2025 DYNAMICS CON

dynamicscon.com

Dataverse: Build a Virtual Tables Custom Provider

Jim Novak
Principal Technical Product Manager
Microsoft, Health and Life Sciences
LinkedIn: jamesnovak
Mastodon: @jamesnovak@hachyderm.io

Session Agenda

- Virtual Tables Overview
- Virtual Tables Custom Providers
- Considerations for implementation
- Demo/Code Review
- Example – Virtual Health Data Tables

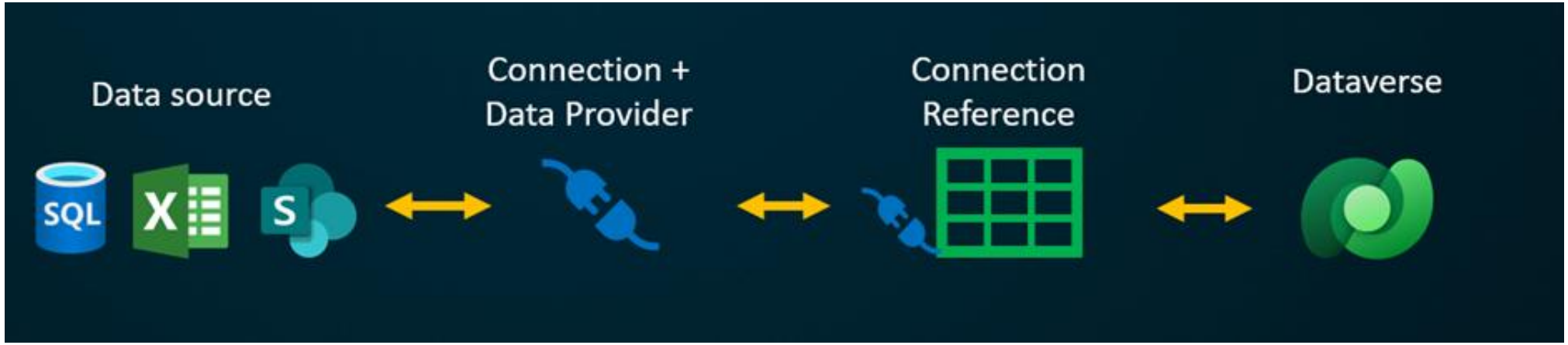
Virtual Data Tables Overview

Key Concepts

- Dataverse Tables bound to data residing in external systems
- No data replication and real time connectivity to external data
- Support for various existing external providers
  - OData v4
  - Azure Cosmos DB
  - Virtual Connection - SQL Server, Microsoft SharePoint
  - Microsoft Fabric
- Not quite full parity with standard Dataverse table capabilities
- Support for creating custom data providers

Virtual Data Tables Overview

Primary Elements of Virtual Tables



- User experience that remains largely the same – working with data from within a Dataverse Model Driven Apps

Virtual Tables Custom Providers

Why build your own Provider?

- OOTB / Configuration only providers do not meet your needs
  - non GUID identifiers
  - Custom authentication
- Full control over external systems data
- Aggregating data – Dataverse and remote data all in one app
- Supported operations
  - Retrieve
  - Retrieve Multiple
  - Create, Update, Delete
- Event Support available

Components and Flow

Common elements implementing a Custom Provider

DataSource (required)

- Contains service connectivity details – Bound to Virtual Table definition

Plugins - Retrieve/RetrieveMultiple, Create, Update, Delete (required)

- Code behind and registered Execution pipeline events for the provider

DataServiceFactory / DataService

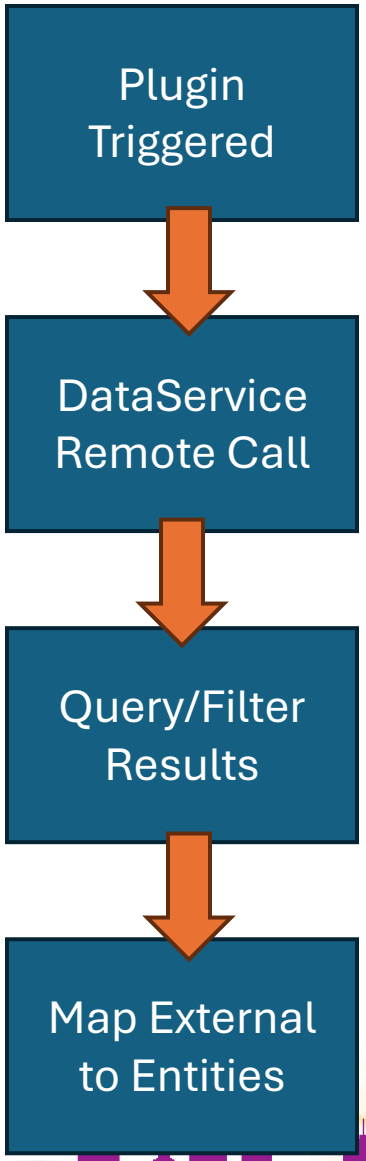
- Pattern you implement that handles interactions with remote service

QueryMapFactory / QueryMap

- SDK Helper class to Automatically map between custom class and Dataverse tables

ExecutionContextFactory / ExecutionContext

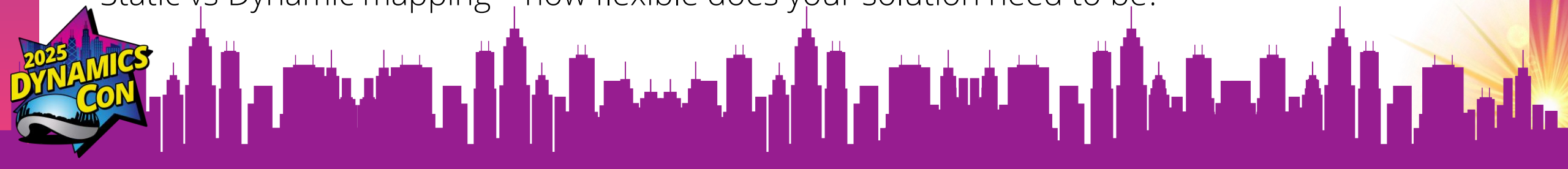
- Pattern you can implement wrapping relevant context for plugin execution



Virtual Tables Custom Providers

Considerations and challenges

- Are any of the limitations deal breakers for your solution?
- What does the remote source support?
- Dataverse plugins - follow many of the same rules and use similar objects
- Transforming queries can be complicated – you decide on what is supported
- Debugging can be a challenge – no profiler support. Unit tests to the rescue!
- Performance – be sure your remote source is not a bottleneck
- Authentication – where can you store your credentials?
- Security – Roles apply to Tables, but you own accessing the remote source
- Static vs Dynamic mapping – how flexible does your solution need to be?



Demo

Check out the code behind and see Virtual Table Providers in action

Review

Session Takeaways

- Virtual Tables can lower storage costs and provide interoperability between systems
- Consider Virtual Table limitations as part of the design
- Out of the box Providers are powerful
- … but you may build your own if needed
- Make sure your remote data source supports your solution requirements
- Consider cost of ownership for a custom provider

Session Resources

- Get started with virtual tables (entities)
- Custom virtual table data providers
- Learn: Manage tables in Dataverse
- Overview of virtual health data tables

Thank you!

Please visit our list of event sponsors!
https://live.dynamicscon.com/sponsors/

Jim Novak
Microsoft, Health and Life Sciences
LinkedIn: jamesnovak
Mastodon: @jamesnovak@hachyderm.io