#SPSBE

Understanding SharePoint Framework Extensions

Paolo Pialorsi | @PaoloPia

BIWUG

SharePoint Saturday Belgium 2018

Thanks to our sponsors!

Platinum

dox42®

FireStart BPM Suite

Gold

AMPLEXOR

devoteam

gfi

icefire STUDIOS

Quest | Metalogix Now part of Quest

REALDOLMEN to get there, together

sii BELGIUM Business Critical Digital Solutions

spikes DELIVERING PROGRESS

«ventigrate» your sharepoint and .net specialist

Xylos

Silver

advantive innovative people valuable solutions

ctg

skybow®

happit

u2u

vanroey.be

SharePint

Rencore

Community

EUROPEAN COLLABORATION SUMMIT

European SharePoint Office 365 & Azure Conference

Special Tribute Edition

#SPSBE

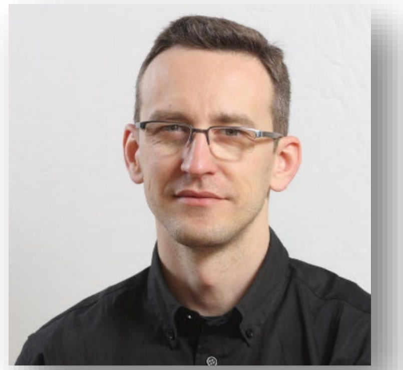
Patrick Tisseghem

Founding father of BIWUG & true SharePoint master

17/10/1968 - 3/9/2008

About me

- Project Manager, Consultant, Trainer
- About 50 Microsoft certification exams passed
  - MCSM – Charter SharePoint
  - MVP Office Apps and Services
  - Office 365 Dev PnP Core Team Member
- Focused on SharePoint and Office 365 since the beginning
- Author of many books about XML, SOAP, .NET, LINQ, SharePoint, and Office 365
- Speaker at main IT conferences



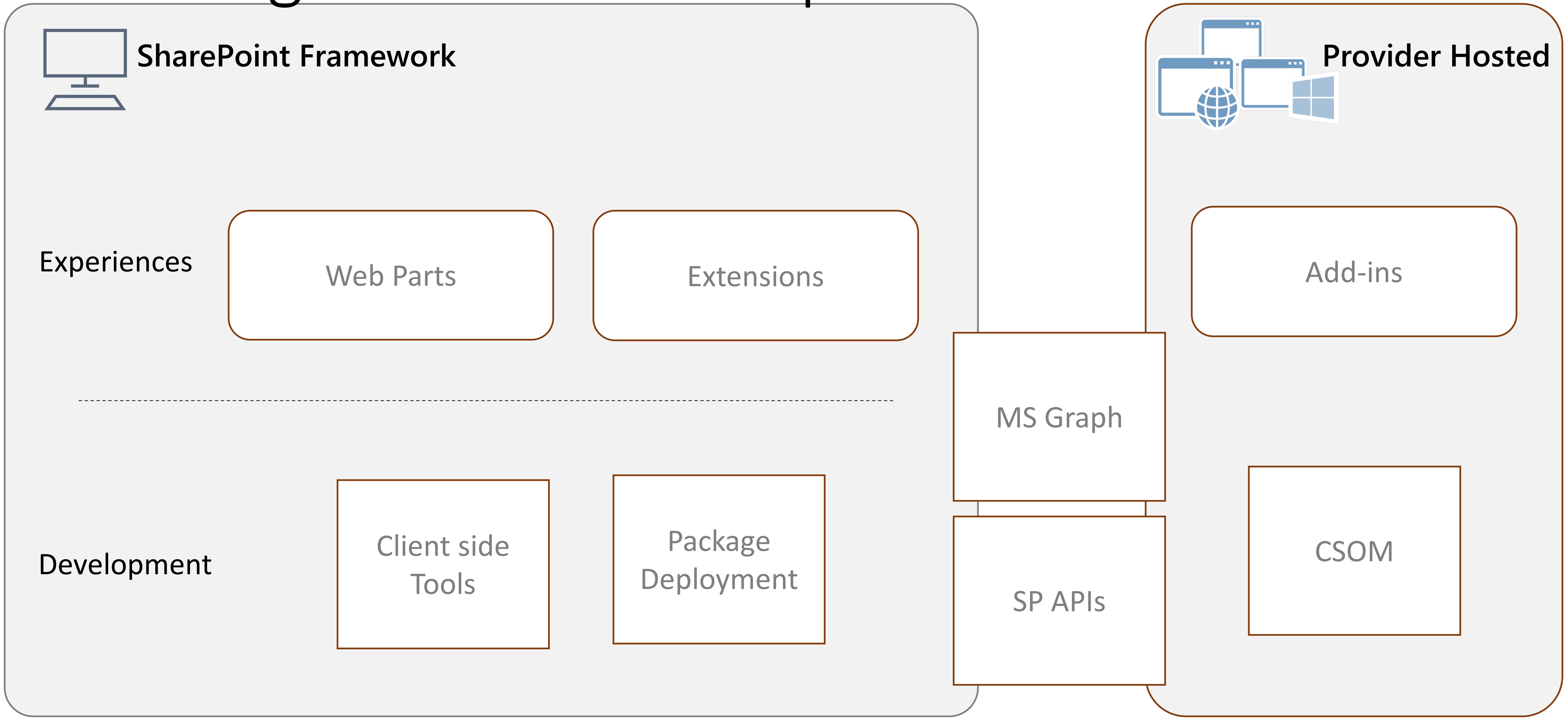
Microsoft SharePoint 2013 Developer Reference
Build Windows 8 Applications with Microsoft Visual C# and Visual Basic
Step by Step
Microsoft SharePoint 2010 Developer Reference
XML & Web Services
XML
Introducing Microsoft LINQ
Programmare con XML
Building Windows 8 Apps with Microsoft Visual C++
Programming Microsoft LINQ in .NET 4
Programming Microsoft LINQ
Programming Microsoft Office 365

