

News from the OData V4 Model

Latest Features and Innovations





Mathias Uhlmann

Development Manager SAP SE



Lars Kissel

Senior Developer SAP SE

Agenda



01 Latest Binding Capabilities

02 Aggregations

03 Selection

04 Q&A



Latest Binding Capabilities

Expensive Properties, Upsert,

OneWay Property Bindings for Objects



Expensive Navigation Properties

- v4.ODataListBinding: \$\$separate
- Parallel loading of single-valued navigation properties
- Better UX by improved performance

From

GET EntitySet?\$select=ID,Name&\$expand=NotExpensive,**Expensive**

To

GET EntitySet?\$select=ID,Name&\$expand=NotExpensive GET EntitySet?\$expand=**Expensive**



Expensive Navigation Properties

Handle Loaded Data with separateReceived Event

```
oListBinding.attachSeparateReceived(async (oEvent) => {
   const {property, start, length} = oEvent.getParameters();

   const aContexts = await oListBinding.requestContexts(start, length);
   const aResults = aContexts.map((oContext) => oContext.getObject(property));
   // ...
});
```





Expensive Navigation Properties

Handle Errors with separateReceived Event

```
oListBinding.attachSeparateReceived(async (oEvent) => {
   const aErrors = oEvent.getParameter("messagesOnError");

if (aErrors) {
   const aMessages = aErrors.map((oMessage) => oMessage.getMessage());
   MessageBox.error(aMessages.join("\n"));
}
```





Upsert

- Create an entity
- Associated by single-valued navigation property
- Intuitive usage via TwoWay bindings
- Optional APIs
 - <u>v4.Context#setProperty</u>
 - <u>v4.Context#resetChanges</u>



OneWay Property Bindings for Object Values

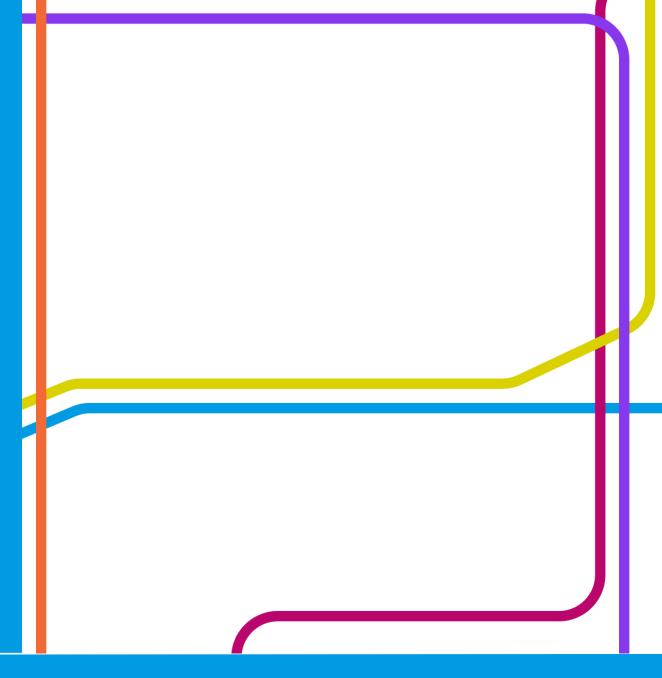
- Support OneWay bindings for complex types
- Bound objects can be used for formatter functions

```
<Text text="{
    path : 'Address',
    mode : 'OneWay',
    formatter : '.formatAddress',
    targetType : 'any'
}" />
```



Aggregations

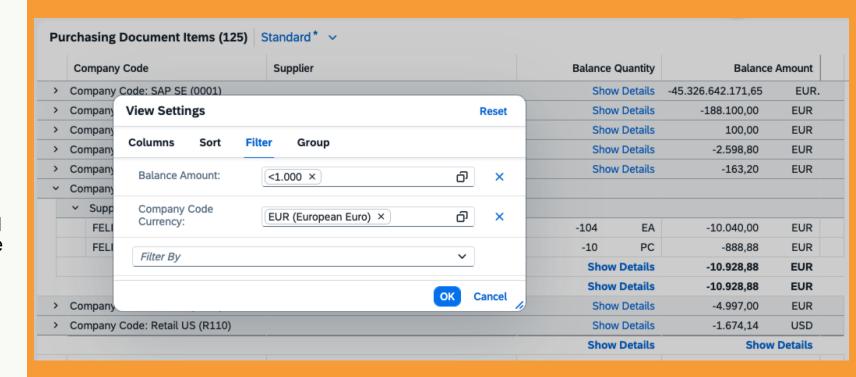
Data Aggregation and Hierarchy Support



Data aggregation

Allow to filter by aggregated properties

• In case the leaf level contains aggregated data, a special \$apply construct needs to be supported by the backend.



```
filter(currency eq 'EUR')/
groupby((companycode), filter($these/aggregate(amount)))
```



