SAP Cloud Application Programming Model

https://cap.cloud.sap

Roundtable #8

December 12th, 2024

PUBLIC

Today's hosts:

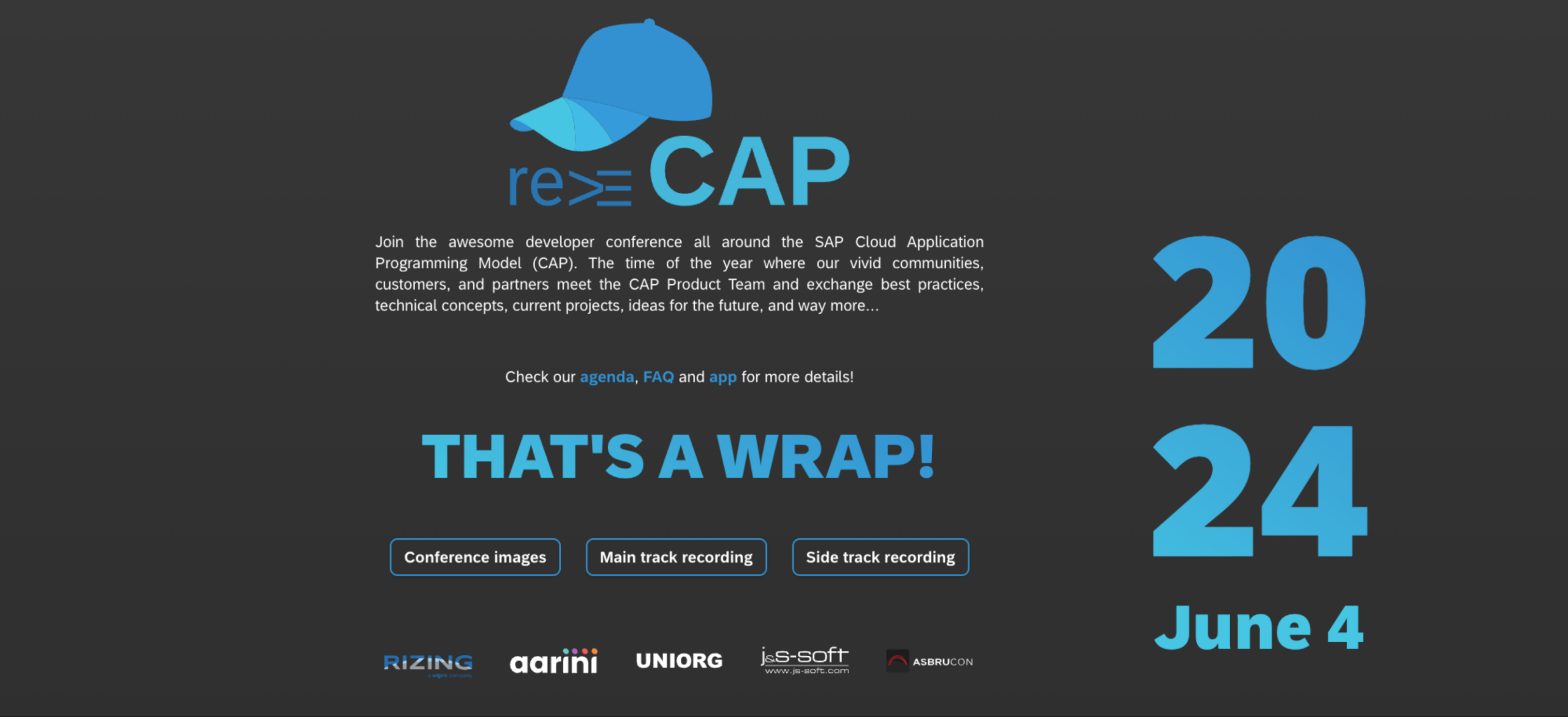
Sebastian Schmidt

Michael Hellenschmidt

THE BEST RUN SAP

Agenda

Intro (Sebastian / Michael)	5 Min
Fiori Tree Table (Evgeny Andreev)	10 Min
RFC (Yuval Morad)	10 Min
Capire Updates (Daniel Hutzel)	5 Min
hana2cds and **cdsgen** for HANA tables and HANA synonyms (Petr Plenkov)	15 Min
Topics raised by Community & Q&A (all)	30 Min