Agenda

- Understanding the role of SharePoint Framework
- SharePoint Framework Extensions
  - Application Customizer
  - List View Command Set
  - Field Customizer
- SPFx Extensions Deployment
- Q&A

Understanding the role of SharePoint Framework

Building out the User Experience



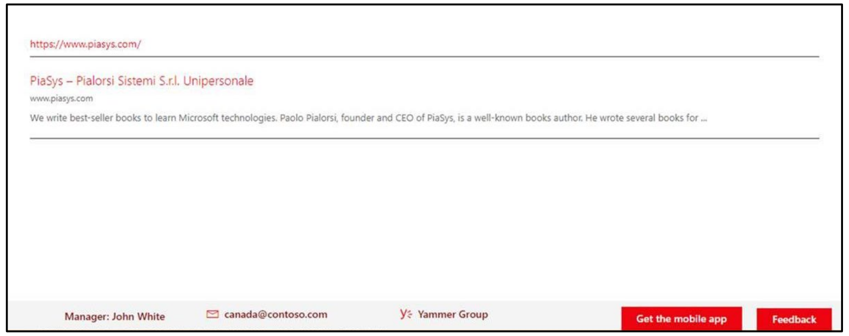
SharePoint Framework (SPFx)

- New framework to build Modern UI customizations
- Built on the well-known web stack
  - Open model, not only for .NET/Microsoft-oriented developers
- Works well on the cloud
  - It is available on-premises, too (SP2016 + FP2 or later)
- Enterprise-ready when used with back-end services
  - Microsoft Graph, REST API and micro-services, Azure Functions, etc.
- Not a new model, but an additional model to build UI elements for the Modern UI
  - We can build client-side Web Parts or client-side Extensions

