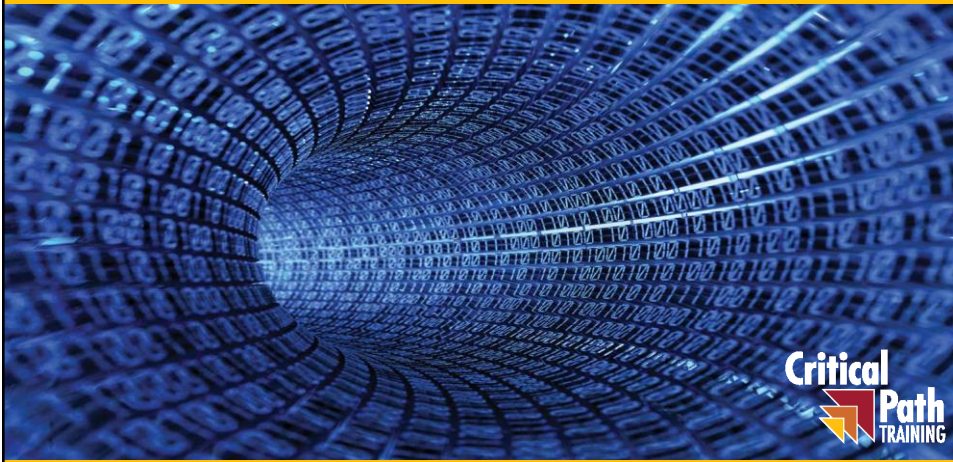


Developing with Power BI Premium

By Ted Pattison



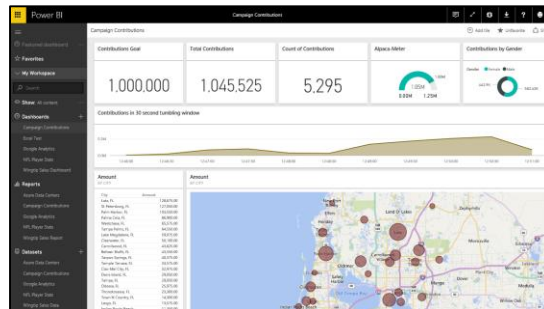
Agenda

- Introduction to Power BI Embedding
- App Workspaces and Premium Capacities
- Developing with 3rd Party Power BI Embedding



The Power BI Service

- Provides cloud-based foundation for Power BI platform
- Accessible through <https://app.powerbi.com>
- Provides browser-based experience for licensed users
- Provides programmatic API for custom development



Power BI Licensing

- Microsoft initially offered two Power BI licensing options
 - Power BI Free license
 - Power BI Pro license (\$10/month)
 - All Power BI resources and processing runs in shared capacity
- Microsoft recently introduced Power BI Premium licensing
 - Power BI Premium customers can create premium capacities
 - Premium capacities useful to organization with many read-only users
 - Premium capacities used by ISVs to reach non-licensed users
- Power BI Premium Details and Pricing are in flux
 - More info at <https://powerbi.microsoft.com/en-us/pricing/>

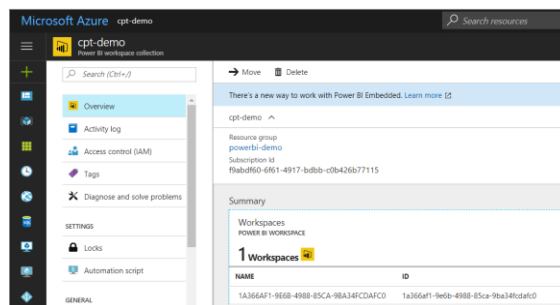
Developer Opportunities in Power BI

1. Designing App Workspaces and Publishing Apps
2. Developing Custom Visuals
3. Programming the Power BI Service API
4. Embedding Power BI Content in Websites



What **is** Power BI Embedded V1?

- Power BI Embedded V1 is an Azure Service
 - PBI Embedded service can be provisioned on-demand
 - Service provisioned in terms of workspace collections
 - PBI Embedded service requires an Azure subscription
 - Pricing model based on number of report sessions



Reflecting on Power BI Embedded V1?

