



Menschen Methoden Lösungen

ABAP is dead: long live SAP CAP(M)





■ Seit 2008

Gründung der BridgingIT GmbH Anfang 2008 als IT-Beratungsunternehmen

■ Über 520 Mitarbeiter

Kundennähe durch ausgeprägten regionalen Fokus. Standorte: Mannheim, Karlsruhe, Frankfurt, Stuttgart, Köln, München, Zug (CH), Nürnberg und Berlin

■ Erfolg

Unabhängigkeit und Stabilität durch Eigenfinanzierung und durchgängigen wirtschaftlichen Erfolg

■ CraftingIT

Gründung der Tochter CraftingIT GmbH Anfang 2014 als IT-Service Unternehmen in Magdeburg

■ Beratungspartner

Neutral, ohne Produkt- oder Vertriebsbindung

■ Netzwerk

Gut ausgebautes Netzwerk aus Applikations- und Technologiepartnern

■ Preis-/Leistung

Attraktives Preis-/Leistungsverhältnis durch geringen Overhead, komfortable Mitarbeiter-Self-Services und konsequentes Outsourcing bei Support Prozessen

■ Qualität

Fokus auf Qualität und Nachhaltigkeit durch hohe Beteiligung der Mitarbeiter an der Unternehmensentwicklung

Wir übernehmen Verantwortung

aufmerksam – positiv – zeitgemäß

Im Projekt

„in time and budget“
Project Service



Im Portfolio

Full Service Anspruch
Innovationsmanagement

In der Zusammenarbeit

Nachhaltige Kundenbindung
Produkt Lebenszyklus

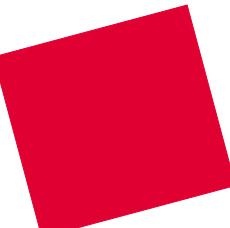


Für die Mitarbeiter

HR Package
Erfolgsbeteiligung

Für die Qualität

Qualifizierung
Coaching und Mentoring

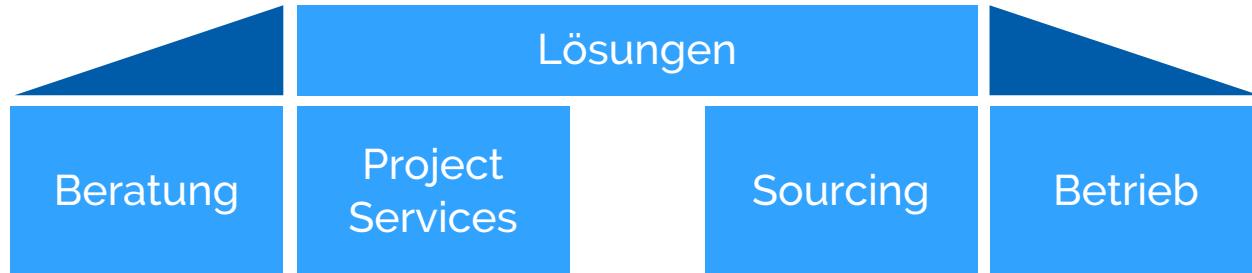


Für die Umwelt

Fair Trade und Umwelt
Soziales Engagement

Portfolio

Unser Full-Service-Ansatz



Beratung

Beratung von der Strategie über die Gestaltung von Geschäftsprozessen bis hin zur Bewertung und Einführung von neuen IT-Systemen und -Anwendungen

Project Services

Unterstützung und Projektsupport bei der Gestaltung von Projekten über den gesamten Lebenszyklus durch Anforderungs-, Projekt- und Testmanagement

Lösungen

Umsetzung von fachlichen und technischen Anforderungen unserer Kunden in bedarfsgerechte, kosteneffiziente und langfristig stabile Lösungen auf der Basis von Standardtechnologien- und Produkten

Sourcing

Unterstützung von der Bereitstellung und Ergänzung von Projektteams über die Beratung in Near-/Offshore-Vorhaben bis hin zum umfassenden Lieferantenmanagement durch CraftingIT GmbH

Betrieb

Dienstleistungen rund um den Betrieb von IT-Systemen und Anwendungen wie Betriebsprozesse, operative Betriebsführung, Beratung zu Fragen der Betriebssicherheit und Wartung

About Me

Tobias Hofmann
Senior Consultant



METHODEN

- Fiori
- Cloud
- Portal
- Mobile
- Architect
- Multicloud

BERUFSFELD

- Twitter: @tobiashofmann
- E-Mail: tobias.hofmann@bridging-it.de

About Me

Tobias Hofmann
Senior Consultant



COMMUNITY

- SAP Inside Tracks Rio, Sao Paulo, Sao Leopoldo
- SAP Meetup Rio de Janeiro
- Topic Lead SCN SAP Portal (3x)
- Speaker: SIT, Meetups, SAP TechEd, SAP Forum, UI5Con, HANA Night
- SAP InnoJam
- Old SCN: Top 100 contributor (15931 Karma Credits)

- Founder: SAP Stammtisch Karlsruhe Baden
- Founder : SAP Stammtisch Bruchsal (merged)
- SAP Stammtisch Stuttgart
- Hackathon (SAP UA, bIT)
- Blogger
- Twitter disturber
- SAP Mentor



#SAP Stammtisch Karlsruhe



Tourdaten

17/12/2018	Weihnachtsmarkt Karlsruhe	Vogelbräu Durlach	30/01/2019
20/02/2019	Litfaß	Kühler Krug	21/03/2019
11/04/2019	Café Galerie	Carls Wirtshaus	09/05/2019
06/06/2019	Erste Fracht	Vogelbräu Karlsruhe	11/07/2019
01/08/2019	Badisch Brauhaus	Multi Kulti	05/09/2019
02/10/2019	Marktlücke	TBA	00/11/2019
00/12/2019	Weihnachtsmarkt Karlsruhe	TBA	00/01/2020

Auswärtstour 2018/2019 Weltherrschaft, Schritt 1: Konsularische Vertretungen

Oktober 2018 München Augustiner, Hofbräuhaus, Schneider Bräuhaus, Hacker-Pschorr

Februar 2019 Berlin BrewDog, Stone Brewing Tap Room, Hofbräuhaus, Mikkeller, Lemke am Alex, Brauhaus Lemke am Hackeschen Markt