SharePoint Framework Extensions

Client-Side Extensions

- Application Customizer
- Command Set
- Field Customizer



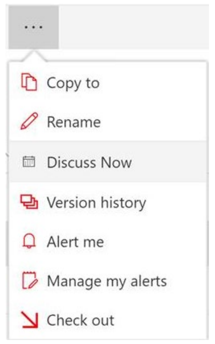
https://www.piasys.com/

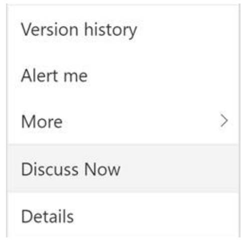
PiaSys – Pialorsi Sistemi S.r.l. Unipersonale

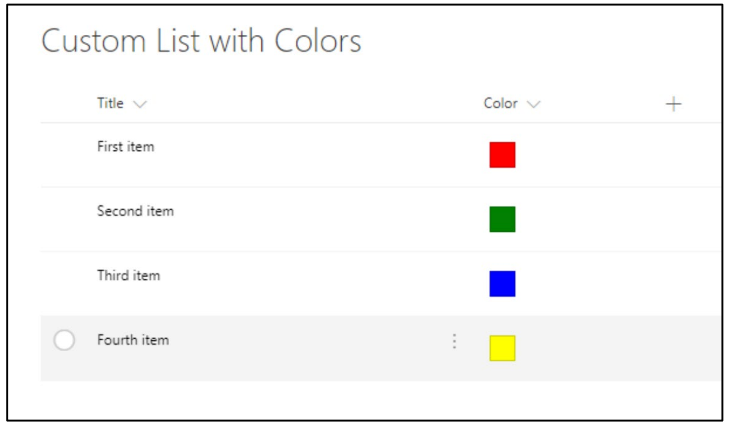
www.piasys.com

We write best-seller books to learn Microsoft technologies. Paolo Pialorsi, founder and CEO of PiaSys, is a well-known books author. He wrote several books for ...

Manager: John White | canada@contoso.com | Yammer Group | Get the mobile app | Feedback







