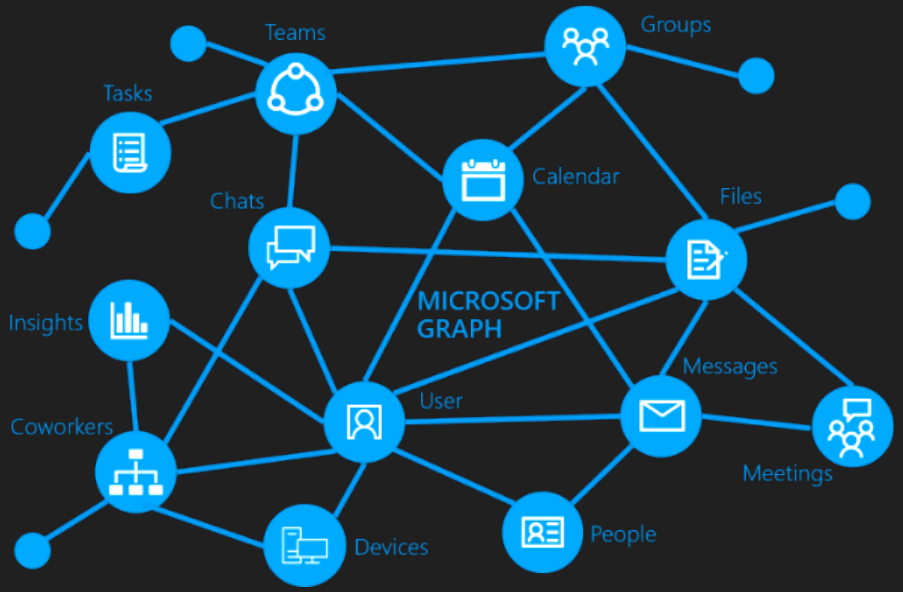
SYNERGY
DEVPARTNER CONFERENCE 2018
OCT 8-12 NEW ORLEANS, LA

Leveraging Microsoft Graph

Presented by Jeff Greene

Microsoft Graph

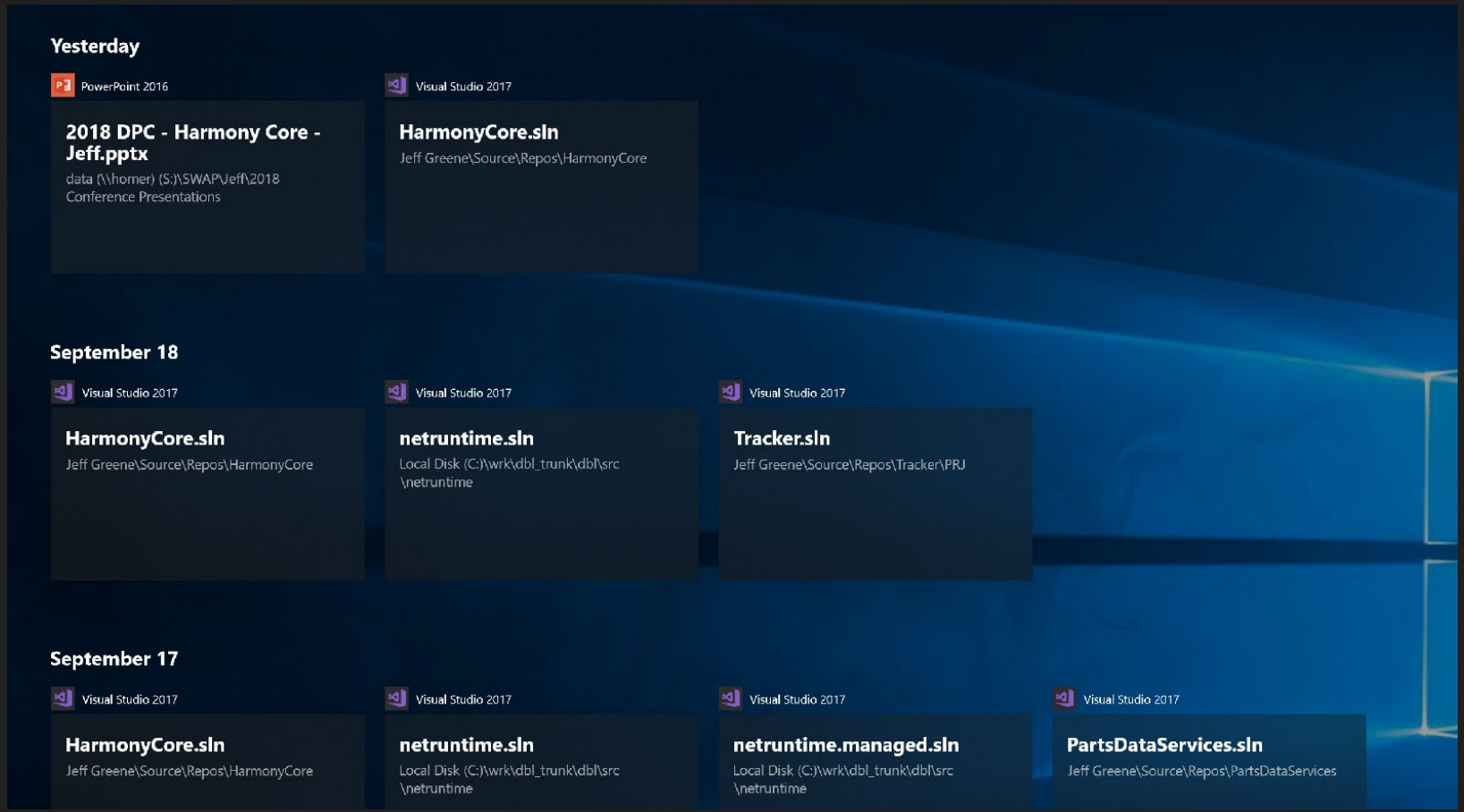
"Microsoft Graph is the gateway to data and intelligence in Microsoft 365. Microsoft Graph provides a unified programmability model that you can use to take advantage of the tremendous amount of data in Office 365, Enterprise Mobility + Security, and Windows 10."



Activity Feed

“Microsoft helps drive user productivity with your apps through experiences like Windows Timeline, Windows Sets, Cortana Pick up Where I left off and Microsoft Launcher, which are all powered by the activity feed.”

Activity Feed



Use Activity Feed for This

- **Record a single activity for a group of related user actions.** If your application is used for a sequence of related content, it probably makes sense to record a single activity for the entire engagement session.

- If the same item is changed multiple times, **include multiple history items** to represent the repeated user engagement.

- S**tore user data to the cloud.** If you want to support cross-device activities, you need to make sure the content required to re-engage the activity is stored to a cloud location. For example, if you publish an activity each time a user edits a document, the document should be stored in the cloud as opposed to locally on the user's device in order to enable cross-device re-engagement.

Don’t Use Activity Feed Like This

- Create a user activity for actions that users do not need to resume in the future. If your application is used to complete simple, one-time operations that do not persist status for you to track in the future, you probably do not need to write a user activity.

- Create a user activity for actions completed by *other users*. If someone sends the user a message, or @mentions the user within your app, you should not write a new activity. These interactions are better served by surfacing notifications.