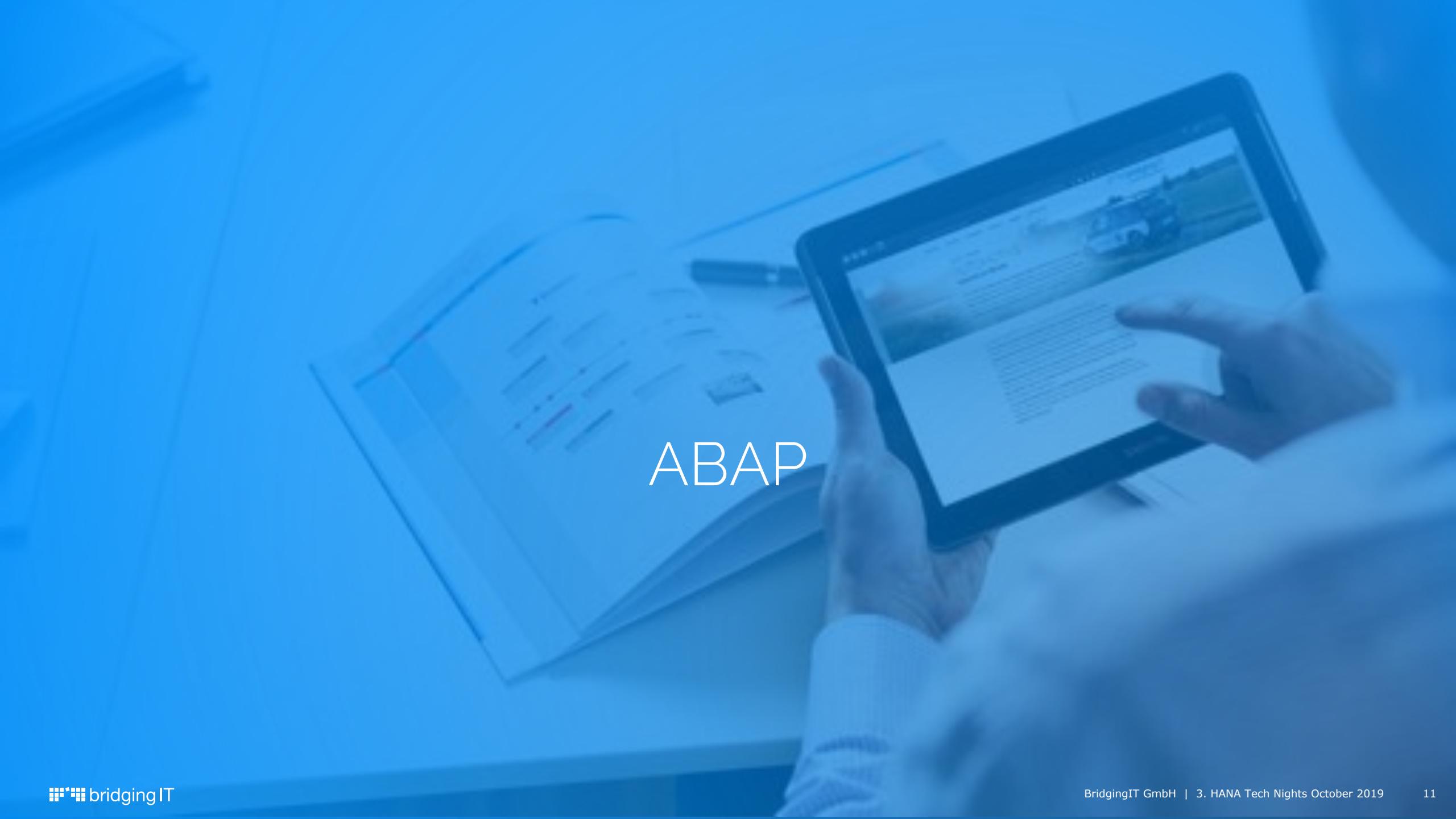
SAP Stammtisch Bruchsal

20/05/2018 - Wallhall

25/05/2018 - Wallhall

Agenda

0 1	ABAP
0 2	CLOUD
0 3	EXAMPLE
0 4	CODE
0 5	LESSONS LEARNED



ABAP

ABAP

Allgemeiner Berichtsaufbereitungsprozessor



Advanced Business Application Programming

ABAP is DEAD

„Niemand hat die Absicht, eine Mauer zu errichten!“

Walter Ulbricht, 15. June 1961

ABAP is DEAD

Die Mauer stand 28 Jahre und 88 Tage

<https://www.bundesregierung.de/breg-de/themen/deutsche-einheit/-niemand-hat-die-absicht-eine-mauer-zu-errichten--393932>

ABAP is DEAD

ABAP:

Started: 1983

End of support

NW ABAP 7.52 ABAP: 31.12.2025

Time:

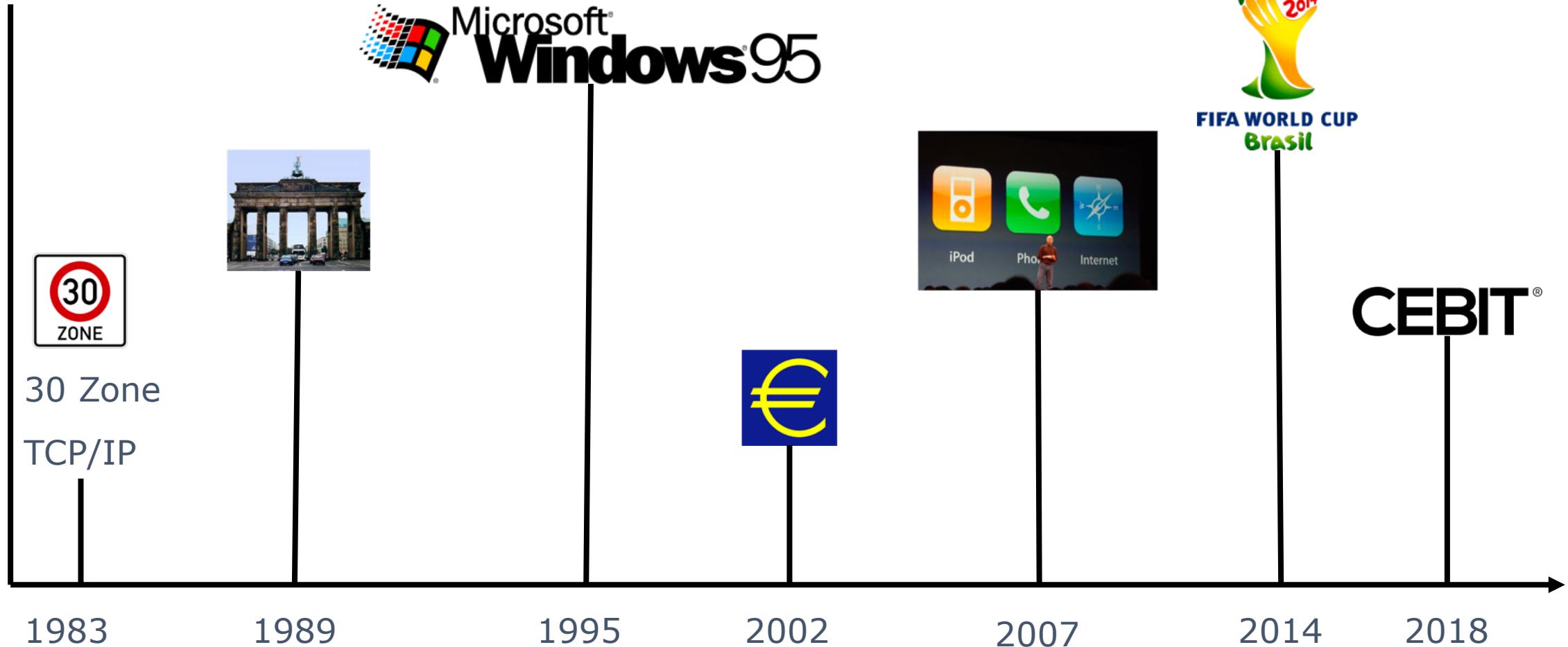
1983 – 2025: 42 Years



ABAP is close to the dream of a german company worker:
Abschlagsfreie Rente
(needs 45 years, still missing 3 more years)

ABAP is DEAD

What happened during the last years

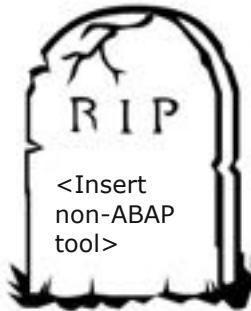


ABAP is DEAD

DEAD*

*Don't Ever Admit Defeat

ABAP killed survived



ABAP is flexible and it works

Why SAP and ABAP is so great: ABAP as a pizza (AaaP)



