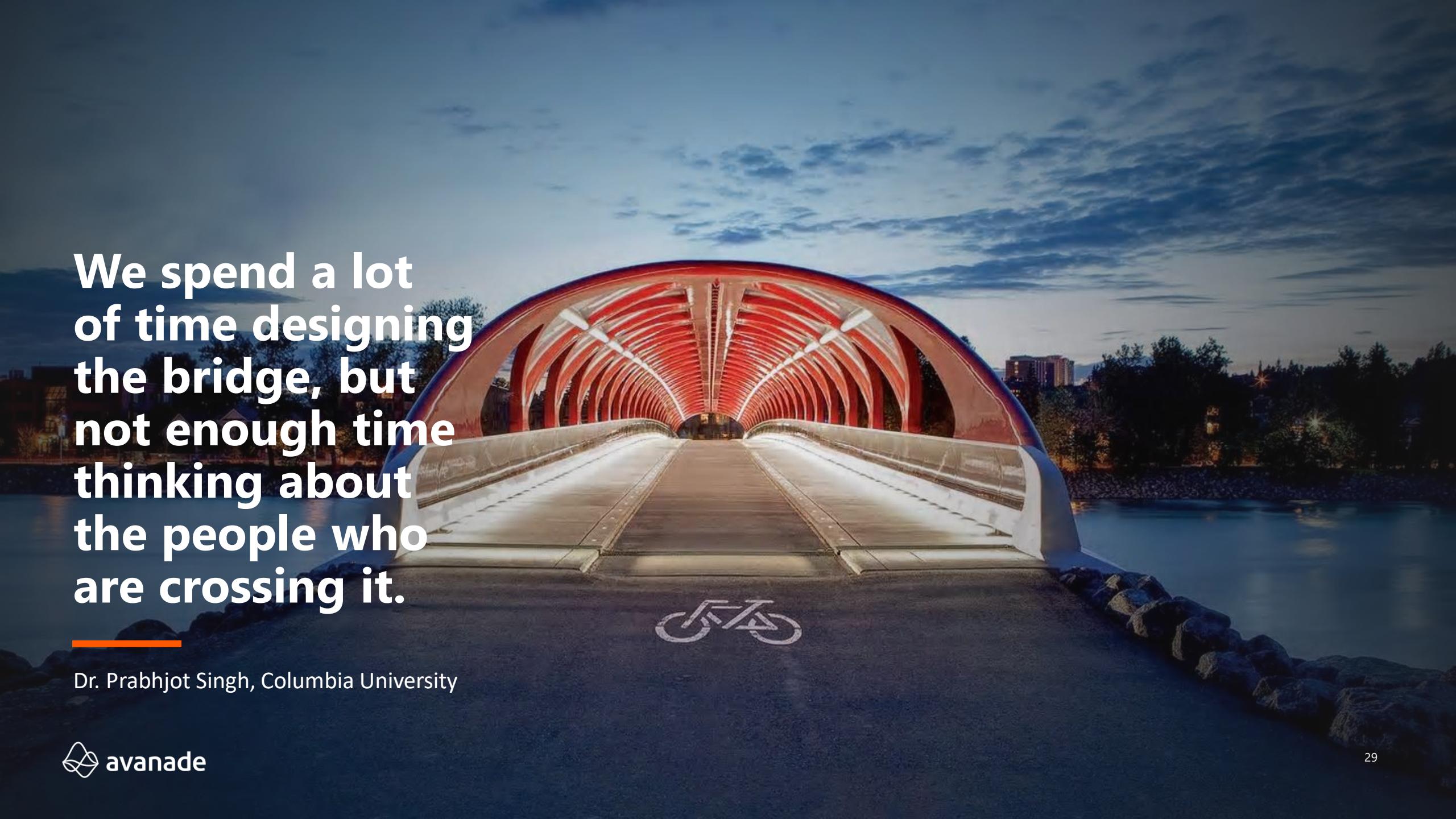
Framework Concept

Framework Interface aka Socket

Interface API SDK provides methods for control lifecycle, data/metadata access and more.

Component implement the framework interface and functional logic.





**We spend a lot
of time designing
the bridge, but
not enough time
thinking about
the people who
are crossing it.**

Dr. Prabhjot Singh, Columbia University

Thinking before acting

What problem do I want to answer ?