Main track recording: https://broadcast.sap.com/replay/240604_recap
Side track recording: https://www.youtube.com/playlist?list=PLJDFklpD_2cQtnAF22FKky5gmujxDd4AY

New REQs? Check-out the Customer Influence Tool

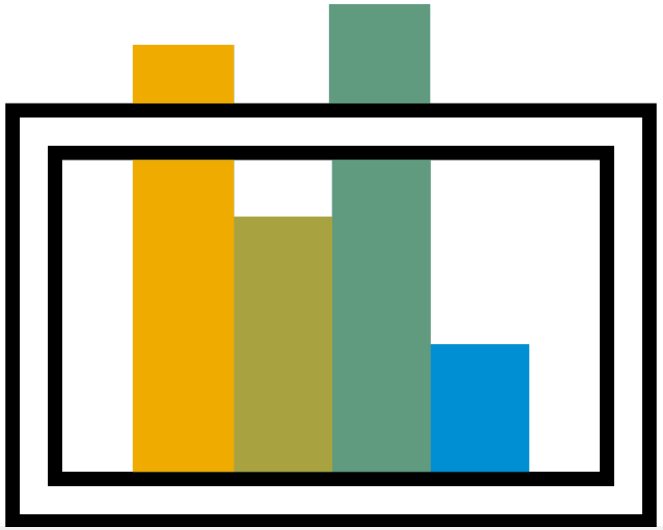
https://influence.sap.com/sap/ino/#/campaign/2280

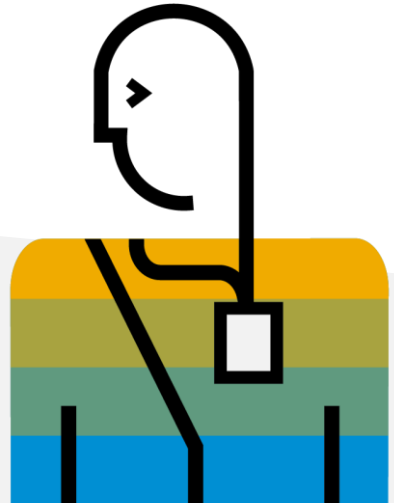
“Event handlers for inbound batch requests” - meeting tomorrow

- Highest voted topic in Customer Influence: https://influence.sap.com/sap/ino/#/idea/310291
- Submit requests to e.g. S/4 HANA in a batch
- “Performing inserts as well as updates in one batch request. Assume a sales order with multiple order positions. In that case we update the head sales order information (like order date, description & sales quote data), delete an order position and insert a new one as replacement. So we would have three CRUD operation within one batch request. The advantage of this approach is that the backend system is able to cumulate the data and validate all information. Furthermore we can lock the whole process data against changes made by others at the same time.”
- Meeting on Friday, 12th of December at 10.00 AM CET.
  → Link to join Teams Meeting ←

