UI5con

On the road to general adoption of OData V4

Recursive Hierarchy support, Data aggregation, ...

Dr. Thomas Chadzelek, SAP SE
Dr. Mathias Uhlmann, SAP SE

Recursive Hierarchies

Data aggregation

Frequent problems

CRUD methods not at the ODataModel

#created promise is not getting rejected

Data sharing

Recursive Hierarchies

Hierarchies

Introduction

In many use cases, data is hierarchical and needs to be displayed and edited as such. Well known examples are cost center or business partner hierarchies.

The provision of an OData V4 hierarchy support was a key missing point for general S/4HANA adoption of OData V4.

The two solutions for OData V2 are insufficient in different aspects.

- Performance (even in the improved solution)
- Intransparent, not-spec compliant extension of the OData V2 protocol
- Complicated setup in the ABAP backend

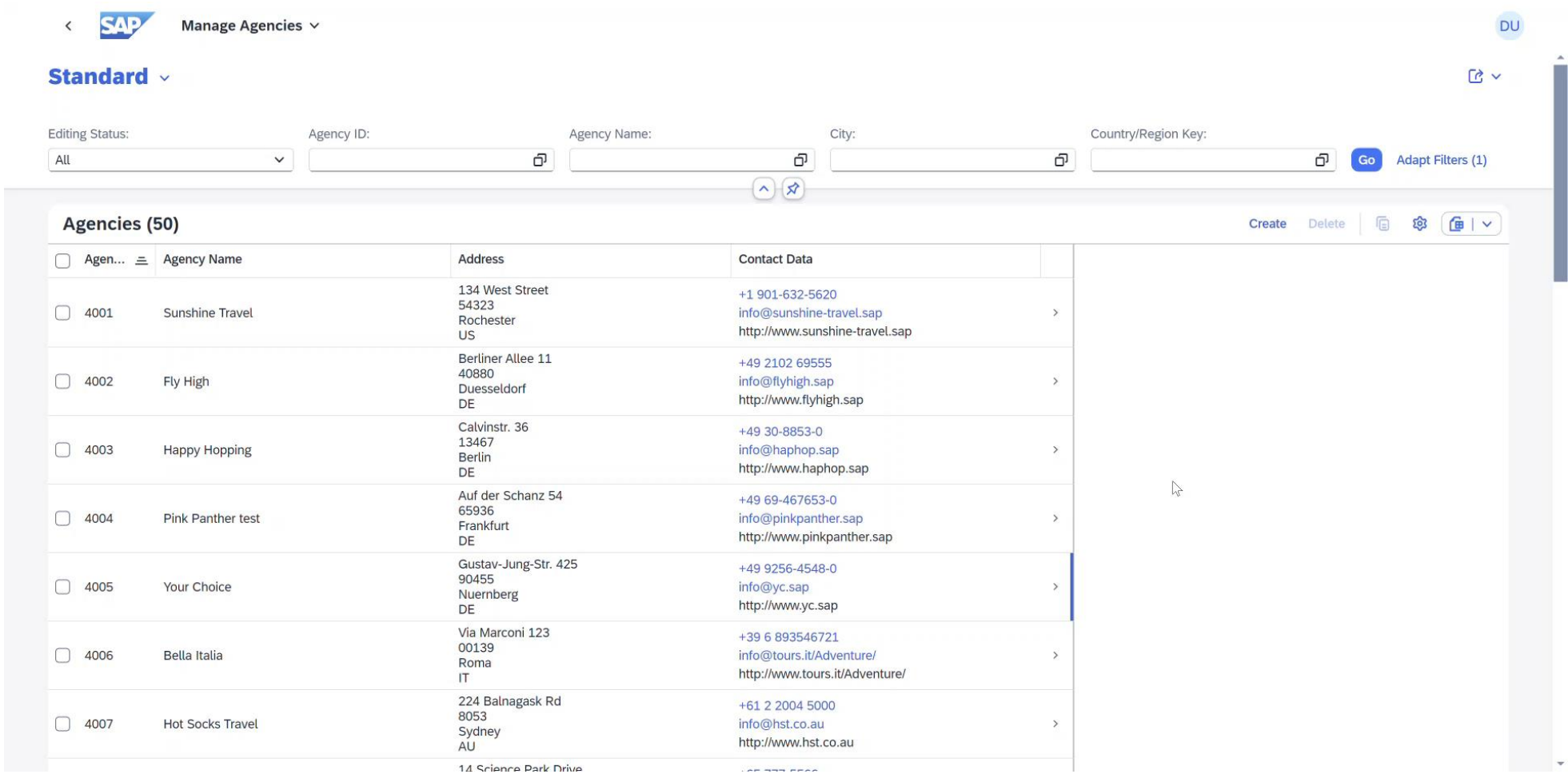
OData V4: Use the previous experience to provide a better solution