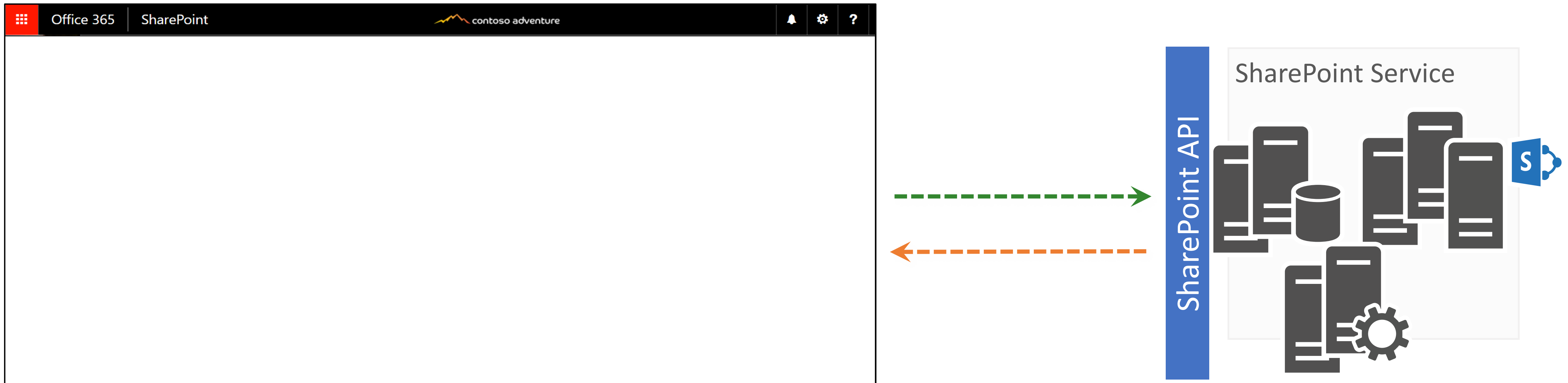
Custom List with Colors

Title	Color
First item	🟥
Second item	🟩
Third item	🟦
Fourth item	🟨

DEMO

SPFx Extensions Overview

Extensions Execution Logic

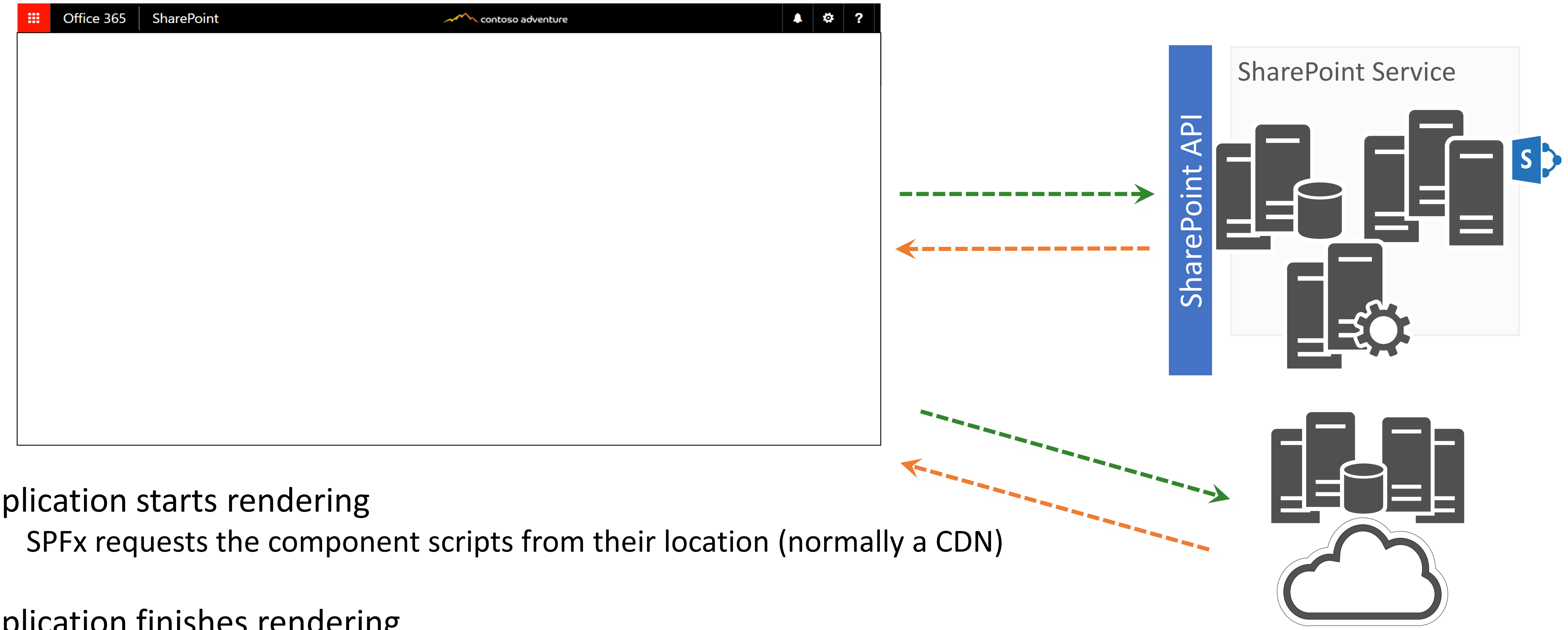


User navigates to the list, library or page

Server Returns:

Application Data (List data or Page data)
SPFx JavaScript Libraries
Manifest for all ***active*** SPFx components (extensions, web parts)