Recursive Hierarchy

Additional features provided in the last year

- Expand/Collapse a node completely
 - Model API: v4.Context#expand and v4.Context#collapse
- Copy & Paste of nodes
 - Model API: v4.Context#move
- Create in backend position (instead of showing a newly created node as the first sibling)
 - Model API: v4.ODataListBinding#setAggregation, parameter createInPlace
 - Fiori elements: option createInPlace for the creationMode as documented in Tree Tables

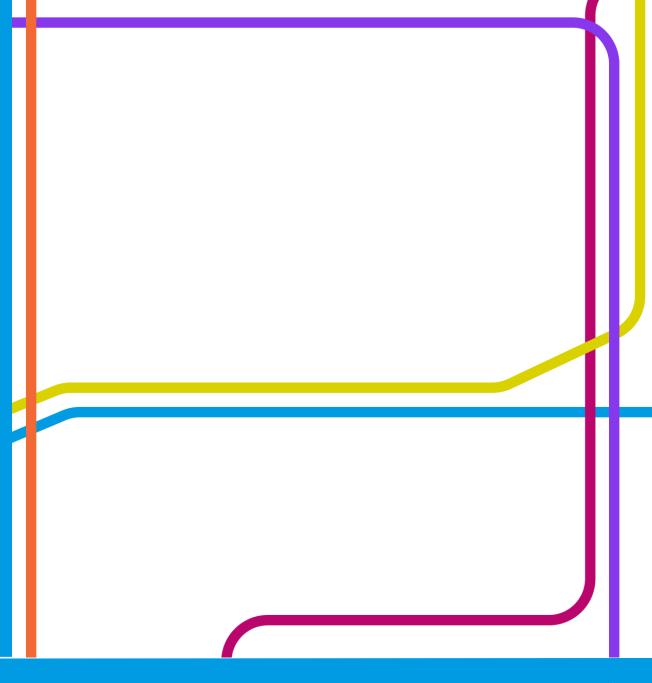
Backend availability

- RAP: Available with the 2502 release
- CAP (Java and Node): available as Beta
 - Note also the session "TreeTables Support in CAP" at reCAP on 2025-07-09 in the Blue room at 16:00





Selection



Selection

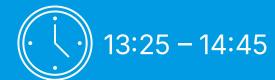
The OData V4 model is now managing selection.

- The selection is already used by the tables.
- Documentation: Selection
- Model APIs
 - v4.Context#setSelected
 - v4.Context#isSelected
- Caution: The data of a selected context may not be up to date or available. Always use requestObject or requestProperty to access data.
- Vision: Allow to efficiently perform mass operations in the backend using filter segments, \$each and collection bound actions provided with OData V4.
 - But it is unclear when we will get there.





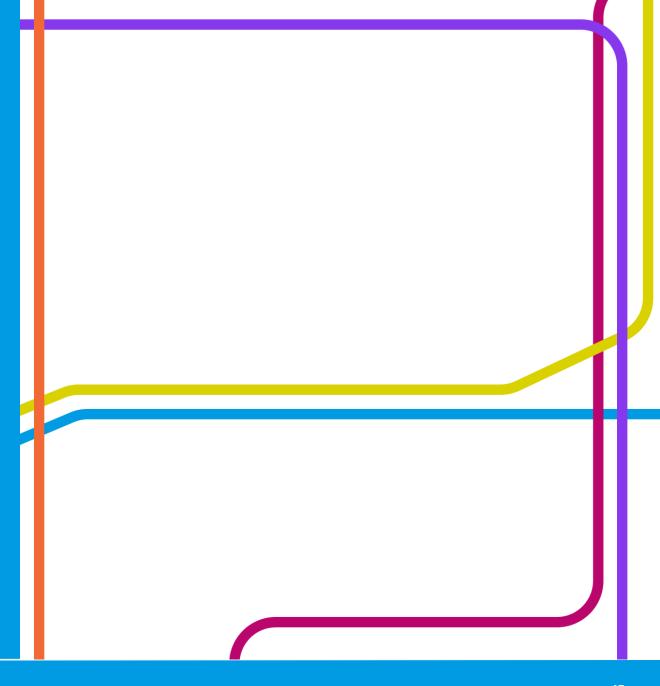
Expert Corner "UI5 Models"







Q&A





Got feedback? Let's hear it!



News from the OData V4 Model