UI5con

Hierarchies

Demo

Lab preview



Hierarchies

Availability

End to end solution with RAP, SAPUI5 and Fiori elements using the OData Extension for Data Aggregation Version 4.0.

At the moment, at least the Fiori elements Table building block has to be used.

Hierarchy maintenance requires Draft.

Availability:

- Read-only: SAPUI5 1.117, RAP: W-2311
- Hierarchy maintenance: planned SAPUI5 1.125, planned RAP W-2502 (and may still change)

UI5con

Hierarchies

Documentation

The SAPUI5 documentation Extension for Data Aggregation describes the ODataModel part of the recursive hierarchy features.

Several standard transformations (ancestors, descendants, orderby, filter, search) of $apply are used. Of special importance is the com.sap.vocabularies.Hierarchy.v1.TopLevels transformation.

The ODataModel exposes a set of APIs that are typically used by Fiori elements or the Table. A list binding is used, each node is represented as a context, you can #expand/#collapse, #move around, #getParent etc. Data is enhanced with @$ui5.node.level and @$ui5.node.isExpanded annotations.

Documentation Fiori elements: Tree Tables, FPM explorer Tree Table

Documentation RAP: Implementing Treeviews

Data aggregation

Data aggregation

Introduction

Often aggregated data shall be shown in business software.

Examples are charts showing aggregated data, grand totals showing sums or visual grouping of data showing subtotals.

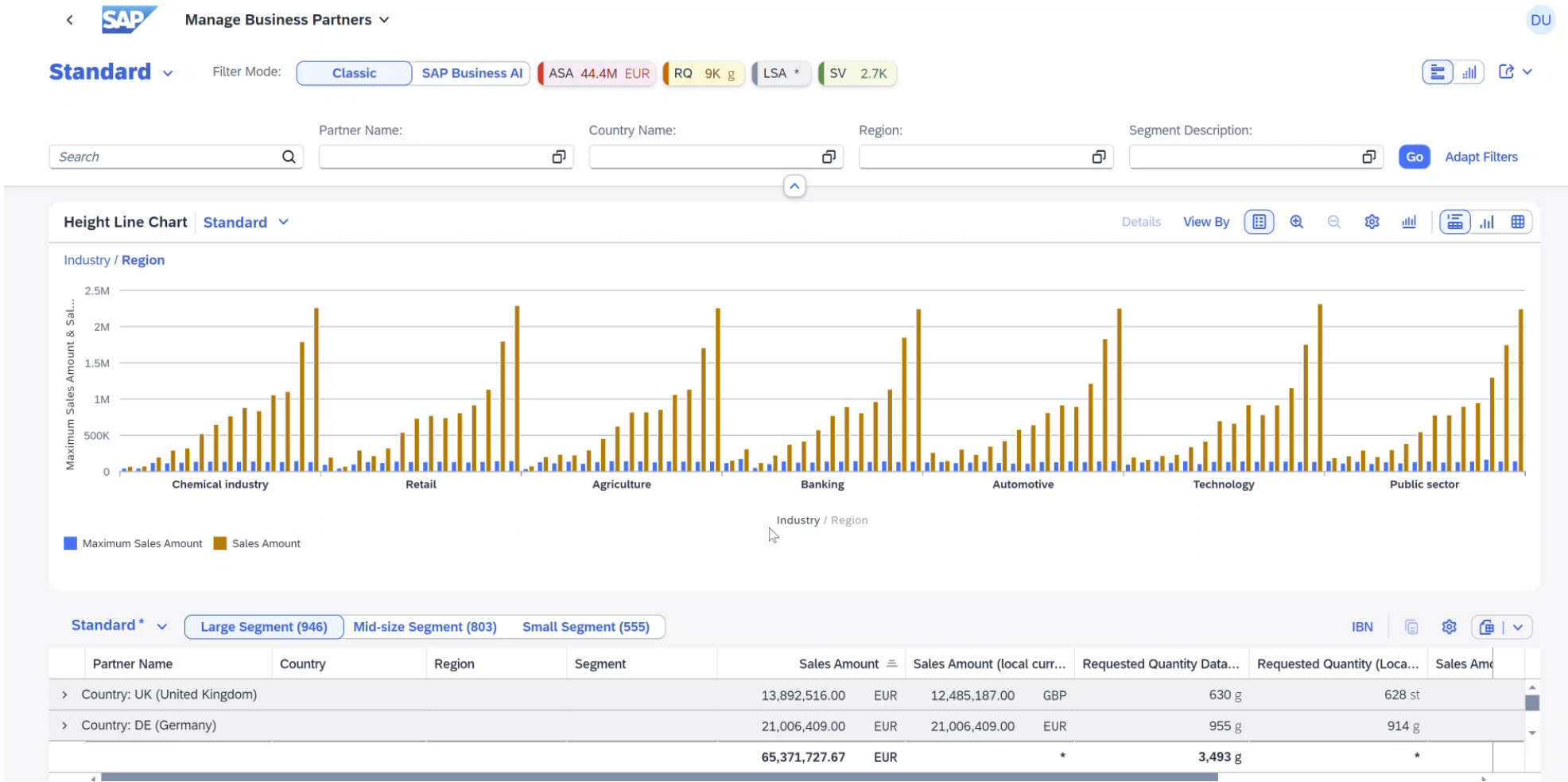
The currently available state was developed some years ago with CAP Java, SAPUI5 and Fiori elements V4. The UI part is available with SAPUI5 1.96 (but there have been improvements afterwards).