Pick Up Where You Left Off

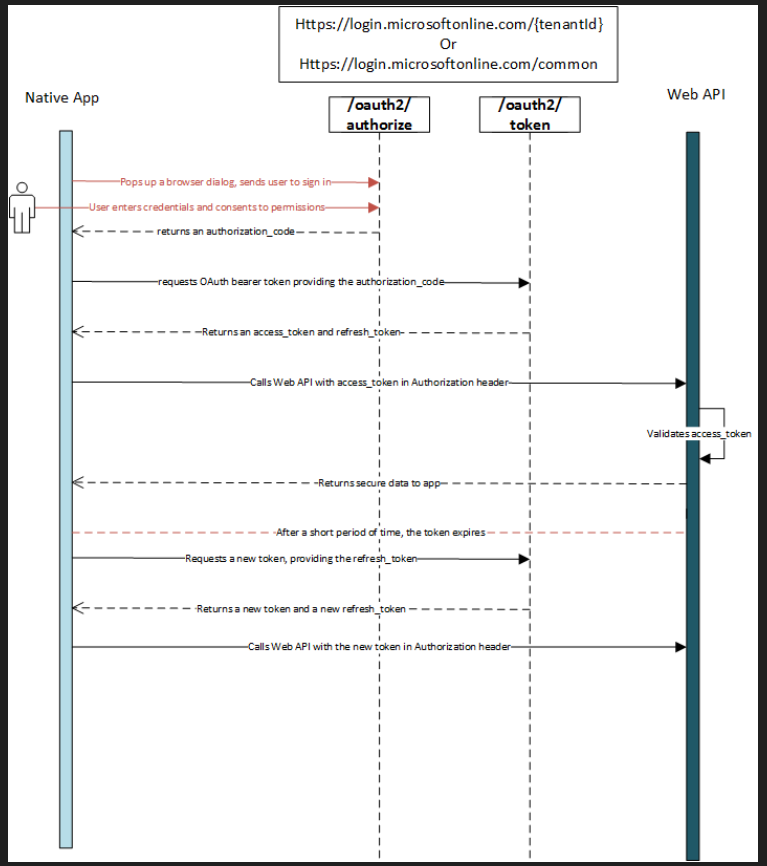
- Go back to an item you were working on previously
  - Needs some kind of unique ID so your application knows how to get back to the item the user wants

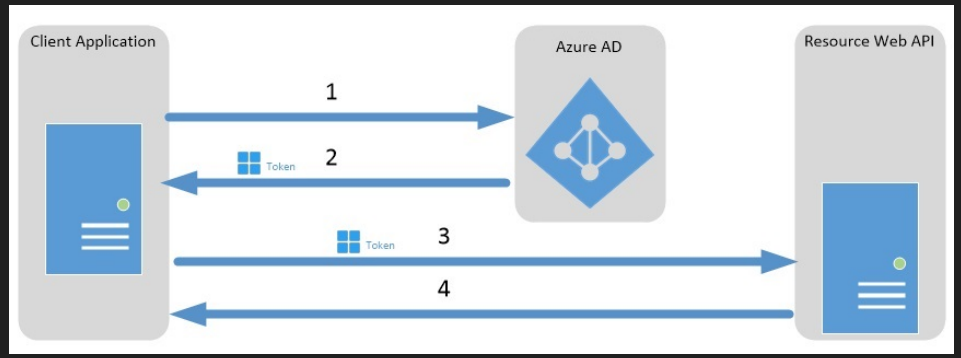
- Go back to an unfinished item
  - Needs a unique ID, but you probably need to store the unfinished work somewhere so you can look it up by ID

Security

- **Register the application**
  - https://apps.dev.microsoft.com/
  - https://developer.microsoft.com/en-us/graph/docs/concepts/auth_register_app_v2
- **Authenticate using OAuth2**
  - https://developer.microsoft.com/en-us/graph/docs/concepts/auth_v2_user
- **Choose appropriate permissions**
  - **Don’t just ask for everything!**
  - **Delegated permissions**
  - **Application permissions**
  - **https://developer.microsoft.com/en-us/graph/docs/concepts/permissions_reference**

OAuth2 – Flow Options





Adaptive Cards

- JSON formatted
- Portable schema for describing UI content
- Rendered natively by host applications
- Usable for bots, email clients, chat applications, Windows notifications, Cortana skills, and Windows Timeline
- Today we’re using it to display our activity feed item inside Windows Timeline

Adaptive Cards - Examples

Publish Adaptive Card schema

Matt Hidinger
Created Mon, Feb 13, 2017

Now that we have defined the main rules and features of the format, we need to produce a schema and publish it to GitHub. The schema will be the starting point of our reference documentation.

Board: Adaptive Card
List: Backlog
Assigned to: Matt Hidinger
Due date: Not set

Set due date | Comment

Your registration is almost complete

What type of food do you prefer?