What is the added value of building a component?



What features must be implemented to take full advantage of this component?



What types of data will it use?
Will it be used on a field or on a dataset?



What parameters do I want to keep free to modify?

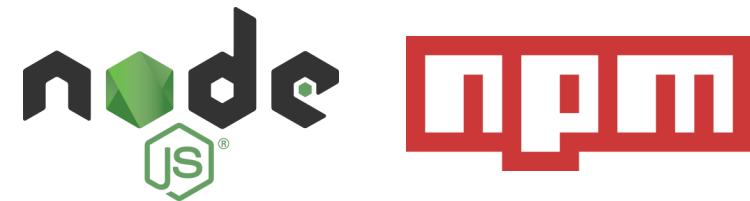


External services ?

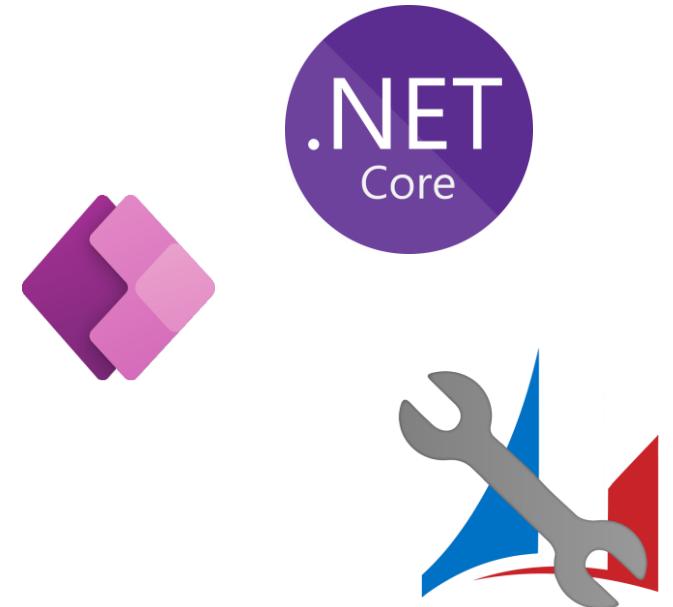


Equip yourself!

Install [Npm](#) or [Node.js](#), one will come with the other in any case!



Make sure you have the [.NET Framework 4.6.2 Developer Pack](#).



Install [Microsoft PowerApps CLI](#).

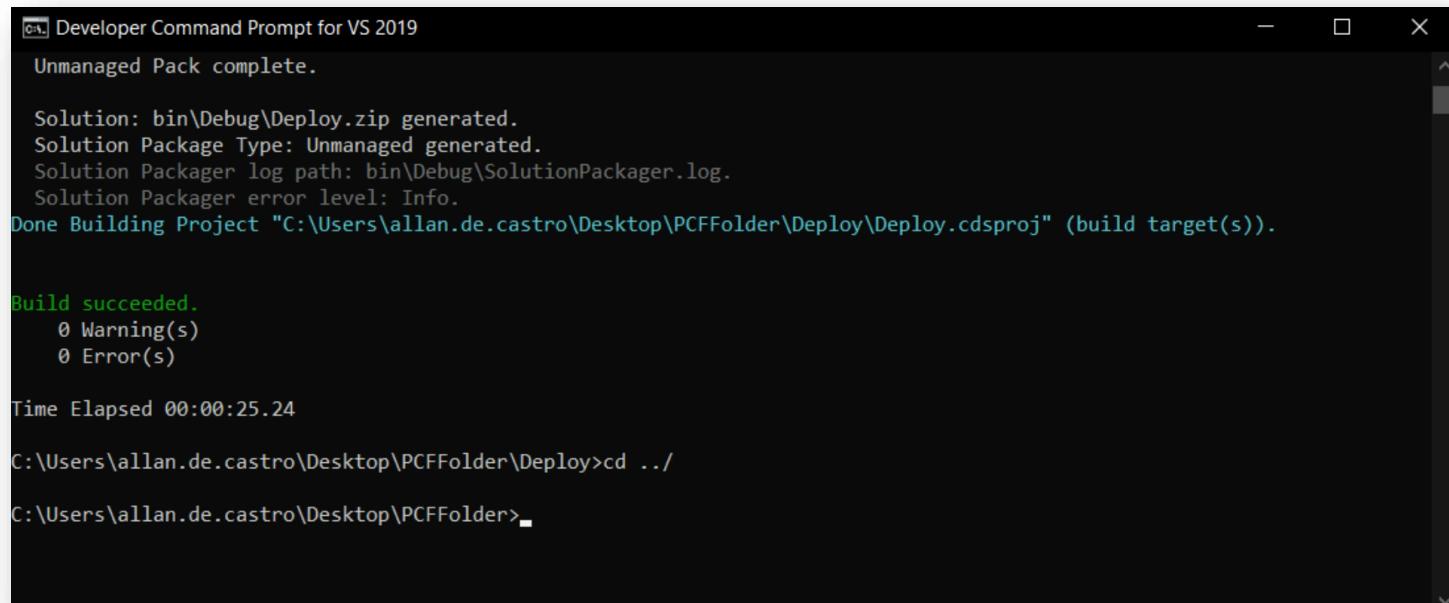
Note: XrmToolBox plugin exists [PCF Custom Control Builder](#).