Standard



Gives: What you need



Add: What you want



<https://de.wikipedia.org/wiki/Pizza>

<https://imgur.com/gallery/oGLJt>

ABAP as a Pizza

Tudo acaba em pizza
- Milton Peruzzi

Brazilian Proverb

Translation: All's well that ends well

Tobias' law: **All ABAP projects will end as Brazilian Pizza**

Keep the core clean

Problem: cleaning up the mess

SAP Standard transaction does not start with Z

SAP offers solutions for a specific business problem, not a programming platform

Process may stay over releases, underlying technology not

Do not depend too much on SAP code

Problems

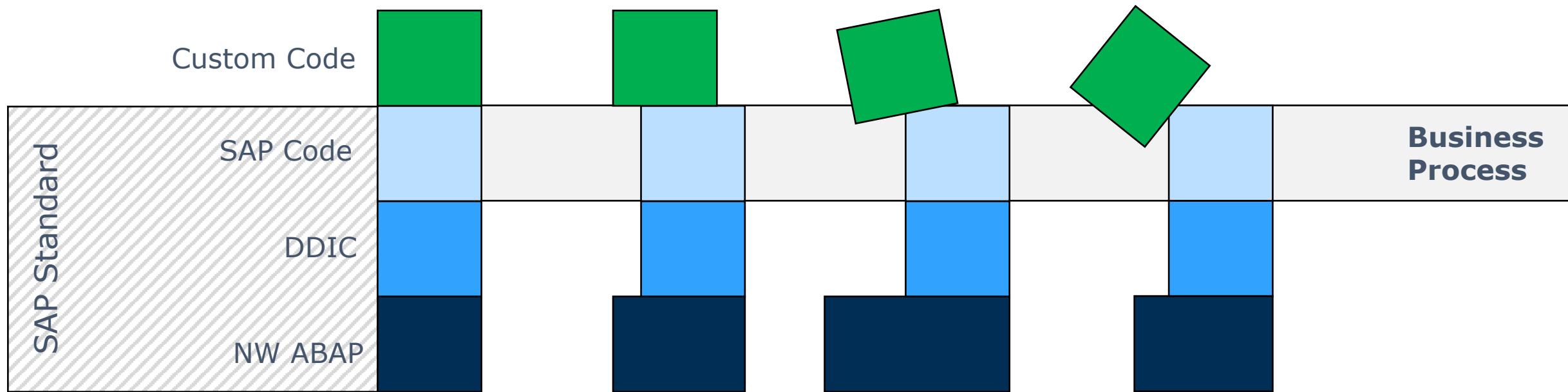
SAP did not do a lot in the last decades to prevent technical debt

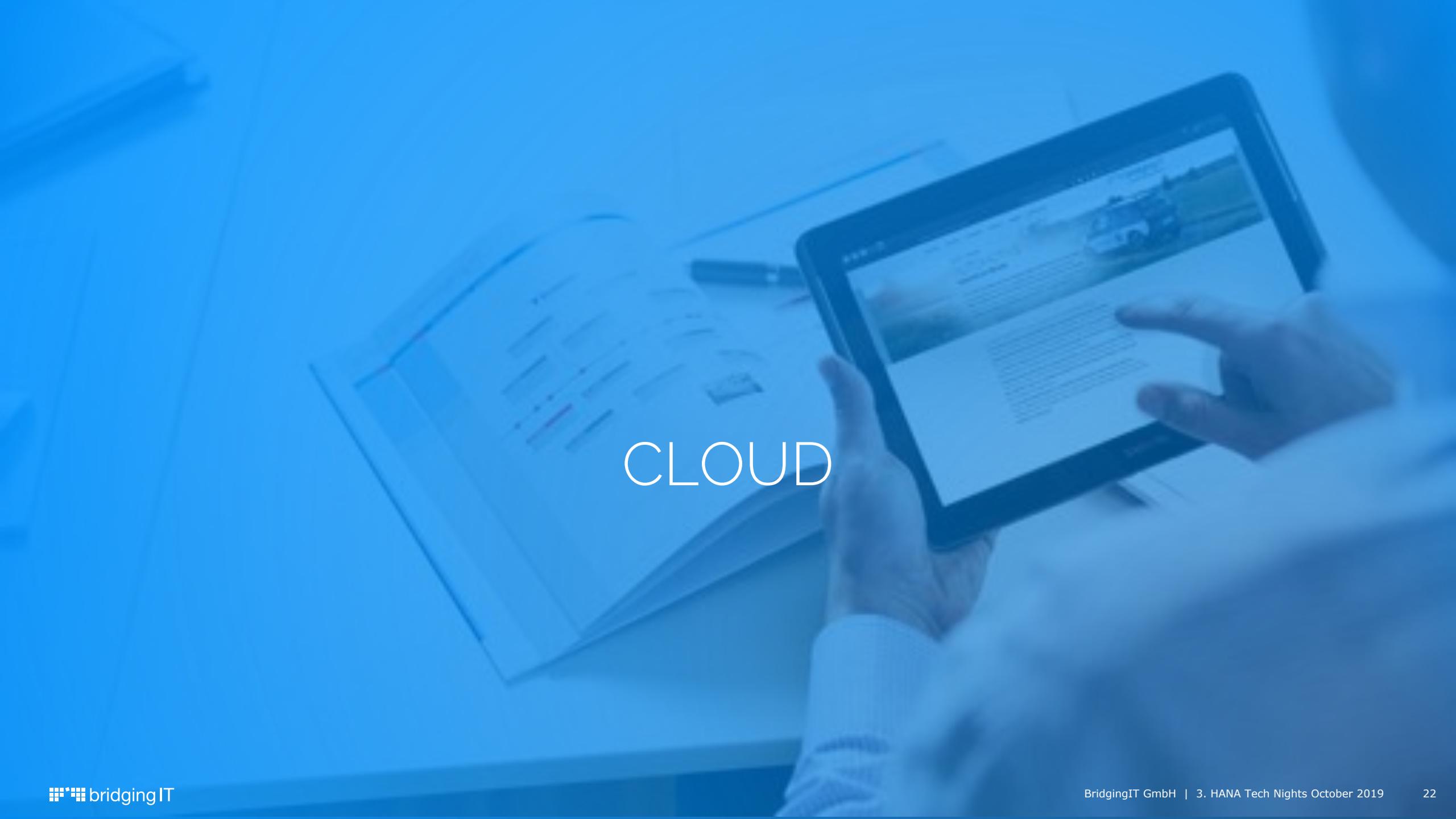
API driven development discovered only since a few years

20+ year old programming techniques are still new

Allocation of resources, long term assignments, projects, SI, customers, governance, ...

SAP Standard





CLOUD

SAP CLOUD

SAP Cloud Application Programming Model

ABAP will stay around for a long time: as core

The future is cloud

- 
- Clean core
 - Cloud SDK
 - ABAP is just an option
 - Many first class citizen

SAP Cloud SDK with SAP Cloud Application Programming Model (SAP CAP)

ABAP@Cloud: Cloud Restful Application Programming (ABAP CRAP) © SAP Stammtisch KA

Idea

*Add what you want
Don't touch the recipe
Stay sane*

Both do the same

- Use CDS to create a service
- Display information with Fiori (Elements)

SAP CLOUD

SAP as Japanese Food



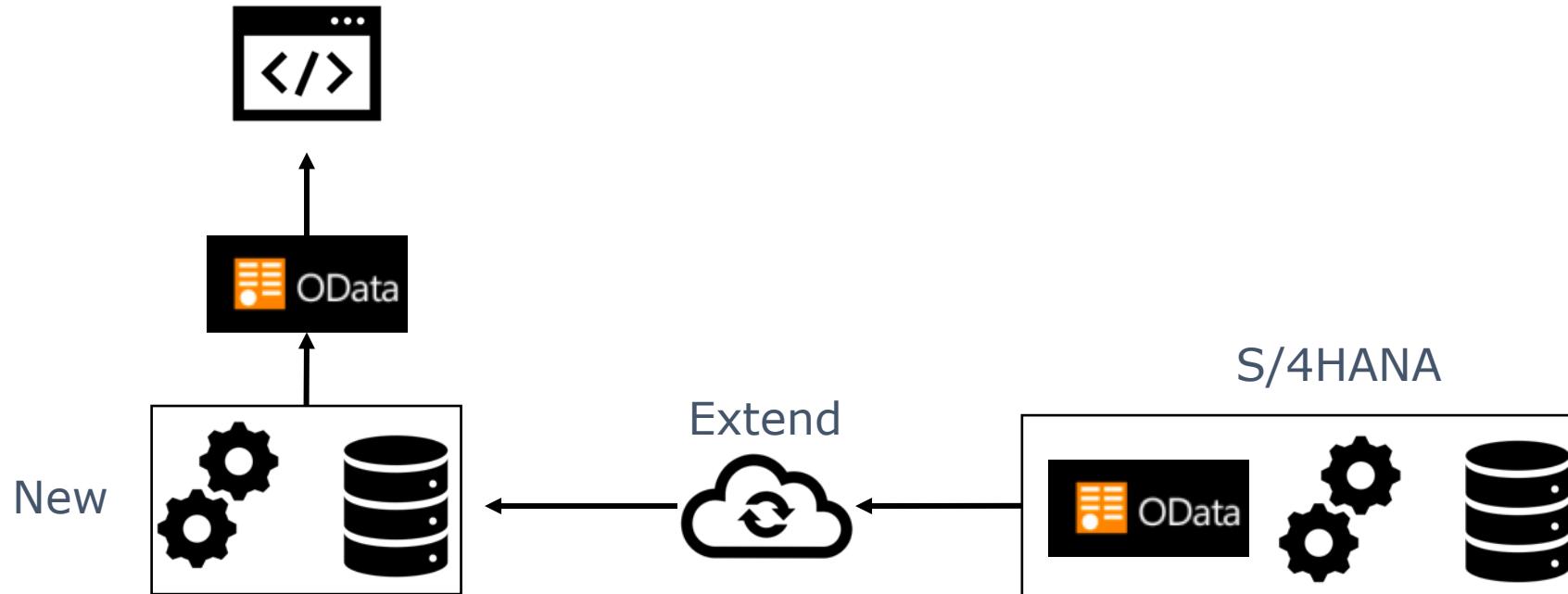
https://en.wikipedia.org/wiki/List_of_Japanese_dishes

