



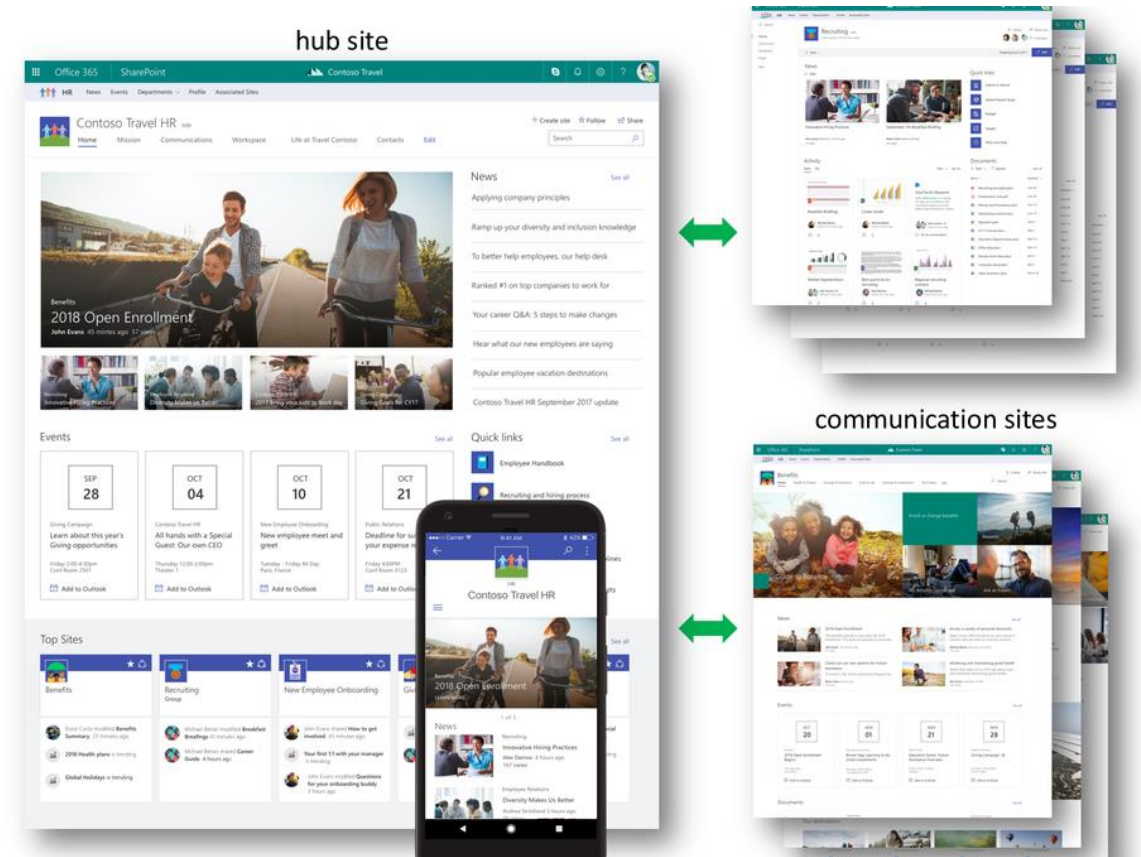
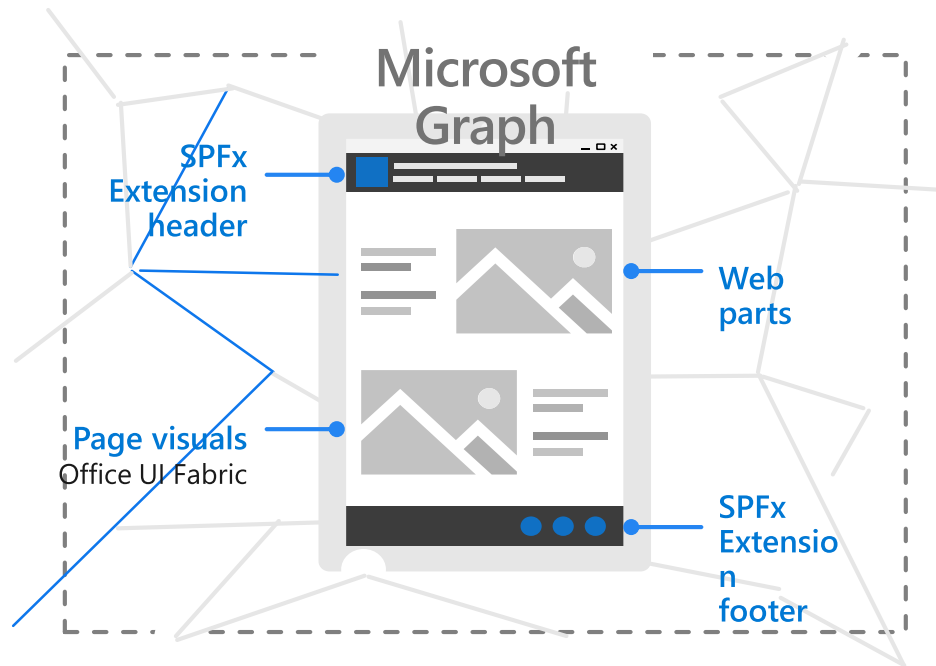
Thriveconference