Power Apps CLI

A unique tool that initially focuses on developing custom controls, plugins and web resources.

Command Line Interface which enables developers to be more efficient in their PCF development !

- Control templates (field and dataset)
- Build in validations to catch issues prior to deployment
- Can be used as part of your automated processes



```
Developer Command Prompt for VS 2019
Unmanaged Pack complete.

Solution: bin\Debug\Deploy.zip generated.
Solution Package Type: Unmanaged generated.
Solution Packager log path: bin\Debug\SolutionPackager.log.
Solution Packager error level: Info.
Done Building Project "C:\Users\allan.de.castro\Desktop\PCFFolder\Deploy\Deploy.cdsproj" (build target(s)).

Build succeeded.
0 Warning(s)
0 Error(s)

Time Elapsed 00:00:25.24

C:\Users\allan.de.castro\Desktop\PCFFolder\Deploy>cd ../

C:\Users\allan.de.castro\Desktop\PCFFolder>
```

XrmToolBox Plugin by Danish



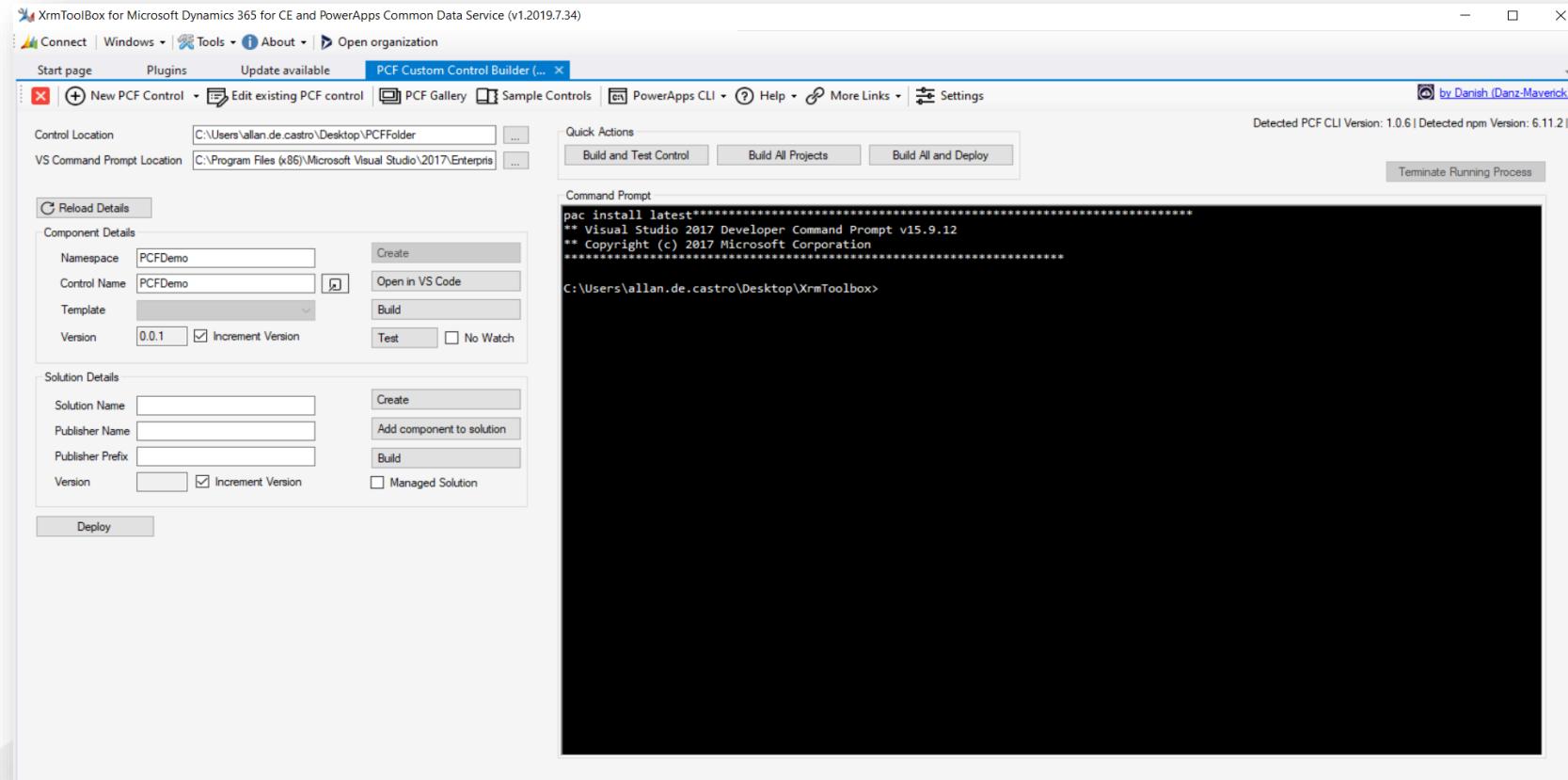
PCF Custom Control Builder

Version: 2.2019.10.22

Author: Danish Naglekar

Easily create, build and deployment solution for your custom control using PCF.

Facilitates the creation, building and deployment of custom PCF controls by eliminating the need to write commands.



What makes a custom component ?

Manifest Input File

Component Implementation

Resource Files

Control Definition:

- Name
- Version
- Properties
- Resource files

Code:

- TypeScript / Javascript
- User Interface
- Functionality

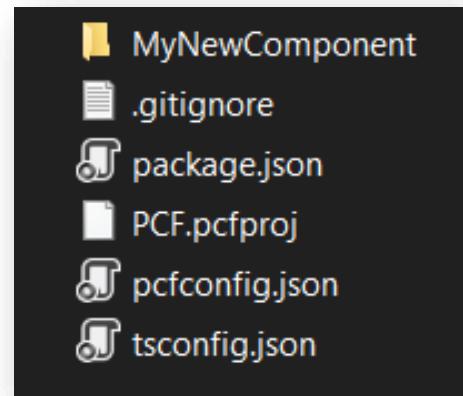
Static Resources

- CSS
- Images
- ...

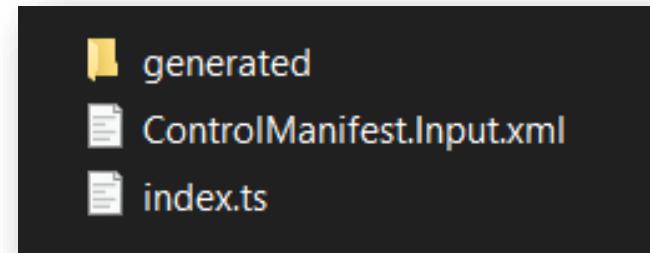
Creating a component

```
pac pcf init --namespace MyNewComponentNameSpace --name MyNewComponent --template field
```

Namespace for this component



Name of the component



Type of PCF project

Note: npm install to ensure to get required project dependencies.