SAP Cloud SDK

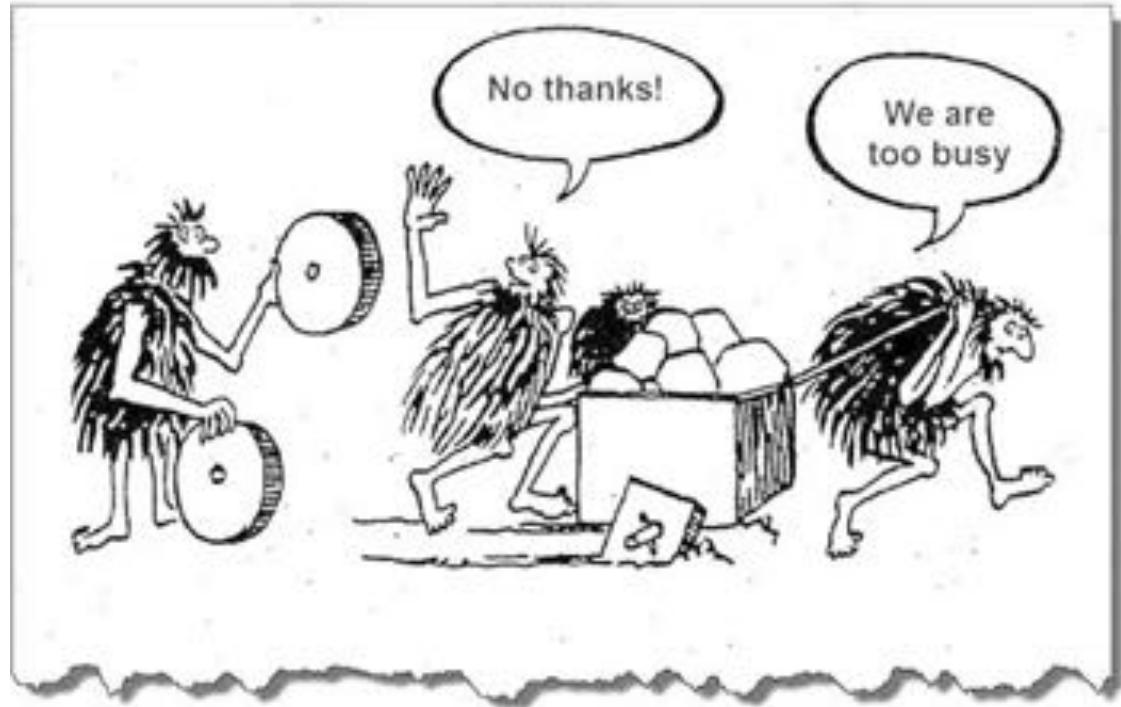
SAP Cloud Application Programming Model

High Level

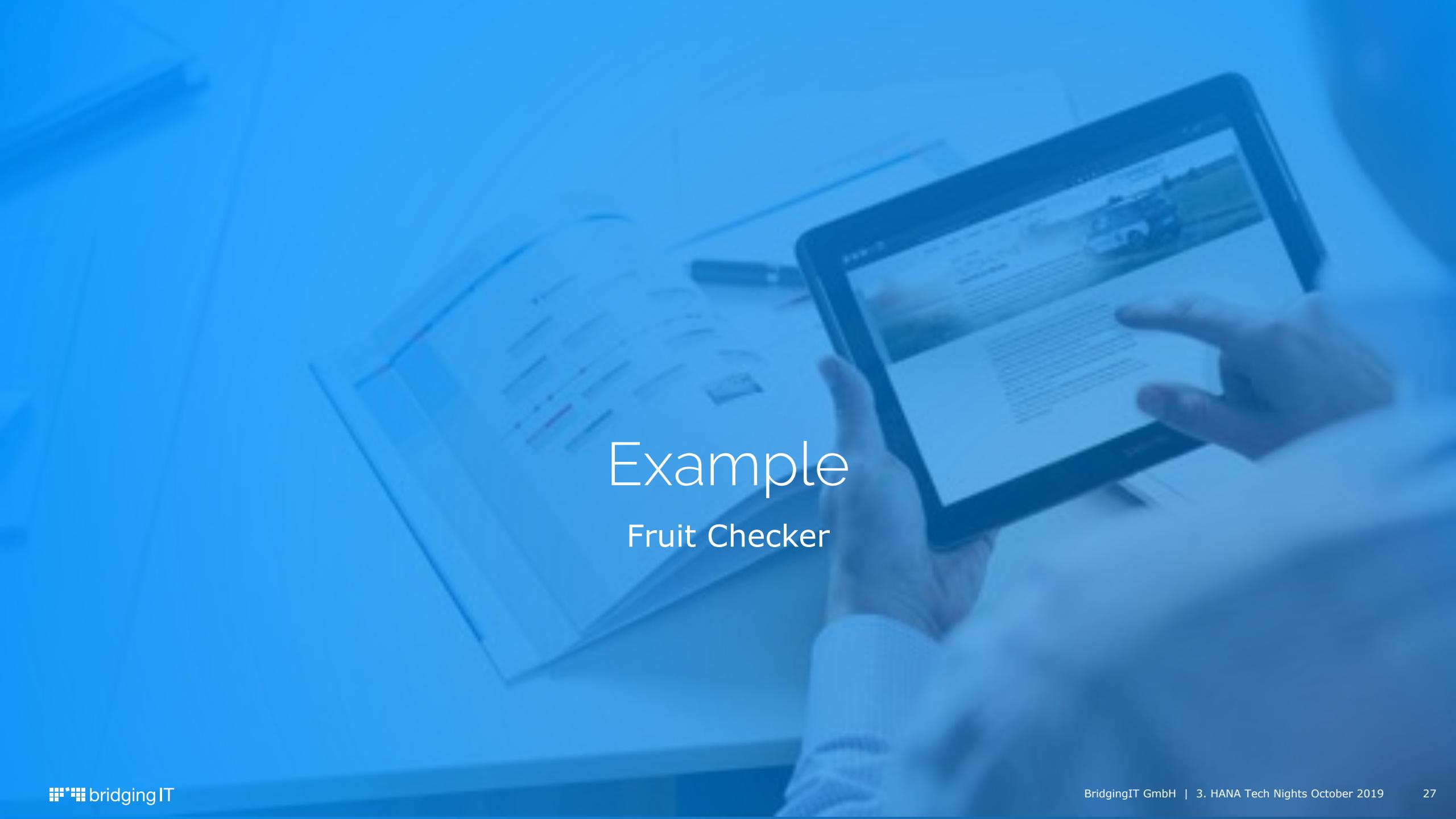
- Flavors: Java, Javascript / Node.JS
- Create / Extend SAP Services / Apps
- Develop and run on SAP Cloud / HANA



HANA Architecture



<https://www.creativityatwork.com/2014/06/10/busy-innovate/>



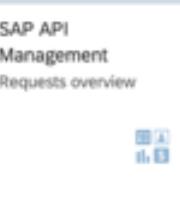
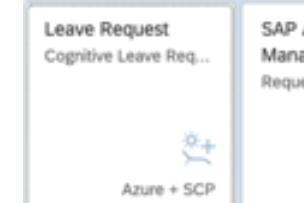
Example

Fruit Checker

Fruit Checker

Store Manager

Cognitive Leave Workflow Additional Information



Workflow

My Inbox
All Tasks
 33

Monitor Workflows
Workflow Instances


Monitor Workflows
Workflow Definitions


Additional Information

bridgingIT
Menschen Methoden...