- Good Points about Power BI Embedded V1
 - It eliminates need for Power BI license for each user
 - It decouples user security from app security
 - It opens up PBI platform to commercial applications
- Pain Points with Power BI Embedded V1
 - Requires developers to have Azure subscriptions
 - No out-of-box UX to upload and manage PBIX files
 - It uses separate APIs from Power BI Service API
 - Cannot estimate costs with per-session pricing model
 - It's deprecated and not available to new customers



Power BI Embedded Version 2

- Power BI Embedded V2 has same good points as V1
 - It eliminates need for Power BI license for each user
 - It decouples user security from app security
 - It opens up PBI platform to commercial applications
- Power BI Embedded V2 significantly improves upon V1
 - Embedding features merged into Power BI service
 - Developers now use Power BI Service API for embedding
 - Standard PBI UX used to upload and manage PBIX files
 - New pricing models allow for predictable costs per month
 - No need to create, manage and monitor any Azure services
- The term “Power BI Embedded” is now ambiguous
 - Better to refer to the “Embedding features in Power BI”



Embeddable Resources

1. Reports

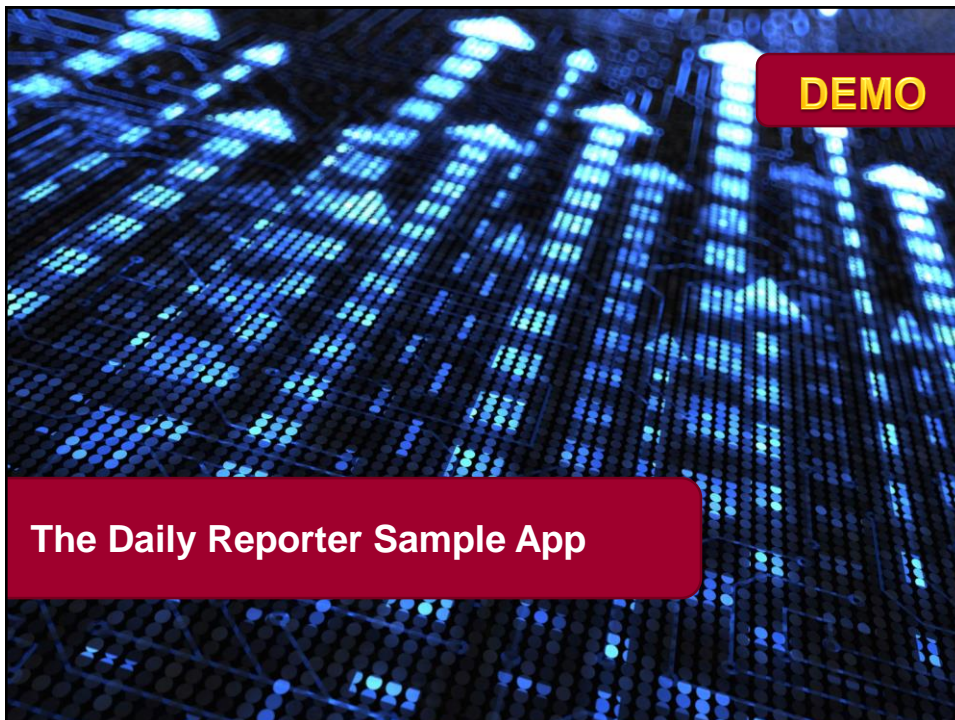
- Provides user with full interactive experience
- Allows editing existing reports & creating new reports

2. Dashboards

- Provides user with limited interactive experience
- Provides support for real-time dashboards