Extensions Execution Logic



Application starts rendering

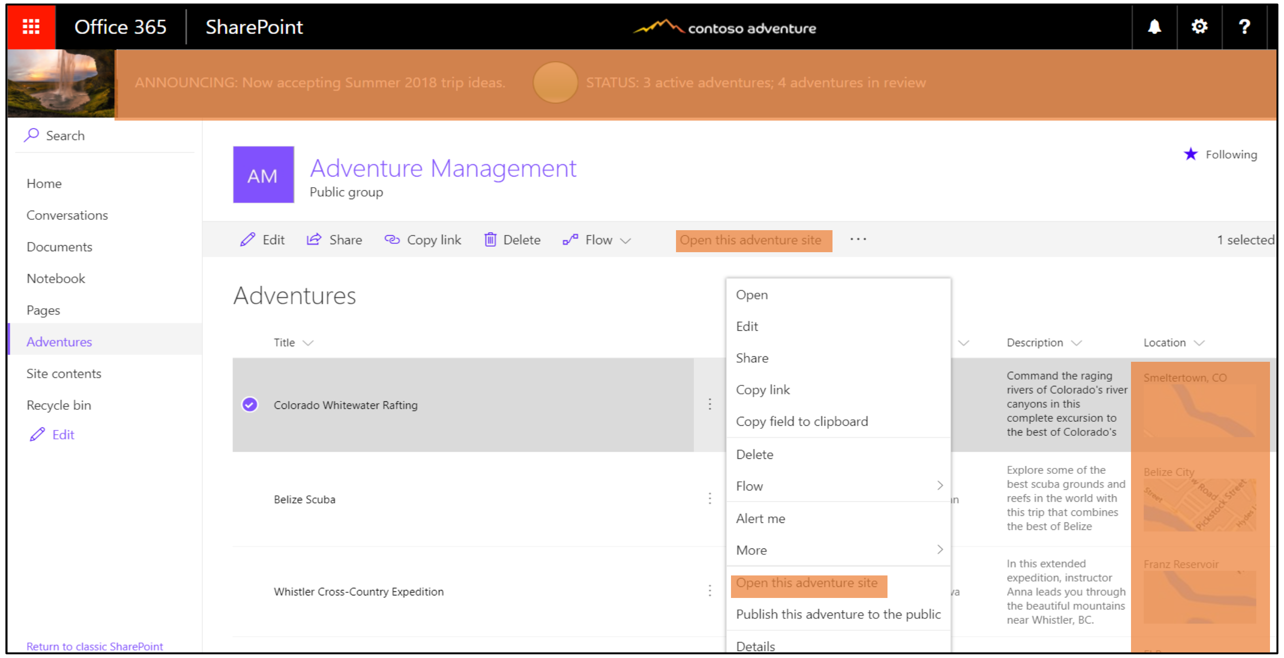
SPFx requests the component scripts from their location (normally a CDN)