Microsoft Cognitive Services
Infuse your apps, we...

SAP Translation Hub
Develop apps in one ...

SAP in Azure
Bring unparalleled p...

SAP API Management
Securely share your ...


SAP Cloud Fiori Launchpad

SAP Fiori 2.0

SAP CAPM App ausgeführt in Microsoft Azure Kubernetes

Odata v4 Backend

Fruit Checker

Manage store

The screenshot shows the 'Fruit Checker Admin Dashboard' interface. At the top, there's a navigation bar with icons for Home, Back, Forward, and Refresh. Below it, a breadcrumb trail reads: Home / Bridging IT / Microsoft / Azure / Fruit Checker Admin Dashboard. To the right of the breadcrumb, there are status indicators: Business Area: Retail, Technical Area: Innovation, and Cost Center: Machine Learning. A note says 'Availability: Only during SAPPHIRE NOW'. On the left, a sidebar titled 'Available stores (4)' lists four locations with their addresses: BridgingIT Frankfurt (Solmstraße 4, 60 486 Frankfurt am Main), BridgingIT Köln (Martinstraße 3, 50 847 Köln), BridgingIT Mannheim (N7 5-6, 68 161 Mannheim), and BridgingIT Stuttgart (Marienstraße 17, 70 178 Stuttgart). A search bar labeled 'Suchen' is at the top right of the dashboard area.

The screenshot shows the 'BridgingIT Stuttgart' store summary page. At the top, there's a circular logo with a blue grid pattern and a red dot in the center. The store name 'BridgingIT Stuttgart' and subtitle 'Kleiner City Store' are displayed. Below this, contact information for Tobias Hofmann is listed: 'I do the best to have the best fruits', phone numbers '+04912377889965' and '+49 0721 3434567', and email 'tobias.hofmann@bridging-it.de'. To the right, the address 'Marienstraße 17, 70 178 Stuttgart' and 'LongLat: 9.172568/48.772248' are shown. Below this, there are two tabs: 'ORDERS' (selected) and 'MAP'. The 'ORDERS' tab displays a table of three orders:

Fruit	Description	Delivery Date	Amount	Status
banana	Dringender Bedarf an Bananen	01.01.2019	12 kg	Submitted
pineapple	Brauchde mal wieder Ananas	01.01.2019	2 kg	Submitted
banana	We go Bana	21.05.2019	25 kg	Information

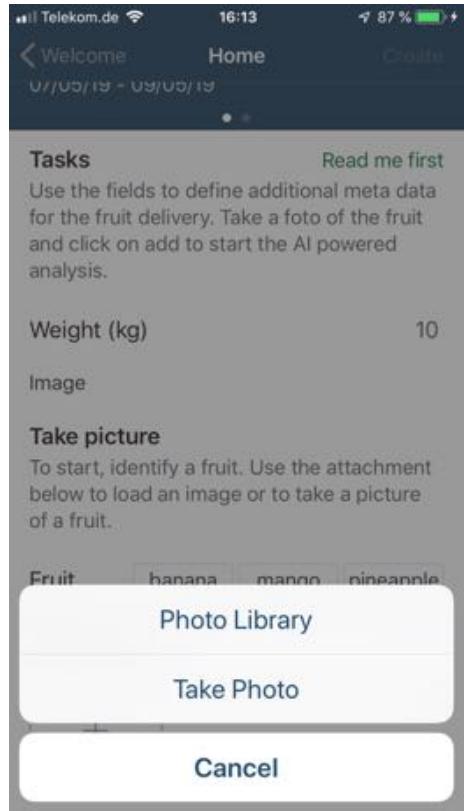
The 'MAP' tab is partially visible on the right. A large blue molecular structure icon is positioned to the right of the orders table.

Liste der verfügbaren Supermärkte in einer Tabelle mit Suche
Auswahl eines Supermarkts für die Anzeige der Bestellungen
Detailansicht der Liste der Bestellungen