Speaker Gašper Rupnik
Email gasper.rupnik@kompas-xnet.si
Social @RaspeR87; <https://rasper87.wordpress.com>

Provision SPFx Web Parts to Classic Sites

SharePoint Framework (SPFx)

- An application framework used to build and extend modern experiences in SharePoint



#ThriveITConf

SharePoint Framework (SPFx)

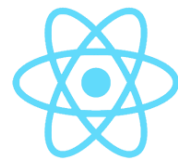
- Good
 - Modern client-side development (no more script injection)
 - Works in Classic and Modern Pages / On-Prem and Online
 - Local Workbench for development and debugging
 - Mobile friendly
- Bad
 - Change is hard if you are .NET developer
 - You have to learn new skills

SharePoint Framework (SPFx)

- Skills you need to embrace
 - Javascript/HTML/CSS
 - TypeScript
 - React, Angular, KnockoutJS ...
 - Node.js, NPM, Yeoman, Gulp
 - Office UI Fabric (Front-end visual framework)
 - Etc.



TypeScript



React



Visual Studio Code

SharePoint Framework (SPFx)

- Supported SP versions
 - SP 2016 Feature Pack 2
 - SP 2019
 - SP Online
- Supported SPFx versions
 - SP 2016 FP2 | SPFx 1.1 (Web Parts Only)
 - SP 2019 | SPFx 1.4 (Web Parts & Extensions)
 - SP Online | SPFx 1.7 - latest (Web Parts & Extensions and more ...)

Latest SPFx 1.7 features:

- Dynamic Data
- Official teams tabs support
- Full page apps
- ...

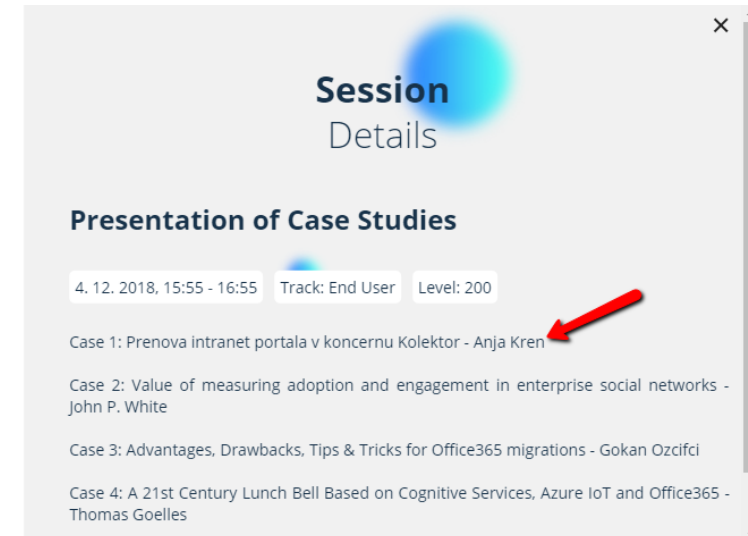
Situation in Slovenia (our customers)

- SharePoint 2003
- SharePoint 2007
- SharePoint 2010
- **SharePoint 2013**
- **SharePoint 2016**
- SharePoint 2019
- SharePoint Online

.NET

SPFx

15:55 - 16:55
Presentation of
Case Studies

Session Details

Presentation of Case Studies

4. 12. 2018, 15:55 - 16:55 Track: End User Level: 200

Case 1: Prenova intranet portala v koncernu Kolektor - Anja Kren

Case 2: Value of measuring adoption and engagement in enterprise social networks - John P. White

Case 3: Advantages, Drawbacks, Tips & Tricks for Office365 migrations - Gokan Ozcifci