Presentation #1

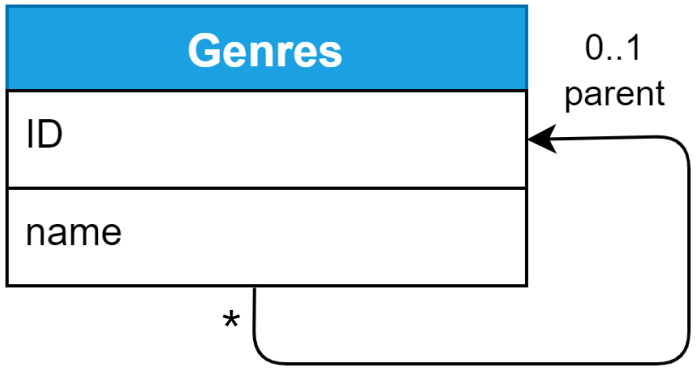
Fiori Tree Table – Evgeny Andreev





Recursive Hierarchies

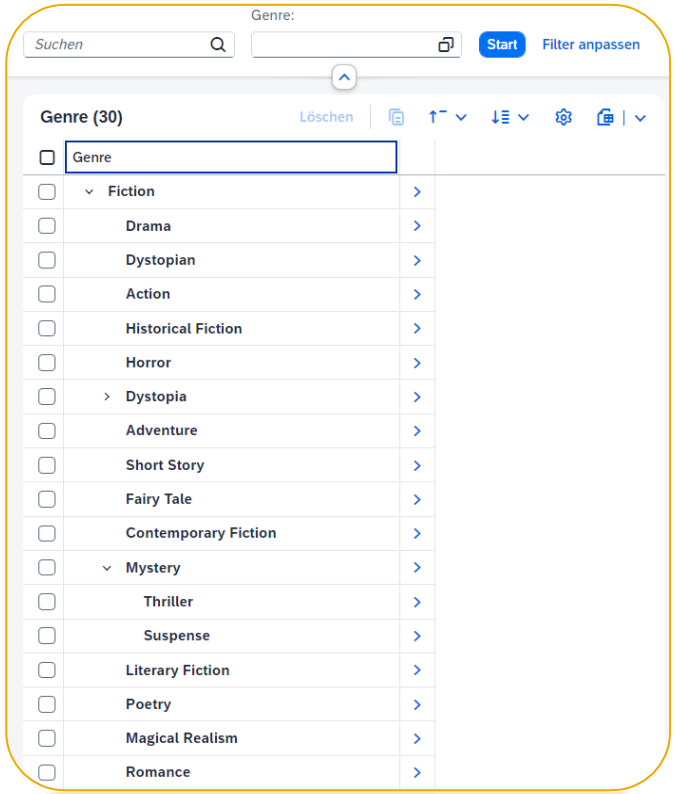
- **Node property** (ID is a good candidate)
- **Parent navigation property** (association to parent node)

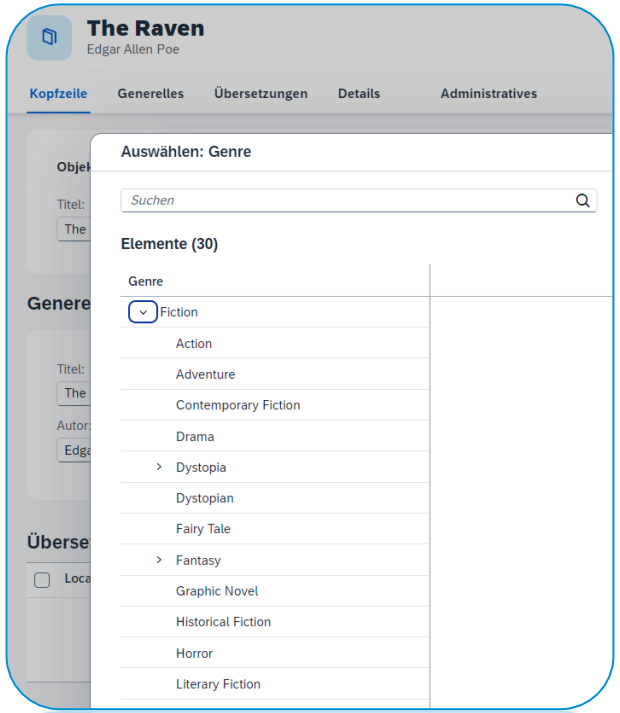


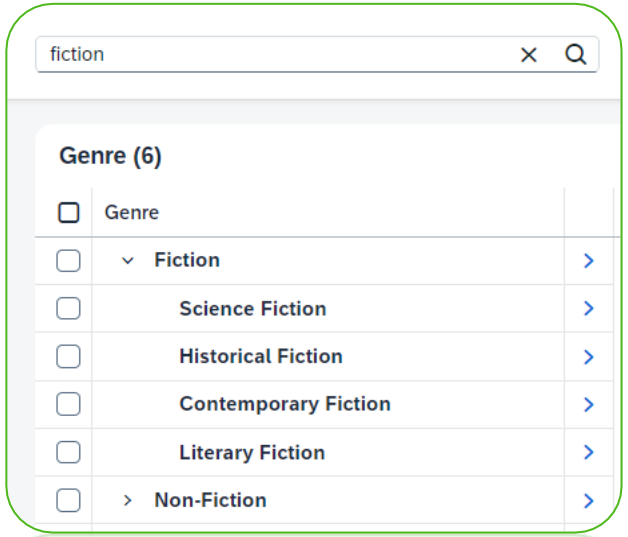
ID	parent	name
fiction		Fiction
action	fiction	Action
adventure	fiction	Adventure
drama	fiction	Drama