Fruit Checker

App

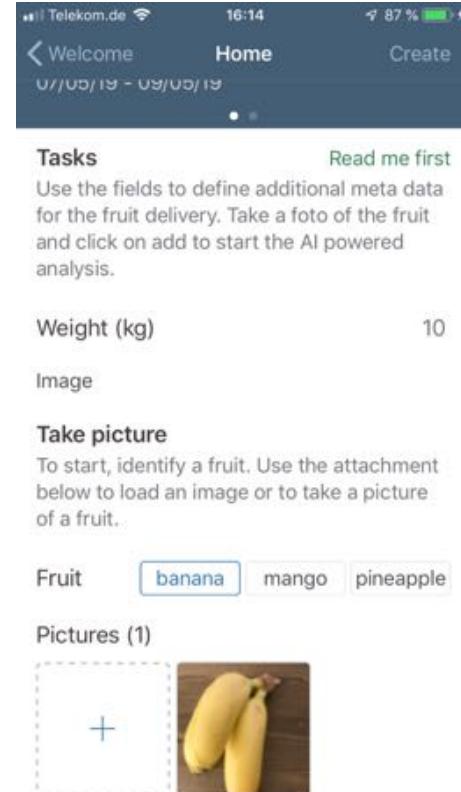
Picture



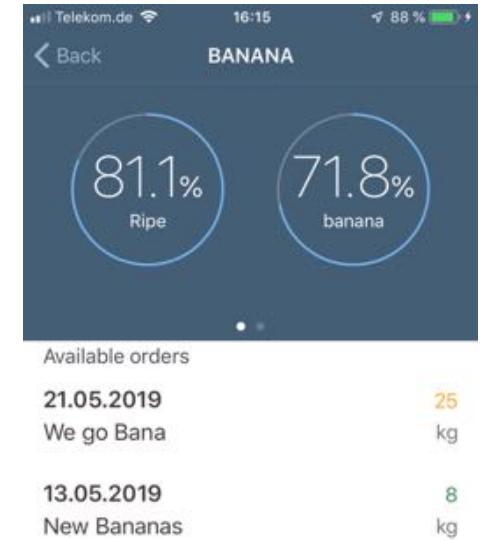
Take Picture



Edge Processing



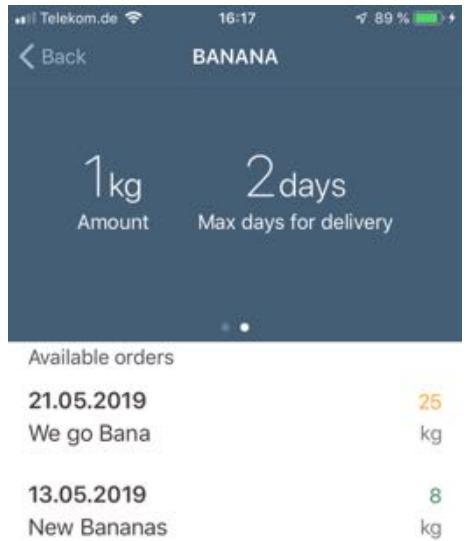
Analyse



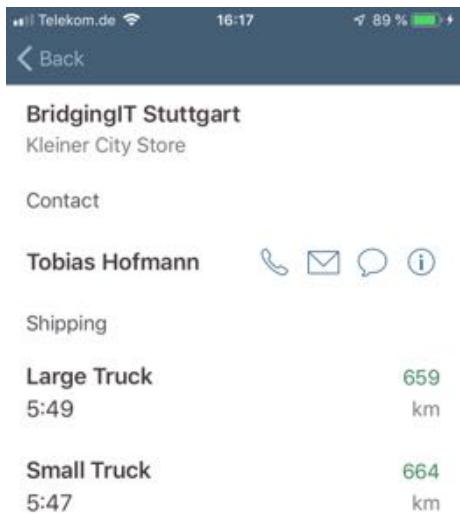
Fruit Checker

App

Analyse



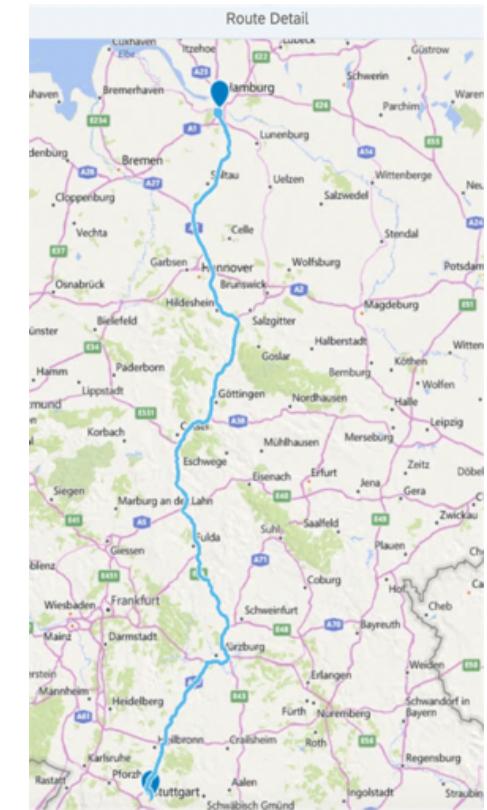
Select Route



Microsoft Truck API



Web





Yours is without a doubt the worst code I've ever run



But it runs

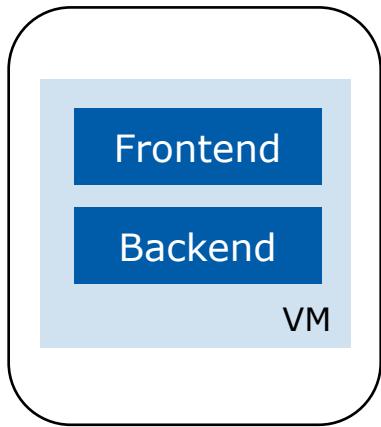
Most Innovative Project IA4SP



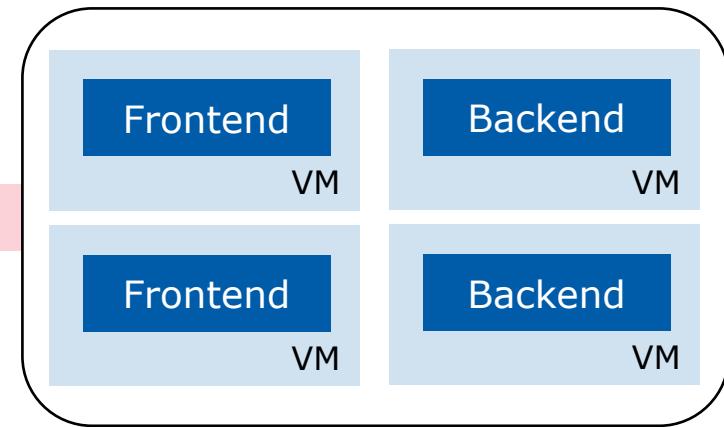
CAPM

Deployment

Classic

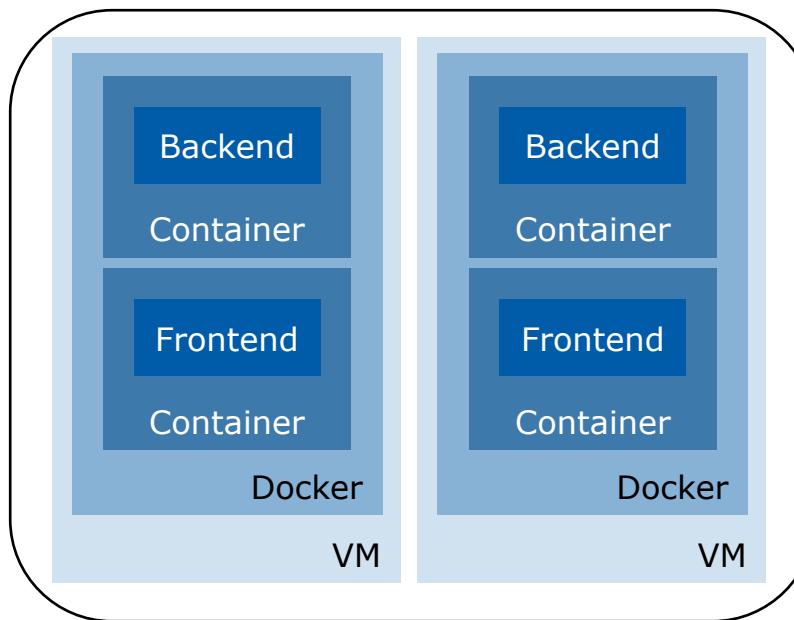


Classic cluster

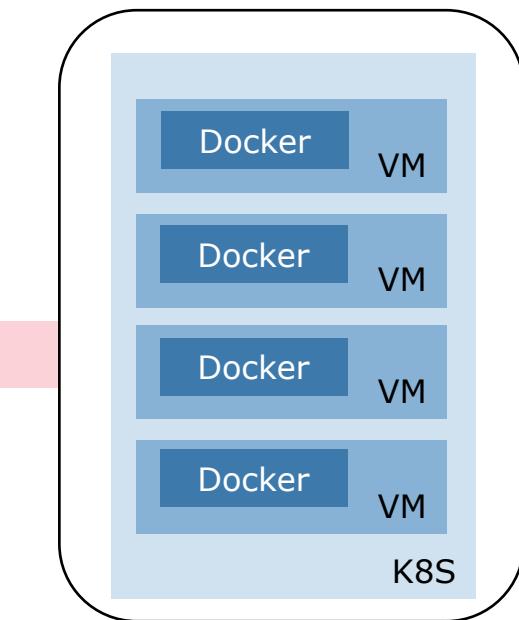


Connect / Deploy

Learning Docker



At scale