Steak | Chicken | Tofu

SHADES

Dec 4
Final

7 - 40

SKINS

Adaptive Cards - Tooling

- Authoring SDKs
  - .NET
  - JavaScript
- Hosting SDKs
  - UWP
  - WPF
  - HTML
  - iOS
  - Android
  - .NET Image
- Visualizer
  - http://adaptivecards.io/visualizer/

Protocol Handlers

- Register a protocol handler for your application
  - https://blogs.msdn.microsoft.com/noahc/2006/10/19/register-a-custom-url-protocol-handler/
  - Your protocol handler can be a batch file or a separate launcher or dbr.exe with your dbr as an argument

Deep Linking

- CMDLN
  - https://www.synergex.com/docs/#lrm/lrmChap9CMDLN.htm
  - Maximum processable command line by CMDLN is 2k in length

- Take arguments from the command line that activated your app

- Navigate within your app to the requested context

REST Calls

- PUT /me/activities/{appActivityId}
- GET /me/activities
- GET /me/activities/recent?$expand=historyItems($filter=lastModifiedDateTime gt 2018-01-22T21:45:00.347Z and lastModifiedDateTime lt 2018-01-22T22:00:00.347Z)
- DELETE /me/activities/{id}

PUT Example

```
PUT https://graph.microsoft.com/v1.0/me/activities/%2FTracker%3F38033
{
   "appActivityId": "/Tracker?38033",
   "activitySourceHost": "https://www.contoso.com",
   "userTimezone": "America/Los_Angeles",
   "appDisplayName": “Tracker",
   "activationUrl": “Tracker:38033",
   "fallbackUrl": "http://synergex.com/tracker?id=38033",
   "contentInfo": {
      "@context": "http://schema.org",
      "@type": “WriteAction",
      "author": “LCB",
      "name": “’Equals’ keyword is miss-classified as an in-proc keyword"
   },
   "visualElements": { Adaptive card content goes here }
}
```

Demo

Tracker is a traditional Synergy Toolkit application used internally at Synergex.

Device Relay

- Cross platform (iOS, Android, Windows)
- Explicit mechanism for sending an activity to a different device
- Currently in preview
- Enables very sci-fi user experiences

OneDrive

/me/drive	User's OneDrive
/me/drives	Enumerate OneDrive resources available to the user.
/drives/{drive-id}	Access a specific **Drive** by the drive's ID.
/drives/{drive-id}/root/children	Enumerate the **DriveItem** resources in the root of a specific **Drive**.
/me/drive/items/{item-id}	Access a **DriveItem** in the user's OneDrive by its unique ID.
/me/drive/special/{special-id}	Access a special (named) folder in the user's OneDrive by its known name.
/users/{user-id}/drive	Access another user's OneDrive by using the user's unique ID.
/groups/{group-id}/drive	Access the default document library for a group by the group's unique ID.
/shares/{share-id}	Access a **DriveItem** by its **sharedId** or sharing URL.

Contacts

- https://graph.microsoft.com/v1.0/me/contacts
- Filterable
  - https://graph.microsoft.com/v1.0/me/contacts?$filter=emailAddresses/any(a:a/address eq '@domain.com')

Example response


```
{
  "value": [
    {
      "parentFolderId": "parentFolderId-value",
      "birthday": "datetime-value",
      "fileAs": "fileAs-value",
      "displayName": "displayName-value",
      "givenName": "givenName-value",
      "initials": "initials-value"
    }
  ]
}
```

OData

- Microsoft Graph is powered by OData
- Graph Explorer
  - https://developer.microsoft.com/en-us/graph/graph-explorer
- OData queries are supported
  - $top, $skip, $filter, $select, $orderby, $expand
  - Some examples use the key as segment convention, but you can also use the normal parenthesized function argument convention

Where Can You Use This?

- REST API – usable everywhere
  - Synergy HTTP document transport API works great here
- .NET SDKs – usable anywhere .NET Core can be run

Additional Links

- https://developer.microsoft.com/en-us/graph/examples
- https://stackoverflow.com/questions/tagged/microsoft-graph
- https://channel9.msdn.com/Events/Build/2018/THR2437
- http://adaptivecards.io/
- https://youtu.be/nNdirfTlU4E