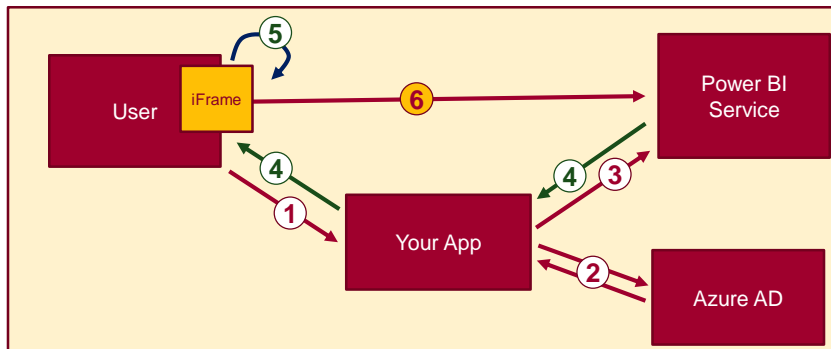
3. Dashboard Tiles

- Provides flexibility to embed selected tiles
- No support for tiles which receive real-time updates



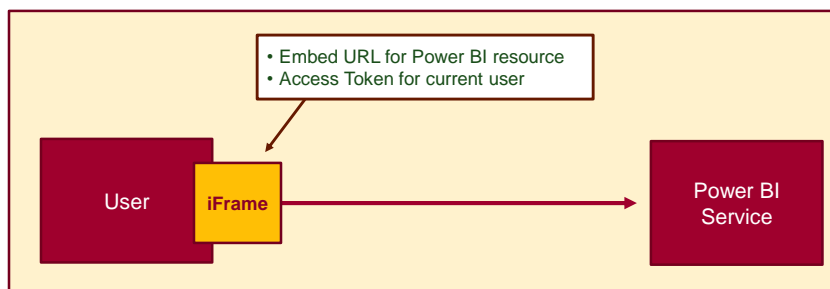
Power BI Embedding – The Big Picture

1. User launches your app using a browser
2. App authenticates with Azure Active Directory and obtains access token
3. App uses access token to call to Power BI Service API
4. App retrieves data for embedded resource and passes it to browser.
5. Client-side code uses Power BI JavaScript API to create embedded resource
6. Embedded resource session created between browser and Power BI service



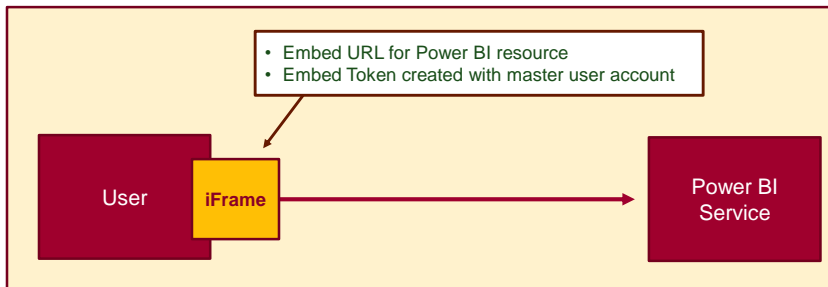
First Party Embedding

- App authenticates current user with Azure AD
 - Your code accesses Power BI Service as current user
 - Embedding requires Azure AD access token for user
 - User requires Azure AD account and Power BI license
 - Your code has access to whatever user has access to



Third Party Embedding

- App authenticates using Master User Account
 - Your code accesses Power BI Service as master user
 - Embedding uses embed token instead of access token
 - Users don't need AAD accounts and Power BI licenses
 - Your code has access to whatever master has access to



First Party vs Third Party Embedding

- What scenarios use first party embedding?
 - Organizations where users have Power BI licenses
 - Users can already access Power BI with browser
 - Development should go beyond out-of-box experience
- What scenarios use third party embedding?
 - Scenarios where users don't have Power BI licenses
 - Applications which have custom identity providers
 - Applications which use identity provider other than AAD



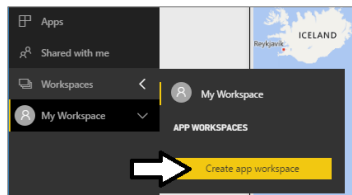
Agenda

- ✓ Introduction to Power BI Embedding
- App Workspaces and Premium Capacities
- Developing with 3rd Party Power BI Embedding