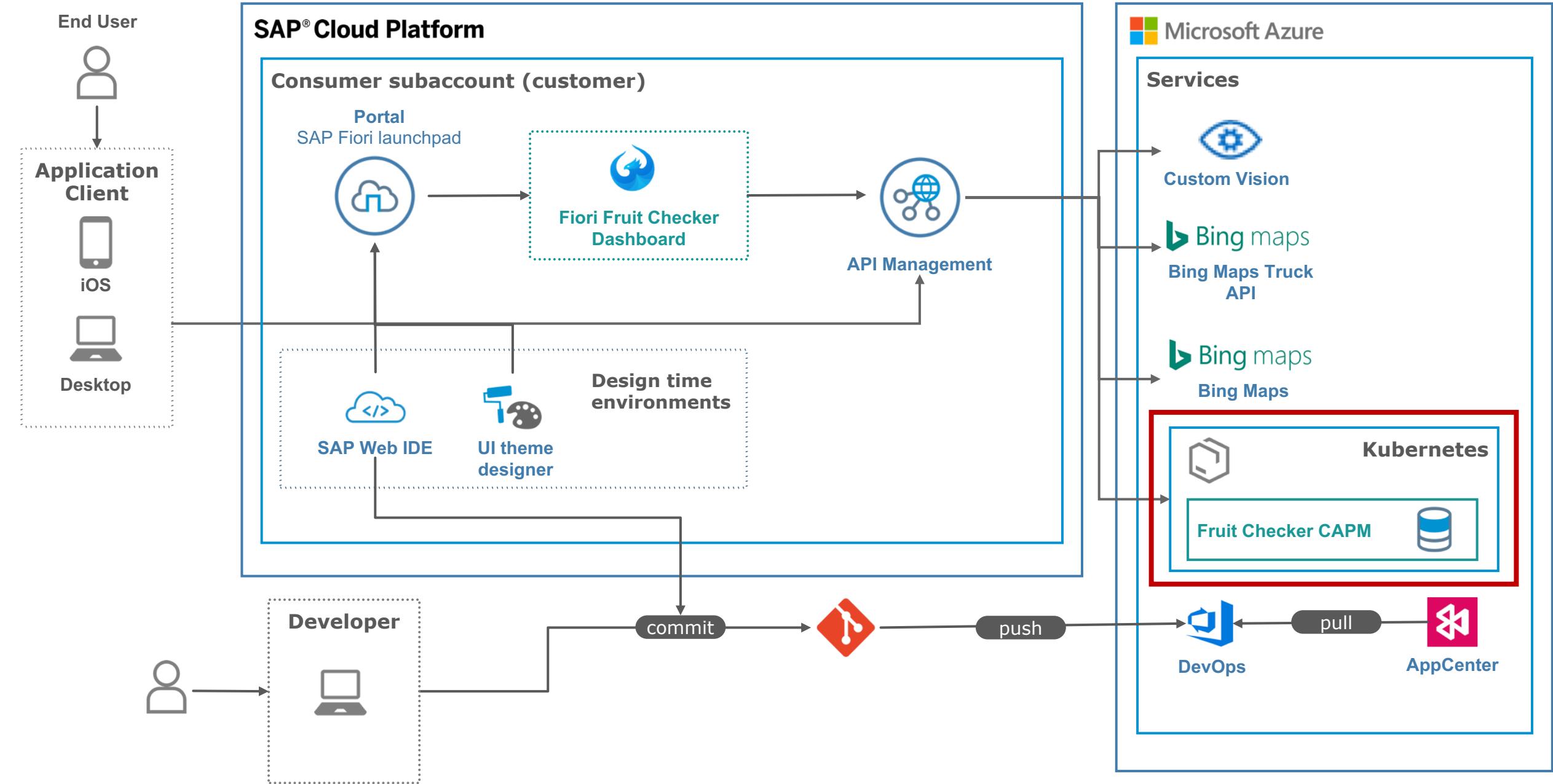
SCP

HANA

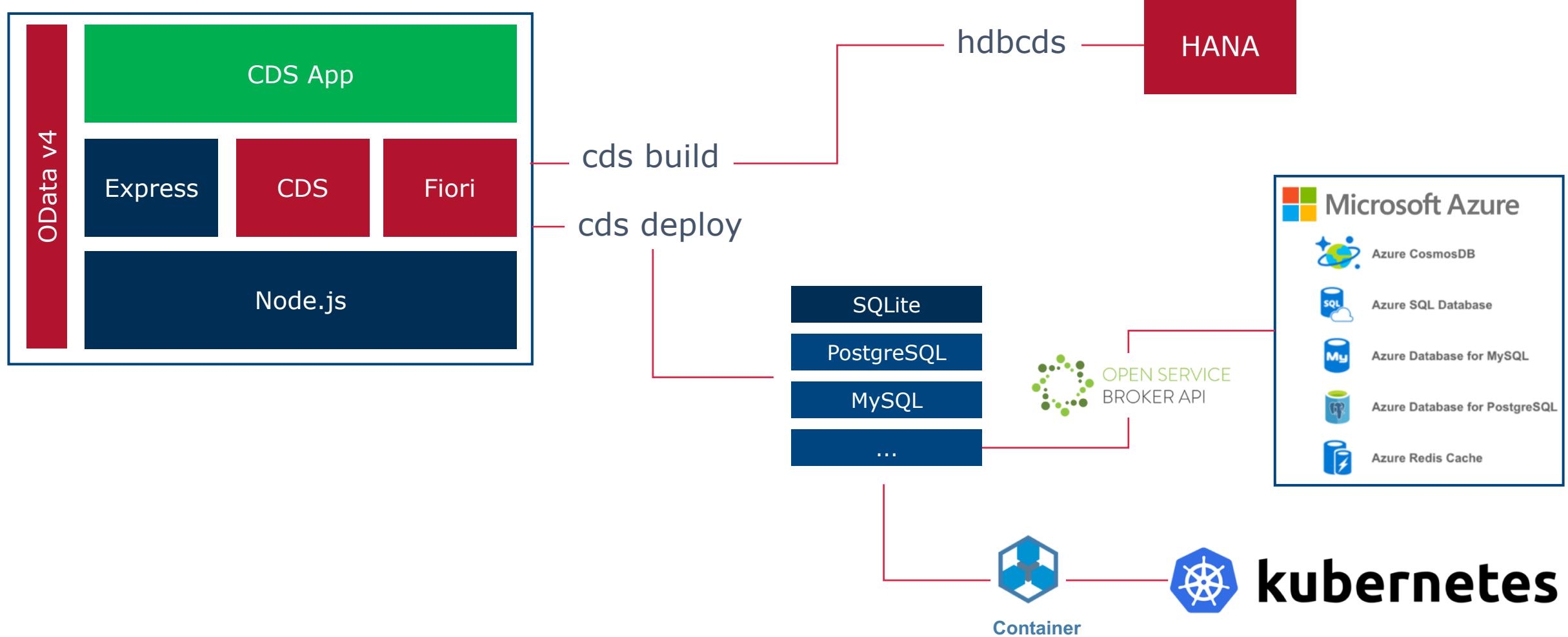
S/4HANA

SAP

Connect / Deploy



SAP CAP





SAP CAP Possibilities

Depends 100% on SAP and its plans with CAP

Create native cloud applications

Use your SAP CAP knowledge to create other apps

Full stack development with CDS & Fiori

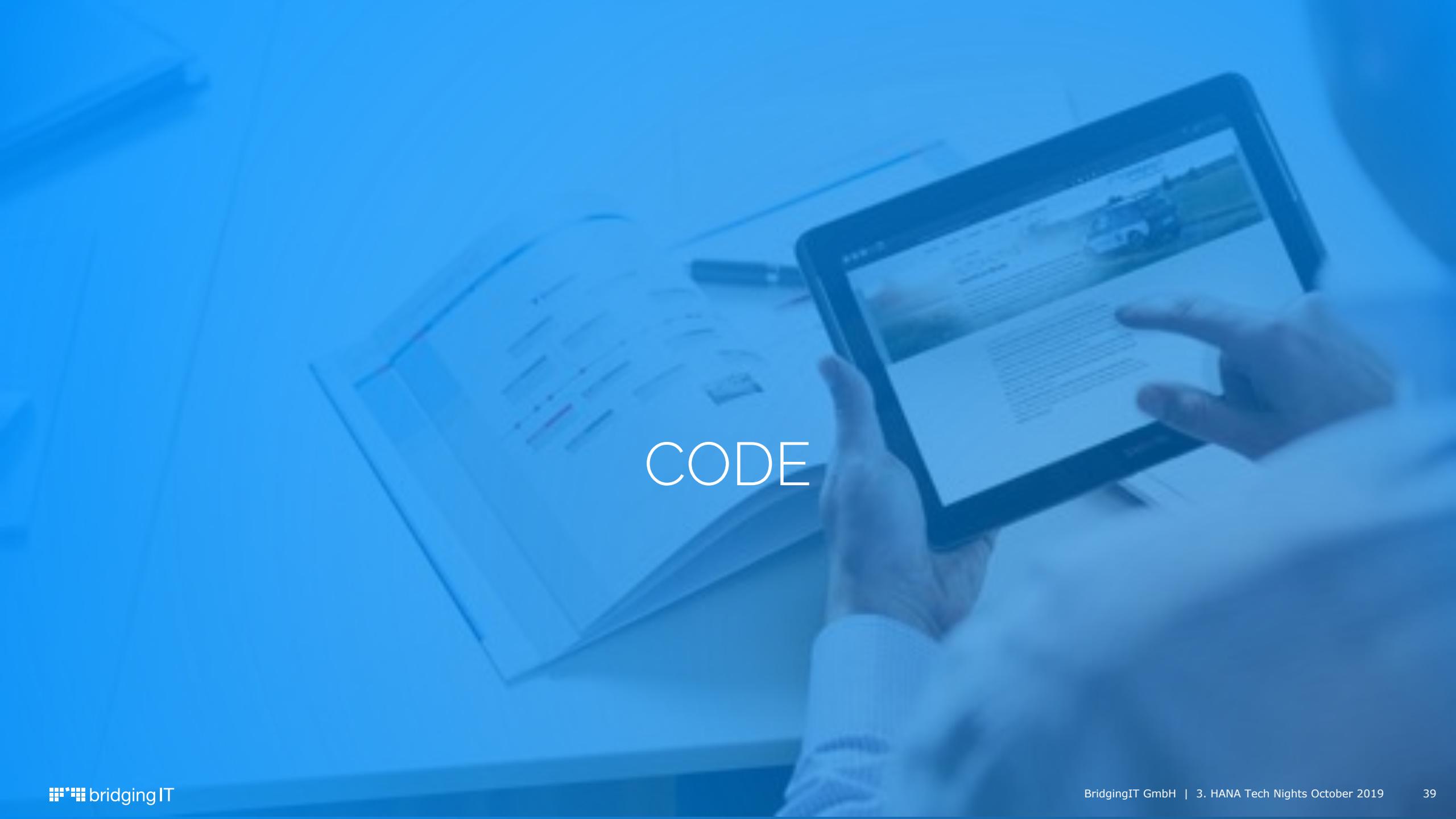
Talk a language people understand: Node.js, Javascript, NPM, Container, K8S

Run everywhere: real multicloud

Java devs can understand CDS: JPA, annotations

Extend with Open Source Tools

Cost efficiency. Go to HANA when it offers value to you



CODE

What we are going to develop

OData Service

Welcome to cds.services

These are the paths currently served ...

