Reach for the Cloud: Build Solutions with the Power of Microsoft Graph

Bill Ayers

#SPDoctor
A Huge “Thank You!” To Our Sponsors …
Eur Ing Dr Bill Ayers
MCM/MCSM Charter SharePoint

MVP, MCT, MCTS, MCITP, MCSD, MCAD, MCSA, MCDBA, Professional Scrum Master, etc. etc.

Flow Simulation Ltd.
www.SPDoctor.net
BillA@flosim.com
@SPDoctor

Consultant specialising in SharePoint and Office 365 Development and Architecture for Collaboration and Mobile Development
Agenda

- What is a Graph?
- The Microsoft Graph
- Raw REST API
- MSGraphClient
- Conclusions
What is a Graph?
Microsoft Graph

a unified REST API
and comprehensive developer experience
for integrating the data and intelligence
exposed by Microsoft services.
Office 365 Platform

Extensions

Office Canvases
- Documents
- Conversations
- Pages
- Embedded canvases

Standalone web, device, and service apps

Microsoft Graph
Microsoft Graph

Access user, group and organizational data

One endpoint
One token
All users

https://graph.microsoft.com

https://graph.microsoft.com

Users | Groups | SharePoint | Outlook | OneNote | Planner | Teams | Excel | Intune | Azure AD | more...
Drawbacks of Office 365 Service APIs

- Multiple inconsistent APIs and endpoints to retrieve data
- Need to obtain multiple access tokens
- REST and OData standards have continued to evolve
- Different authorization mechanisms for work and personal
Microsoft Graph API vs. Service APIs

- Microsoft Graph:
  - Simplest way to access data in Office 365 and other Microsoft cloud services

- Service specific endpoints:
  - Available and supported
  - Require resource specific access tokens

https://graph.microsoft.com
The core data that drives business is accessible through the Microsoft Graph – expect more...

Rich Context
- Is this person out of the office?
- Who is their manager?
- Where do they need to be next?
- What documents have they been working on recently?

Real-time Updates
- Reschedule meeting when a conflict appears
- Notify owner when a file is modified
- Continue a process immediately after approval mail is received

Deep Insights
- What documents are most interesting to this person?
- What’s the best time to meet for this group of people?
- Who should this person contact for info on this topic?
Calling the API

https://graph.microsoft.com

/{version}/{resource}/{id}/{property}?{query-parameters}

- HTTP verbs dictate the request intent: GET | POST | PATCH | PUT | DELETE
- Version: /v1.0 or /beta
- Resource: /users, /groups, /sites, /drives, /devices, more...
- Member from collection: /users/bill
- Property: /users/bill/department
- Traverse to related resources via navigations: /users/bill/events
- Query parameters: /users/bill/events?$top=5
  - Format results: $select | $orderby
  - Control results: $filter | $expand
  - Paging: $top | $skip | $skiptoken

OData
Explore REST Endpoint

https://dev.office.com/ (follow link to Microsoft Graph then link to Graph Explorer)
Using REST with (e.g.) JavaScript

Preferably:

```javascript
jQuery.ajax({
    url: encodeURI("https://graph.microsoft.com/v1.0/me/")
    headers: { accept: "application/json;odata=verbose" }
}).done(function (data) {
    $('#output').text("Site title: " + data.d.Title);
}).fail(function (msg) {
    $('#output').text("Request Failed: " + msg);
});
```

Whereas:

```javascript
var xhr = new XMLHttpRequest();
xhr.open('GET', "https://graph.microsoft.com/v1.0/me/");
xhr.setRequestHeader("Accept", "application/json;odata=verbose");
xhr.onload = function () {
    var data = JSON.parse(xhr.responseText);
    document.getElementById("output").innerText = "Site title: " + data.d.Title
}
xhr.send();
```

Note: an access token is usually required in the authorization header – outside the scope of this talk.
Wrapper SDKs

• Language-specific bindings
• Easier to use
• Varying degrees of ‘strong typing’
Code-genned Wrapper APIs

$metadata/OpenAPI

Code generation

Manual Fine-tuning

API

https://github.com/microsoftgraph/msgraph-sdk-javascript
Demo: Traversing the Microsoft Graph using Client Libraries

https://github.com/microsoftgraph/msgraph-sdk-javascript
MSGraphClient
Which to use?

- Legacy APIs: CSOM, EWS, etc.
- REST APIs: Outlook, SharePoint, etc.
- Language-specific wrappers, e.g. PnP-JS-Core
- Microsoft Graph REST API
- Microsoft Graph language-specific APIs
- Context-based APIs (MSGraphClient)
Conclusions:

• Microsoft Graph – single endpoint and open standards allow secure interoperability
• Microsoft Graph improving with more content
• The best is yet to come!
• Changing fast – need to be Agile
Thank You!