Manage properties & resources

```
<?xml version="1.0" encoding="utf-8" ?>
<manifest>
  <control namespace="BusinessProcessFlowViewer" constructor="BusinessProcessFlowViewer" version="1.0.0" display-name-key="BusinessProcessFlowViewer">
    <data-set name="dataSet" display-name-key="DataSet View Used" cds-data-set-options="displayCommandBar:true;displayViewSelector:true" />
    <property name="parametersBPF" display-name-key="parametersBPF" description-key="This parameter is used to retrieve the different parameters for the Business Process Flow View." />
    <property name="completedColor" display-name-key="completedColor" description-key="This Hex Color define the background color of the completed entities." />
    <property name="completedTextColor" display-name-key="completedTextColor" description-key="This Hex Color define the text color of the completed entities." />
    <property name="activeColor" display-name-key="activeColor" description-key="This Hex Color define the background color of the active entities." />
    <property name="activeTextColor" display-name-key="activeTextColor" description-key="This Hex Color define the text color of the active entities." />
    <property name="notActiveColor" display-name-key="notActiveColor" description-key="This Hex Color define the background color of the not active entities." />
    <property name="notActiveTextColor" display-name-key="notActiveTextColor" description-key="This Hex Color define the text color of the not active entities." />
    <property name="progressTrackLineColor" display-name-key="progressTrackLineColor" description-key="This Hex Color define the background color of the progress track line." />
    <property name="pulseColor" display-name-key="pulseColor" description-key="This Hex Color define the pulse color. If empty pulse color will be the same as active color." />
    <property name="displayEntityName" display-name-key="displayEntityName" description-key="This value is used to display or not the entity name in the preview." />
    <resources>
      <code path="index.ts" order="1"/>
      <css path="css/style.css" order="1" />
      <img path='img/preview.png' />
    </resources>
    <feature-usage>
      <uses-feature name="WebAPI" required="true" />
    </feature-usage>
  </control>
</manifest>
```

Control	Web	Phone	Tablet
Read-only Grid (default)	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>
BusinessProcessFlowViewer	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>

Add Control...

BusinessProcessFlowViewer

activeTextColor *	#2183DD (SingleLine.Text)	<input type="button" value="edit"/>
notActiveColor *	#DFE3E4 (SingleLine.Text)	<input type="button" value="edit"/>
notActiveTextColor *	#666666 (SingleLine.Text)	<input type="button" value="edit"/>
progressTrackLineColor *	#DFE3E4 (SingleLine.Text)	<input type="button" value="edit"/>
pulseColor	#95c6e6 (SingleLine.Text)	<input type="button" value="edit"/>
displayEntityName	true (SingleLine.Text)	<input type="button" value="edit"/>

Life cycle of a component (index.ts)

Init: Initializes the control. Build the skeleton of its DOM and possibly call a logic to initialize what your component should do.

updateView: Called when the value of a property are changed (dataset refreshed..).

Destroy: Called when we decide to remove the component from the DOM.

getOutputs: Called when the output values are modified (usually the data is sent back to the application here, e.g a field that is modified).

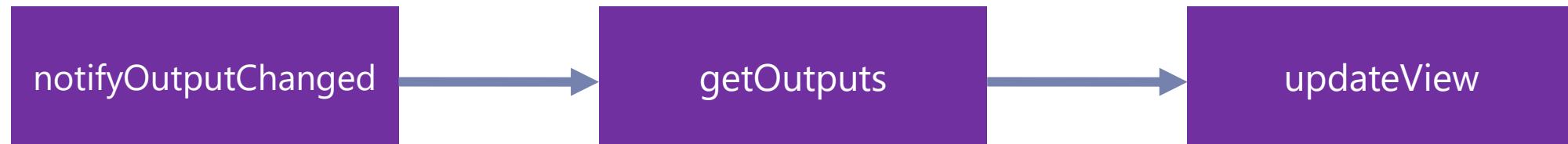
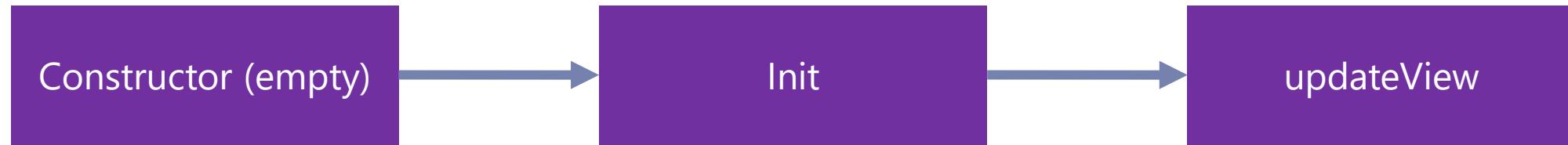
```
/* initialize component instance*/
init(context, notifyOutputChanged, state, container)

/* update call when value in the context changes */
updateView(context)

/* control is to be removed from the DOM tree*/
destroy()

/* updated values for the bound properties*/
getOutputs()
```