Understanding App Workspaces

- App workspaces required for team-based development
 - Assets in personal workspaces can only be edited by one person
 - Assets in app workspace can be edited by team members
- App workspaces provide secure, updatable deployment
 - App workspace can be secured using private membership



Create an app workspace

Name your workspace
Test Pattern

Workspace ID

Subscription
Available

Private - Only approved members can see what's inside

Members can edit Power BI content

Add workspace members

Enter email addresses

criticalpathus@20powerbi.com... Admin

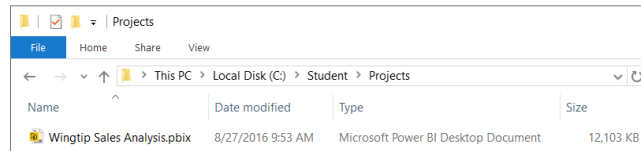
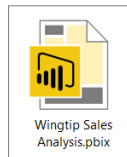
Advanced

Save Cancel



Projects and PBIX Files

- Power BI Desktop projects saved using PBIX files
 - PBIX file contains data source definitions
 - PBIX file contains query definitions
 - PBIX file contains data imported from queries
 - PBIX file contains exactly one data model definition
 - PBIX file contains exactly one report
 - PBIX file never contains data source credentials



Premium Capacities

- Power BI workspaces run in two possible environments
 - Shared Capacities
 - Premium Capacities (*formerly known as dedicated capacities*)
- Premium capacity acts as dedicated resource
 - Premium capacity only used by single organization
 - PBIX file uploads not limited to 1GB
 - Data refresh frequency can exceed 8 times per day
 - Each premium capacity defines its own set of admins

Premium Capacity Nodes

- Power BI Premium Purchased using Nodes
 - Node type defines v-core and RAM capabilities
 - P nodes used for embedded or service deployments
 - EM nodes used only for embedded deployments

Capacity Node	Total cores	Backend Cores	Frontend Cores	Direct Query Limits	Page renders/hour
EM1	1 v-cores	.5 cores, 3GB RAM	.5 cores		1-300
EM2	2 v-cores	1 core, 5GB RAM	1 core		301-600
EM3	4 v-cores	2 cores, 10GB RAM	2 cores		601-1,200
P1	8 v-cores	4 cores, 25GB RAM	4 cores	30 per second	1,201-2,400
P2	16 v-cores	8 cores, 50GB RAM	8 cores	60 per second	2,401-4,800
P3	32 v-cores	16 cores, 100GB RAM	16 cores	120 per second	4,801-9600



Agenda

- ✓ Introduction to Power BI Embedding
- ✓ App Workspaces and Premium Capacities
- Developing with 3rd Party Power BI Embedding



Authentication Flows

- User Credentials Flow (*public client*)
 - Used in Native clients to obtain access code
 - Requires passing user name and password
- Implicit Grant Flow (*public client*)
 - Used in SPAs built with JavaScript and AngularJS
 - Application obtains access token w/o acquiring authorization code
- Authorization Code Grant Flow (*confidential client*)
 - Client first obtains authorization code then access token
 - Server-side application code never sees user's password
- Client Credentials Grant Flow (*confidential client*)
 - Authentication based on SSL certificate with public-private key pair
 - Used to obtain access token when using app-only permissions