Fiori Tree Table

DEMO







Fiori Tree Table: CDS Model

```
entity Genres {
    key ID       : Integer;
        name     : localized String(255);
        descr    : localized String(1000);
        parnt    : Association to Genres;

        children : Composition of many Genres
                       on children.parnt = $self;
}
```

```
@path: 'admin'
@odata.apply.transformations
service AdminService @(requires: 'admin') {
```

```
aspect Hierarchy {
  virtual LimitedDescendantCount : Integer64;
  virtual DistanceFromRoot       : Integer64;
  virtual DrillState             : String;
  virtual Matched                : Boolean;
  virtual MatchedDescendantCount : Integer64;
  virtual LimitedRank            : Integer64;
}
```

```
annotate AdminService.GenreHierarchy with @Aggregation.RecursiveHierarchy #GenreHierarchy: {
  $Type: 'Aggregation.RecursiveHierarchyType',
  NodeProperty: ID, // identifies a node
  ParentNavigationProperty: parnt // navigates to a node's parent
};

annotate AdminService.GenreHierarchy with @Hierarchy.RecursiveHierarchy #GenreHierarchy: {
  $Type: 'Hierarchy.RecursiveHierarchyType',
  LimitedDescendantCount: LimitedDescendantCount,
  DistanceFromRoot: DistanceFromRoot,
  DrillState: DrillState,
  Matched: Matched,
  MatchedDescendantCount: MatchedDescendantCount,
  LimitedRank: LimitedRank
};
```

Fiori Tree Table: summary

Out of the box (CAP JAVA *v.3.6.0, December 2024*)

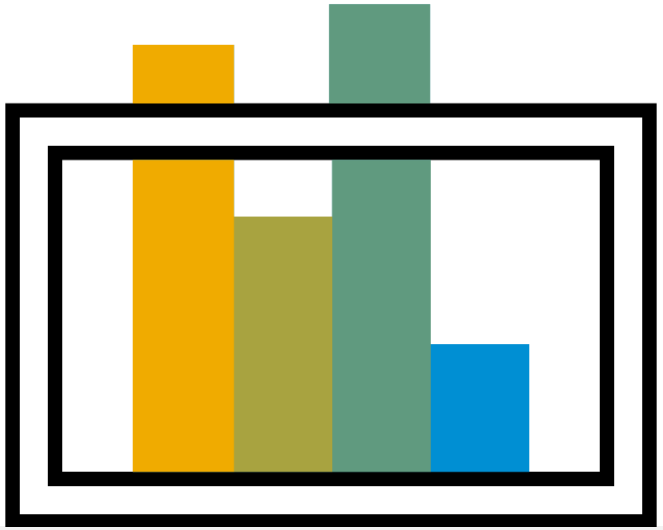
- Read-only scenario
  - *filter, search, sort siblings, etc.*
- For SAP HANA **no** line of code required

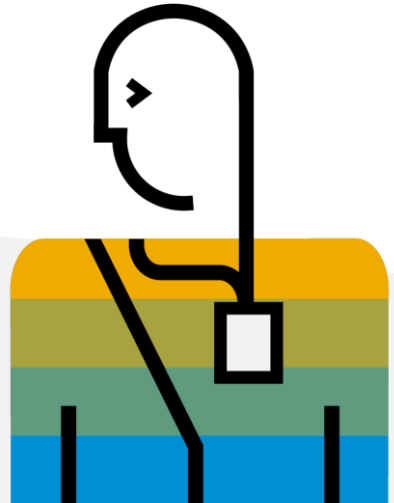
Non-HANA DB **requires** custom code specific to your domain model