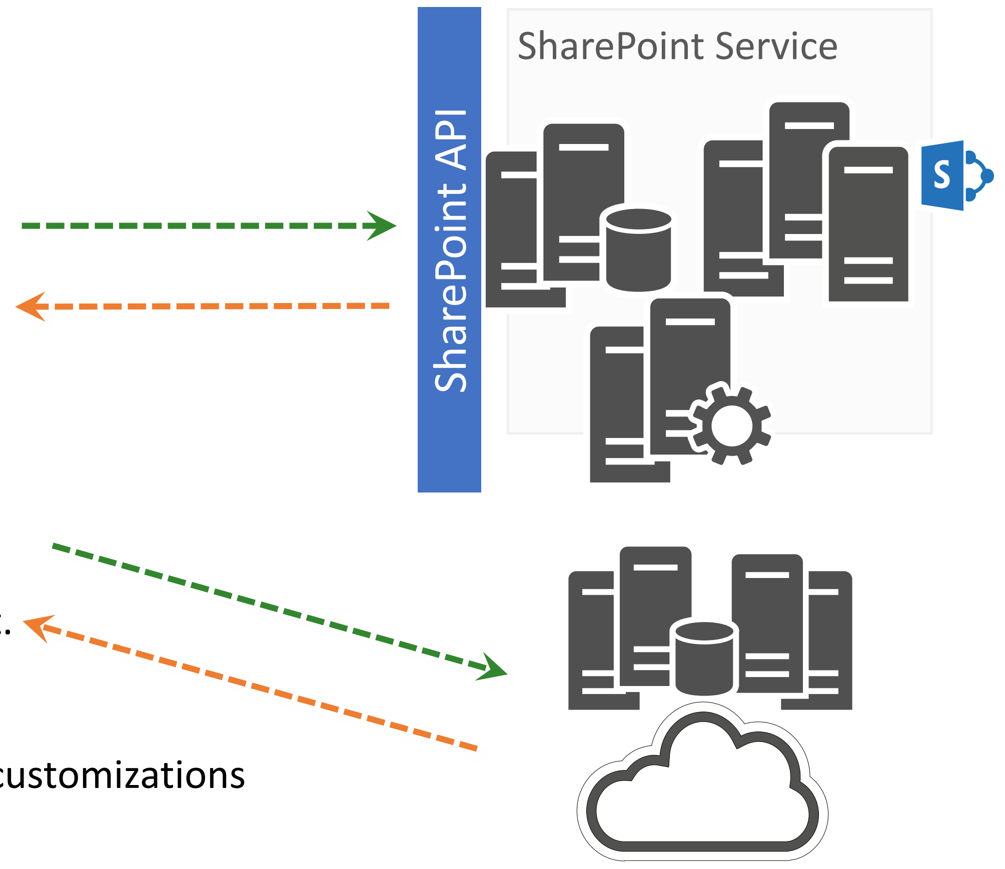
Application finishes rendering

Web parts and extensions are executed

Note: for field customizers data is not rendered until extension code is executed

Extensions Execution Logic


Office 365
SharePoint
contoso adventure
ANNOUNCING: Now accepting Summer 2018 trip ideas.
STATUS: 3 active adventures; 4 adventures in review
Search
Home
Conversations
Documents
Notebook
Pages
Adventures
Site contents
Recycle bin
Edit
Return to classic SharePoint
AM
Adventure Management
Public group
Following
Edit
Share
Copy link
Delete
Flow
Open this adventure site
1 selected
Adventures
Title
Description
Location
Colorado Whitewater Rafting
Belize Scuba
Whistler Cross-Country Expedition
Open
Edit
Share
Copy link
Copy field to clipboard
Delete
Flow
Alert me
More
Open this adventure site
Publish this adventure to the public
Details
Command the raging rivers of Colorado's river canyons in this complete excursion to the best of Colorado's
Explore some of the best scuba grounds and reefs in the world with this trip that combines the best of Belize
In this extended expedition, instructor Anna leads you through the beautiful mountains near Whistler, BC.
Smeltertown, CO
Belize City
Franz Reservoir



List View Command Sets can be used to introduce new custom actions to a list. Can be configured to be active when numerous items are selected.

Specific areas in the page are available for **Application Customizers** to embed customizations for end users. **Application Customizers** can be also invisible for the end users.

List View Field customizers can be used to customize experiences around the specific fields. They can be associated to a specific field instance to make a customization execute when it's used.

Application Customizer

- Build custom UI elements when **onInit()** fires
- Or when placeholders change
- Use **this.context.placeholderProvider** to read available placeholders
- Available placeholders (so far):
  - Top
  - Bottom
- Debug integrated with Visual Studio Code and SPFx generator scaffolding
  - Config serve.json + gulp serve --config=configName

Application Customizer Provisioning

```xml
<?xml version="1.0" encoding="utf-8"?>
<Elements xmlns="http://schemas.microsoft.com/sharepoint/">

    <CustomAction
        Title="SPFxApplicationCustomizer"
        Location="ClientSideExtension.ApplicationCustomizer"
        ClientSideComponentId="46606aa6-5dd8-4792-b017-1555ec0a43a4"
        ClientSideComponentProperties="{&quot;Top&quot;:&quot;Top area of the page&quot;,&quot;Bottom&quot;:
            &quot;Bottom area in the page&quot;}">

    </CustomAction>

</Elements>
```

- Reference the elements.xml file in package-solution.json

DEMO

Developing Application Customizers

Command Set

- Enable/Disable commands by overriding **onListViewUpdated(event)** method
- Override **onExecute(event)** method and switch based on **event.commandId**
- Available locations:
  - ClientSideExtension.ListViewCommandSet.ContextMenu: ECB
  - ClientSideExtension.ListViewCommandSet.CommandBar: Top menu
  - ClientSideExtension.ListViewCommandSet: Both
- Debug integrated with Visual Studio Code and SPFx generator scaffolding
  - Config serve.json + gulp serve --config=configName