UI5con

Data aggregation

Demo

Lab preview



Data aggregation

Documentation

Same documentation Extension for Data Aggregation as before.

Several standard transformations (concat, aggregate, groupby, orderby, filter, search, skip, top) of $apply are used.

The ODataModel exposes a set of APIs that are typically used by Fiori elements or the Table. A list binding is used, each node is represented as a context, you can #expand/#collapse. Data is enhanced with @$ui5.node.level, @$ui5.node.isExpanded, @$ui5.node.isTotal annotations etc.

Documentation Fiori elements: Enabling the ALP in SAP Fiori Elements for OData V4, FPM explorer Analytical Table

Documentation CAP: Data Aggregation

Frequent problems

CRUD methods not at the ODataModel

Problem: With the v2.ODataModel methods like create or read exist at the ODataModel to create, read, update and delete records. The same do not exist at the v4.ODataModel.

Explanation:

- Different architecture of the two ODataModels
- Context API of the v4.ODataModel, that is the context is a much more central object.
- Bindings are more important as they may hold data and cause requests.
- Documentation: Accessing Data in Controller Code
- Blog post by Wouter Lemaire: UI5 OdataModel v4 vs v2 - Custom OData requests

#created promise is not getting rejected

Problem: The promise returned by sap.ui.model.odata.v4.Context#created is not rejected when the POST request fails.

Explanation:

- The failed POST is automatically retried.
- This enables use cases like showing the transient row in the table and allowing the user to edit it.
- The promise can be resolved only once.
- Use the createCompleted event of the v4.ODataListBinding instead to notice when a POST request failed.

Data sharing

Problem: The same entity shall be displayed in a table and a form, for example when using a List Report and an Object Page. The data in table and form shall be in sync.

Explanation:

- The v2.ODataModel uses a global cache for entities so that it does not make a difference how the Object Page / details section / form is bound.
- The v4.ODataModel stores read data in (some of the) bindings.
- Hence, the row context of the table / list binding has to be used as the binding context for the Object Page to ensure that data is in sync.
- Documentation: Data Reuse

UI5con

Thank you!

Like what you just saw?
Please rate our session in the event app:

Dr. Thomas Chadzelek, SAP SE
Dr. Mathias Uhlmann, SAP SE