Demo: CAP Sample App

Presentation #2

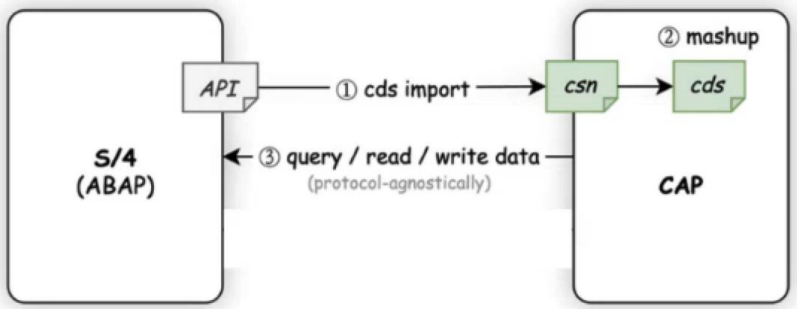
RFC – Yuval Morad / Steffen Weinstock





Remote Function Calls

CAP / Calesi-style Integration Pattern – the 'CAP creed'



(Whatever) API → *cds import* → **CSN**

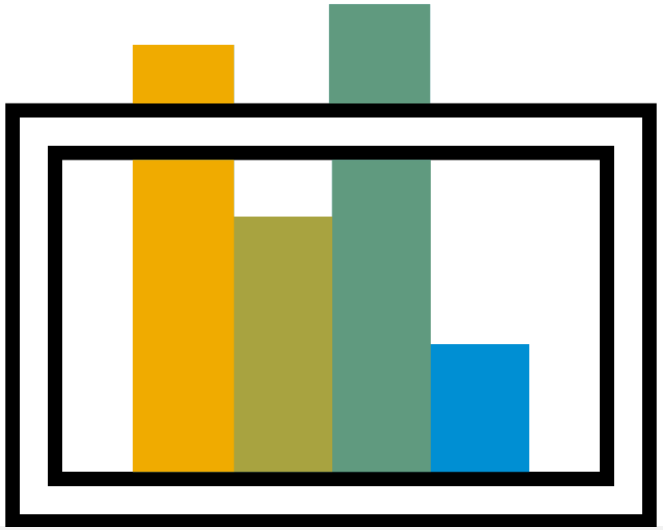
- Everything looks like a CAP service
- We can apply all CDS features → especially projections, extensions, aspects, …

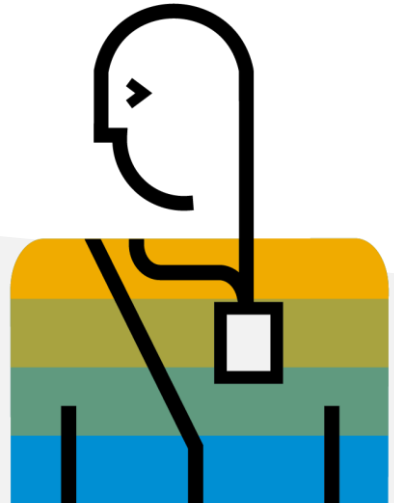
Protocol-agnostic consumption

- Everything feels like a (local) CAP service
- We care about (wire) protocols → they focus on domain
- Late-cut μ services → grow as you go