Life cycle of a component (index.ts)



Build & Debug

npm run build

npm start *watch* (or you can use `start npmstart` to open a new window)

Possibility to use fiddler:
[Debugging Canvas App Component](#)

The screenshot shows the PowerApps Component Framework Test Environment. On the left, there is a **TSDDataSetGrid** component displaying four data cards. Each card contains the following data:

name :	alex
address :	bellevue
email :	h@microsoft.com
balance :	0

name :	alex
address :	redmond
email :	in@microsoft.com
balance :	2

name :	susan
address :	bellevue
email :	n@ms.com
balance :	3

Below the grid, a context input dialog is open, showing a file selection interface. The path is `<< Desktop > Test_A11Y > Test_A11Y`. The file list contains:

Name	File ownership	Date modified	Type	Size
dscontroltest1	corp.microsoft.com	7/15/2019 2:18 PM	File folder	
dscontroltest2	corp.microsoft.com	7/15/2019 2:18 PM	File folder	
Accounts.csv	corp.microsoft.com	7/15/2019 2:06 PM	Microsoft Excel C...	
Catalog.csv	corp.microsoft.com	7/15/2019 2:06 PM	Microsoft Excel C...	

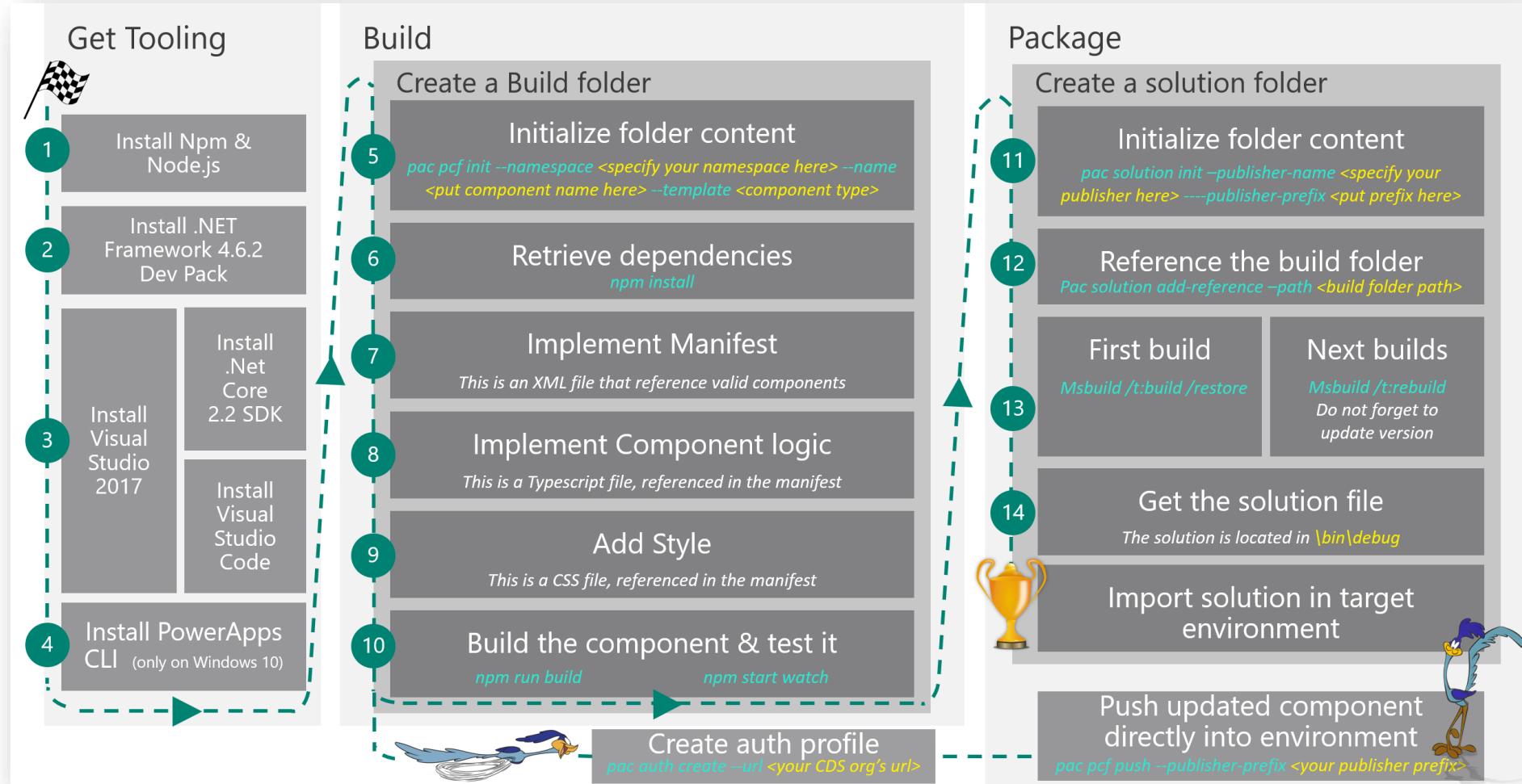
The right side of the screen displays the configuration for the **TSDDataSetGrid** component, showing settings for Context Inputs, Data Inputs, and specific columns like name, address, and email.