Case 4: A 21st Century Lunch Bell Based on Cognitive Services, Azure IoT and Office365 - Thomas Goelles

#ThriveITConf

DEMO

Provision SPFx Web Part to SP (by hand)

#ThriveITConf

Provision SPFx Web Part to SP (by hand)

- Four step Provisioning process:
 - Copy SPFx Web Part Assets to “local CDN”
 - Copy SPFx Web Part Package to App Catalog
 - Add SPFx App to Web
 - Add SPFx Web Part to Page

Provision SPFx to SP (globally)

- ALM API
 - Only for SharePoint Online & 2019
 - No way to programmatically provisioning Apps to SP2016
 - REST API: `/_api/web/tenantappcatalog/Add(overwrite=true, url='test.txt')`
- PnP Provisioning Engine
 - Leverages the ALM API
 - PnP Core Library `web.GetAppCatalog();`
 - PnP PowerShell `Get-PnPTenantAppCatalogUrl`
 - <https://aka.ms/sppnp>

Available operations:

- Upload a package to the AppCatalog
- Install an app
- Deploy/Trust an app
- Upgrade an app
- Uninstall an app
- Retract an app
- Remove an app
- List available apps

Provision SPFx to SP (2016 FP2)

- What we have OOTB for SP 2016 On-Prem?
 - Nothing. 😞
- What now – any workaround?
 - Upload .sppkg file into App Catalog -> **Module Element** + **Feature Event Receiver**
 - Upload assets to local CDN Library -> **Module Element**
 - Install SPFx App to specific SPWeb -> problem 😞
 - Add SPFx Web Part to Page -> ?
- Any workaround?

Provision SPFx to SP (2016 FP2)

- **LoadAndInstallApp** -> app deployment (from SP 2013 era) but doesn't work for SPFx ... really?
 - Make **.sppkg** file compatible with LoadAndInstallApp method
 - Modifications to **AppManifest.xml** within .sppkg
 - Method only works on Root Web of specific Site Collection (no-go with Sub-Webs)
 - *Error: A different version of this App is already installed with the same version number.*
- Any workaround?

Provision SPFx to SP (2016 FP2)

- **SPApp** class inside **Microsoft.SharePoint.dll**
 - Internal method **CreateAppUsingPackageMetadata**
 - Get SPApp object of specific SPFx Web Part from App Catalog
 - Public method **CreateAppInstance**
 - Create instance of SPApp (**SPAppInstance**) to specific SPWeb
 - Call **Install()** to install it

Provision SPFx to SP (2016 FP2)

- Add this code into Web-Scope Feature Event Receiver -> **FeatureActivated**
 - Works fine with manually activation via **Manage Site Features**
 - What about when we include this feature into **Web Template** -> problem ☹
- We create Site Collection inside of **Central Administration** -> code runs under **System Account**
 - *Error: The System Account cannot perform this action.*
- ***RunWithElevatedPrivileges*** (*impersonation*)
 - Person with Site Admin permission -> *same error*
 - Person with less than Site Admin permission -> *Error: Only site administrators may install or uninstall Apps.*
- Any workaround?

Provision SPFx to SP (2016 FP2)

- **SPAppAllowRunAsSystemAccountScope** inside of **SPSecurity** class (from **Microsoft.SharePoint.dll**)
 - Disposable object
 - It set **AllowRunAsSystemAccount == true**
- Any additional problems? 😊

Provision SPFx to SP (2016 FP2)

- **CreateAppUsingPackageMetadata** throw exception if SPFx App is already installed on any SPWeb of current SPSite
 - Error code: -2146232828
 - Go through all SPWebs -> find first **SPAppInstance** of specific **SPApp**
 - Create new **SPAppInstance** from this one
- Any additional problems?
 - **No, finally!** 😊

DEMO

Provision SPFx Web Part to SP2016

#ThriveITConf

Demo steps

- [Include SPFx Assets & Package inside of WSP](#)
- [Deploy SPFx Web Part to SharePoint Server 2016 App Catalog with WSP](#)
- [Install SPFx Web Part to SharePoint Site/Web inside Feature](#)
- [Include SPFx Web Part inside of Web Template](#)
- [Run all SPFx Gulp tasks with one custom PowerShell function](#) (additional)
- [Dynamic External JS References in SPFx Web Parts](#) (additional)



THANK YOU

gasper.rupnik@kompas-xnet.si

@RaspeR87

<https://rasper87.wordpress.com>

#ThriveITConf