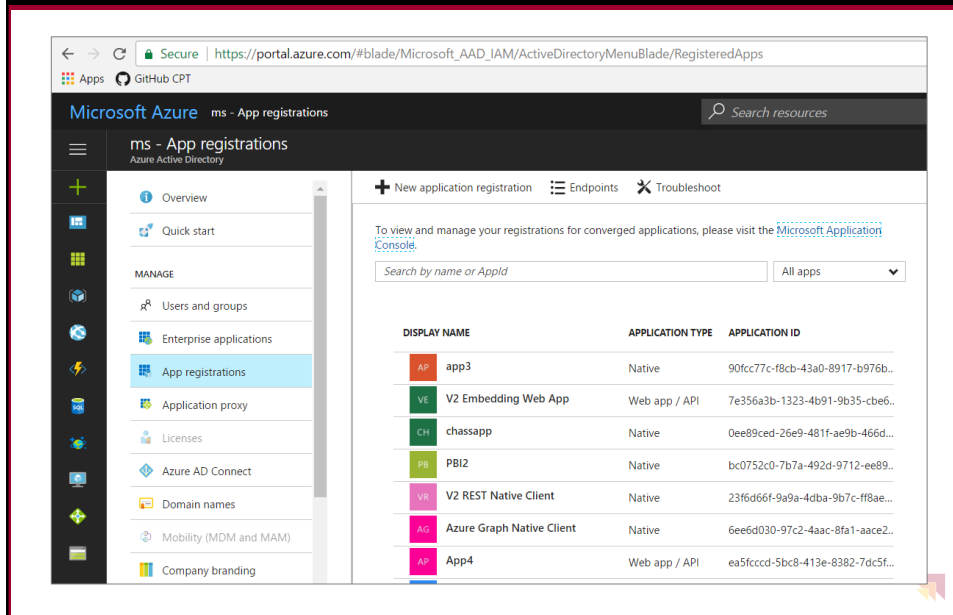


1st Party Embedding vs 3rd Party Embedding

	1st Part Embedding	3rd Party Embedding
Authentication flow	Authentication Code Flow or Implicit Flow	Direct User Credentials
Identity used to call Power BI	Current User	Master User Account
Access to personal workspace	Yes	No
Access to app workspaces	Yes	Yes
Ability to reach non-licensed users	No	Yes



App Registration in the Azure Portal



Embedding Internals

- How are Power BI Reports Embedded?
 - It's done using an iFrame
 - iFrame requires event to load embed token

```
<div id="pbi-report"
  style="height:95vh;"
  powerbi-type="report"
  powerbi-embed-url="https://embedded.powerbi.com/appTokenReportEmbed?reportId=8c6ec229-f3
  powerbi-access-token="eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9.eyJ2ZXN1IjI0IiwiaWF0IjIuMCIs
  <iframe
    src="https://embedded.powerbi.com/appTokenReportEmbed?reportId=8c6ec229-f3
    scrolling="no"
    allowfullscreen="true"
    style="width:100%;height:100%;"
  </iframe>
</div>
```

Power BI JavaScript API

- Used to generate iFrame required for embedding
- Call to **powerbi.embed** creates iFrame for embedding

```
var reportConfig = {  
  settings: {  
    filterPaneEnabled: false,  
    navContentPaneEnabled: false  
  }  
};  
  
var reportContainer = document.getElementById("pbi-report");  
var report = powerbi.embed(reportContainer, reportConfig);
```

- API supports binding to report-level events

```
report.on('loaded', onReportLoaded);  
report.on('pageChanged', onReportPageChanged);  
  
function onReportLoaded()...  
  
function onReportPageChanged(e)...
```



DEMO

The Daily Reporter Pro Demo App

Summary

- ✓ Introduction to Power BI Embedding
- ✓ App Workspaces and Premium Capacities
- ✓ Developing with 3rd Party Power BI Embedding



Critical Path Training

<https://www.CriticalPathTrainig.com>

- **PBI365: Power BI Boot Camp – 4 Days**
 - Audience is Business Users, Analysts and Data Professionals
 - Provides hands-on introduction to the Power BI platform
 - Focuses on build solutions using Power BI Desktop
 - Query design, data modeling and report and dashboard design
 - Apps and App Workspaces
 - Learn about “import” vs “connect to” with Excel workbooks
- **PBD365: Power BI Developer Boot Camp – 4 Days**
 - Audience is Professional Developers
 - Teaches developing custom visuals with TypeScript and D3
 - Teaches R programming and integrating R with Power BI
 - Teaches programming with the Power BI APIs
 - Teaches developing with Power BI Embedded