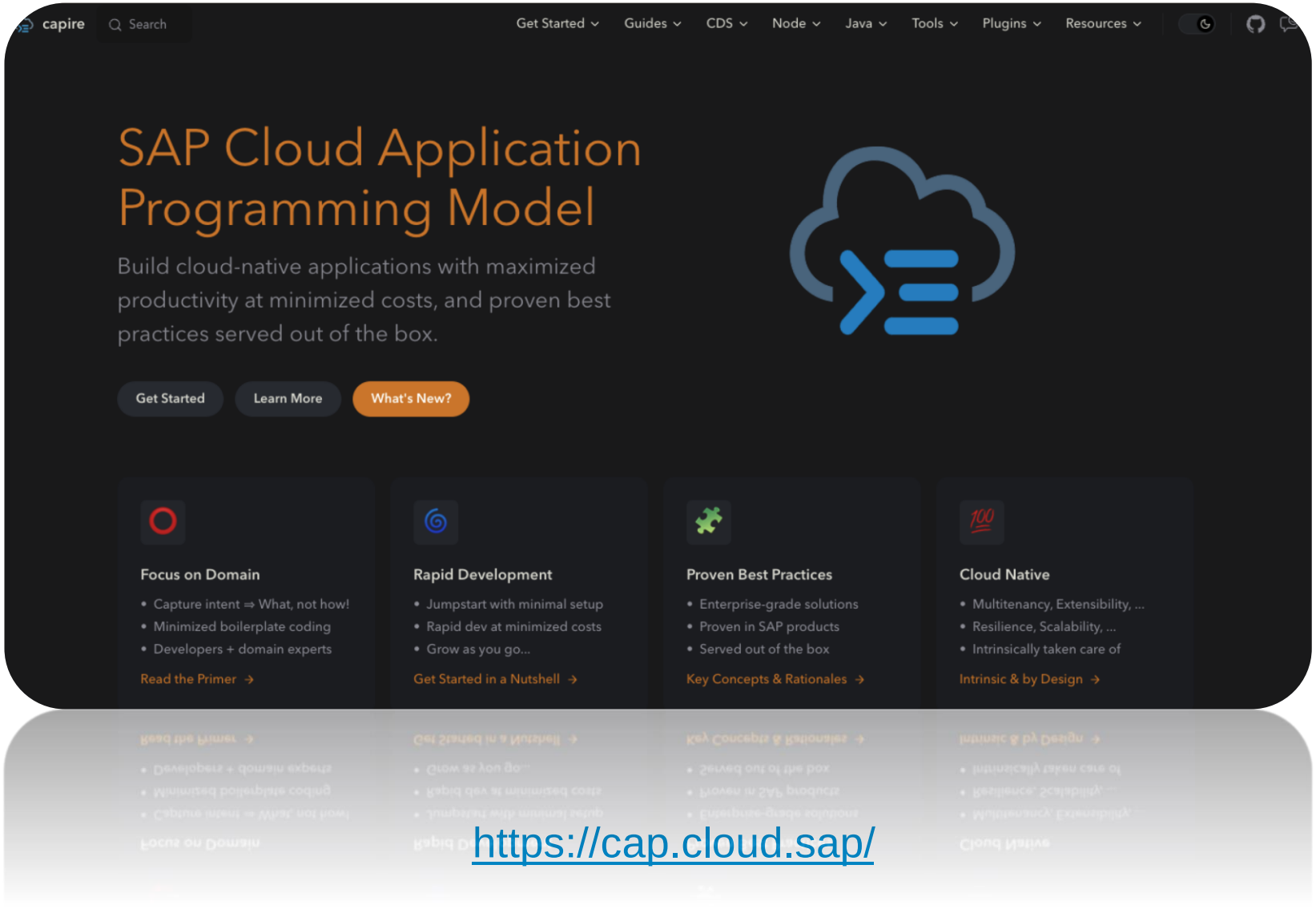
Presentation #3

Capire Update – Daniel Hutzel





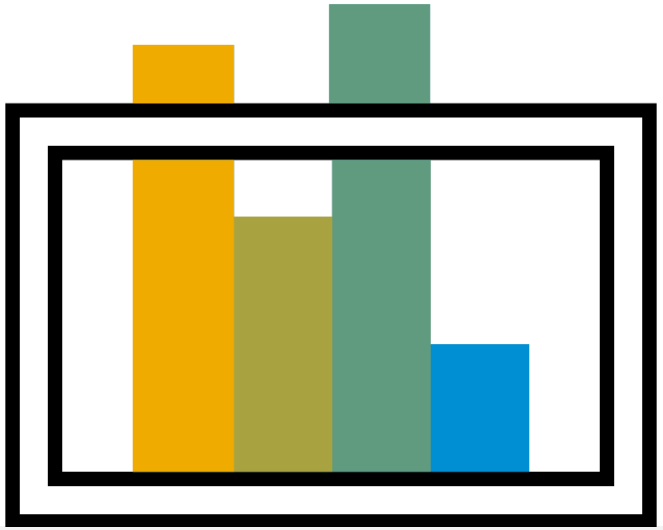
Capire Updates

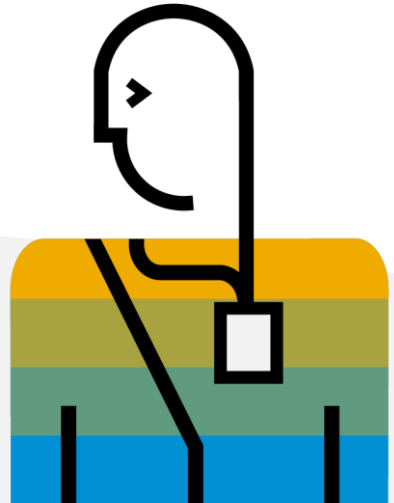


https://cap.cloud.sap/

Presentation #4

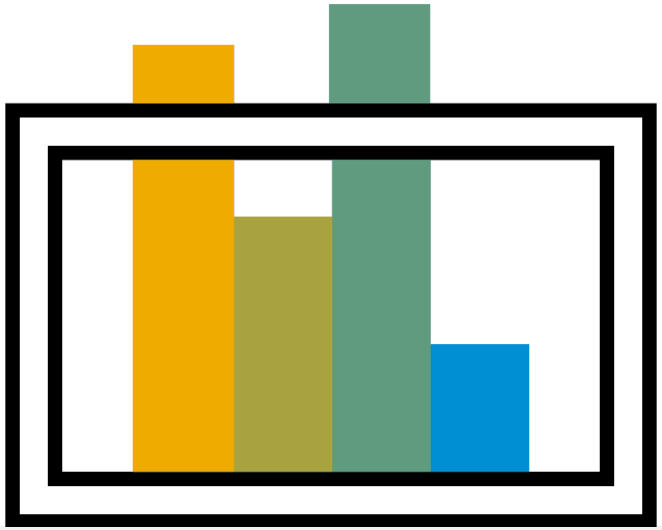
hana2cds and **cdsgen** for HANA tables and HANA synonyms – Petr Plenkov

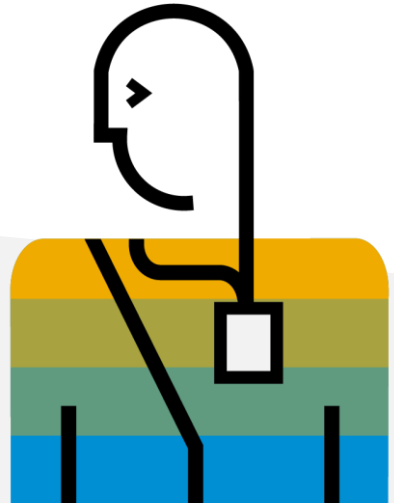




Q&A

Topics raised by Community & Q&A – all





Questions and Topics raised by the community

Q: Timeline / support of **recursive hierarchies in CAP Node.js**?

Q: HANA Hierarchies Support für Node.JS (https://cap.cloud.sap/docs/advanced/odata#hierarchicaltransformations)

Q: Messaging support for **Advanced Event Mesh** / ootb integration with Advanced Event Mesh

Q: Support for **MQTT**

Q: We stumbled across **cds.foreach** again with the comment "Streaming API variant of .run(). Subclasses should override this to support real streaming.",
Ss there something in the pipeline for the standard services?

Q: Feature comparison Java - Node.js?
→ see Feature Matrix: https://cap.cloud.sap/docs/about/features

Q: Current status of **Open Source Strategy**
→ What are your expectations / requirements?

Questions and Topics raised by the community

Q: $compute: Support für Node.JS? (https://cap.cloud.sap/docs/advanced/odata#transformations)

Q: Define CAP Node.JS service endpoints dynamically

Q: Access to @sap-rfc/node-rfc-library repository-Based Shipment Channel –
Case: 1176252/2024
It seems that no external can use @sap/cds-rfc as @sap-rfc/node-rfc-library cannot be loaded.

Q: Even CAP allows us to model and deploy data seamlessly without any HANA experience, sometimes in certain sceanrios (for example cross-HDI access) - it may be quite time consuming to create such artifacts such as synonyms, synonym configs. It would be nice if the compiler would also know how to generate those artifacts from the compiled CDS model. Just thinking of some **annotation like @hdi.synonym('synonym_name')** which would generate the synonym| and ideally synonym config too (to support logical service names instead of hardcoded schemas)

Q: **Plugin API for compilation** https://github.com/orgs/cap-js/discussions/11

Q&A

Thank you.

Contact information:

Sebastian Schmidt
Development Manager
se.schmidt@sap.com

Michael Hellenschmidt
Delivery Head
michael.hellenschmidt@sap.com