Command Set Provisioning

```xml
<?xml version="1.0" encoding="utf-8"?>
<Elements xmlns="http://schemas.microsoft.com/sharepoint/">

  <CustomAction
    Title="SPFxListViewCommandSet"
    RegistrationId="100"
    RegistrationType="List"
    Location="ClientSideExtension.ListViewCommandSet.CommandBar"
    ClientSideComponentId="5fc73e12-8085-4a4b-8743-f6d02ffe1240"
    ClientSideComponentProperties="{&quot;sampleTextOne&quot;:&quot;One item is selected in the list.&quot;,
      &quot;sampleTextTwo&quot;:&quot;This command is always visible.&quot;}">
  </CustomAction>
</Elements>
```

- Reference the elements.xml file in package-solution.json

DEMO

Developing List View Command Sets

Field Customizer

- Override **onRenderCell(event)** method
  - Play with **event.domElement** to replace/update the HTML of the cell
- Debug integrated with Visual Studio Code and SPFx generator scaffolding
  - Config serve.json + gulp serve --config=configName
- Override **onDisposeCell(event)** method
  - To free any resources that you don’t need anymore

Field Customizer Provisioning

```
<?xml version="1.0" encoding="utf-8"?>
<Elements xmlns="http://schemas.microsoft.com/sharepoint/">

   <Field ID="{060E50AC-E9C1-3D3C-B1F9-DE0BCAC200F6}"
          Name="SPFxPercentage"
          DisplayName="Percentage"
          Type="Number"
          Min="0"
          Required="FALSE"
          Group="SPFx Columns"
          ClientSideComponentId="7e7a4262-d02b-49bf-bfcb-e6ef1716aaef">
   </Field>
</Elements>
```

- **Reference the elements.xml file in package-solution.json**

DEMO

Developing Field Customizers

SPFx Extensions Deployment

SPFx Extensions Deployment

- Relies on SharePoint Feature Framework for deployment
  - *<CustomAction />* or *<Field />* elements
- Tenant-wide deployment doesn’t apply Feature Framework XML
  - But there is the *<ClientSideComponentInstance />* element for tenant-wide extensions (SPFx 1.6)
- Thus, to deploy SPFx Extensions you have two options
  - Use non-tenant-wide deployed packages
  - Use tenant-wide packages and
    - **Tenant-wide extensions (since SPFx v. 1.6)**
    - PowerShell
    - CSOM
    - PnP

What are tenant-wide extensions?

- Extensions available across the whole tenant
  - Available starting from SPFx v. 1.6
- Supported for
  - Application Customizer
  - List View Command Set
- Can be filtered based on target
  - Web template
  - List definition

Main steps

- Create a new SPFx 1.6 project
  - Scaffolding will generate sample files to activate tenant-wide deployment
    - You can always remove them
  - Enable Tenant Wide Deployment (i.e. *skipFeatureDeployment: true*)
- Develop your extensions
- Deploy the .sppkg on the tenant *App Catalog*
- Register them in the *Tenant Wide Extensions* list in the tenant *App Catalog*
  - Eventually filter by target *WebTemplateId* or *ListTemplateId*

DEMO

Tenant-wide Extensions with SPFx 1.6

Q&A

Some shameless marketing ☺

- Follow me on Twitter:
  - https://www.twitter.com/PaoloPia
- Subscribe to the “PiaSys Tech Bites” YouTube channel:
  - https://piasys.com/TechBites
- Don’t miss the upcoming trainings
  - https://piasys.com/shop/ (New schedule for 2019 is coming soon …)
- And the upcoming events:
  - https://piasys.com/events/

#SPSBE

Please rate this session!

http://spsbe.be

#SPSBE

BIWUG

SharePoint Saturday Belgium 2018