Package a Component & Import the solution

Use CLI to create a solution (Managed/UnManaged) containing the custom control.

1. `cd <solution folder>`
2. `pac solution init`-publisher-name <enter your publisher name> --publisher-prefix <enter your publisher prefix>
3. `pac solution add-reference` --path <path of your Power Apps component framework project on disk>
4. `msbuild /t:build /restore` (msbuild must be in the path)
5. `msbuild /t:build /restore /p:configuration=Release` (If you want to build a managed package)
6. `Import` the solution into Dataverse Environment using the UI.

How to build a PCF Component



Credit : Nicolas Kirrmann

Links / References 1/2

PCF forum: <https://aka.ms/PCFForum>

Documentation: <https://aka.ms/PCFDocs>

Feature ideas: <https://aka.ms/PCFIdea>

Community components/blogs/demos: <https://aka.ms/PCFDemos>

Business App Summit 2019: Microsoft PowerApps: Building and Extending Applications

Ignite 2019: Using Power Apps component framework to build reusable custom user experiences.

Ignite 2020 : Amplify your pro developer skills with Power Platform

Links / References 2/2

Implementing Security: <http://develop1.net/public/post/2019/11/21/time-to-add-some-finishing-touches-to-your-pcf-controls>

Debugging PCF for Canvas App: <http://develop1.net/public/post/2020/04/30/debugging-canvas-app-pcf-components-with-fiddler>

Using CSS in PCF: <https://benediktbergmann.eu/2020/01/04/pcf-css-for-model-driven-apps-md/>

Articles Series on my blog (<https://blog.allandecastro.com>):

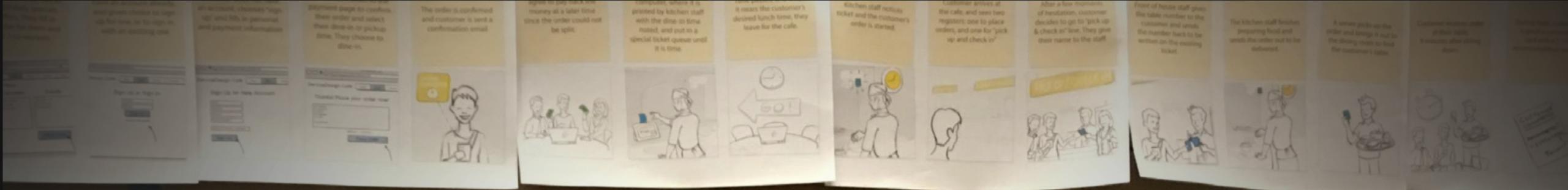
Prelude : [Getting started with PowerApps Component Framework](#)

Episode 1: [First Steps!](#)

Episode 2: [Focus on ... Control Manifest file!](#)

Episode 3: [Focus on ... Architecture & Component's life cycle!](#)

Episode 4 : [Walkthrough to create your first PCF \(based on a field\)!](#)



QnA

Ask me everything Component ideas?

Merci pour votre attention !



Thanks for your attention!