/domain / \$metadata

- [Domains](#) ...in Fiori

/product / \$metadata

- [Domains](#) ...in Fiori
- [ProductDetails](#) ...in Fiori
- [Products](#) ...in Fiori
- [Statuses](#) ...in Fiori
- [Vendors](#) ...in Fiori

/status / \$metadata

- [Status](#) ...in Fiori

/vendor / \$metadata

- [Vendors](#) ...in Fiori

Fiori Elements App

The screenshot shows a Fiori Elements application interface. At the top, there is a search bar and filters for Product, Domain, Vendor, and Status. Below this is a table titled "Products (4)" with columns for Product, Short Name, Vendor, Domain, and Status. One row is selected, showing "SAP Mobile Platform" with "SAP" as the vendor and "Mobile" as the domain. A modal window is open for this product, displaying its details: Product: SAP Mobile Platform, Short Name: SAP, Homepage: https://mmp.sap.com, Vendor: SAP (078ba304-fb1-4a0e-8af6-346ef6f5277ad), Status: GA (0980d4c6-1e58-4f51-8362-b3d39718ec84), and Domain: Mobile (0980d4c6-1e58-4f51-8362-b3d39718ec84). The modal also has tabs for Details and Releases. The Releases section shows two entries: Version 3.0.0 with Status GA, and Version 3.1.0 with Status Planned.

Product	Short Name	Vendor	Domain	Status
SAP Mobile Platform	SAP	SAP	Mobile	GA
Business Suite	ERP	SAP	Portal	GA
IoT Enablement				
Rod				

Version	Status	Documentation
3.0.0	GA	
3.1.0	Planned	

Project Setup

CAP is available via npm

Global

```
npm set @sap:registry=https://npm.sap.com
```

Installation

```
npm i -g @sap/cds
```

Local

NPM Registry Configuration: .npmrc

```
@sap:registry=https://npm.sap.com
```

CLI

```
npm config set @sap:registry https://npm.sap.com
npm install @sap/<node_package>
```

<https://help.sap.com/viewer/4505d0bdaf4948449b7f7379d24d0f0d/2.0.03/en-US/726e5d41462c4eb29eaa6cc83ff41e84.html>

Project Setup

Basic commands

CDS CLI

```
cds init <project name>
```

```
cds deploy  
cds run
```

Install SQLite database

```
npm i sqlite3 -D
```

```
cds deploy --to sqlite:db/<database name>.db
```

npm start

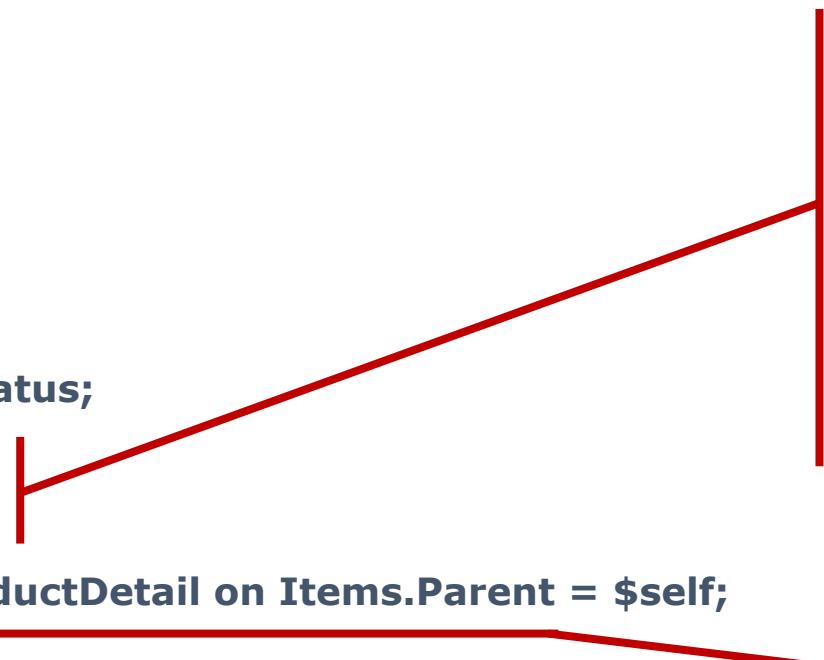
package.json:

```
  "start": "cds deploy && cds run",
```

<https://developers.sap.com/tutorials/cp-apm-nodejs-create-service.html>

Add Entities

```
namespace de.bit.eol;  
  
using { managed, cuid, sap.common.CodeList as CodeList } from  
'@sap/cds/common';  
  
entity Product: managed, cuid {  
    Name: String not null;  
    ShortName: String not null;  
    Homepage: String;  
SupportStatus: Association to Status;  
Vendor: Association to Vendors;  
Domain: Association to Domains;  
Items: Composition of many ProductDetail on Items.Parent = $self;  
}
```



```
entity Vendors : managed, cuid {  
    Name: String not null;  
    Homepage: String not null;  
}  
  
entity Domains : managed, cuid {  
    Name: String not null;  
    Description: String not null;  
}  
  
entity ProductDetail: managed, cuid {  
Parent: Association to Product not null;  
    Version: String;  
    Status: String not null;  
    Documentation: String;  
}
```

Enhance Entities

```
entity Product: managed, cuid {  
    Name: String not null @(  
        title: '{i18n>productName}',  
        description: '{i18n>productNameDescription}',  
        Common.FieldControl: #Mandatory  
    );  
  
    ShortName: String not null @(  
        title: '{i18n>productShortName}',  
        description: '{i18n>productShortNameDescription}',  
        Common.FieldControl: #Mandatory  
    );  
  
    Homepage: String @(  
        title: '{i18n>productHomepage}',  
        description: '{i18n>productHomepageDescription}',  
        Common.FieldControl: #Mandatory  
    );  
    ...
```

```
db/_i18n/i18n.properties  
  
#  
# Product  
#  
productName=Product  
productNamePlural=Products  
productNameDescription=Name of the product  
productShortName=Short Name  
productShortNameDescription=Short name of  
product  
productHomepage=Homepage  
productHomepageDescription=Homepage of the  
product
```

Expose as Service

```
using de.bit.eol as eol from '../db/data-model';

service VendorService {
    entity Vendors as projection on eol.Vendors;
    annotate Vendors with @odata.draft.enabled;
}

service DomainService {
    entity Domains as projection on eol.Domains;
    annotate Domains with @odata.draft.enabled;
}

service StatusService {
    entity Status as projection on eol.Status;
    annotate Status with @odata.draft.enabled;
}

service ProductService {
    entity Products as projection on eol.Product;
    entity ProductDetails as projection on eol.ProductDetail;
    entity Vendors @readonly as projection on eol.Vendors;
    entity Domains @readonly as projection on eol.Domains;
    entity Statuses @readonly as projection on eol.Status;
    annotate Products with @odata.draft.enabled;
}
```

cds run



Welcome to cds.services

These are the paths currently served ...

/domain / \$metadata

- [Domains ...in Fiori](#)

/product / \$metadata

- [Domains ...in Fiori](#)
- [ProductDetails ...in Fiori](#)
- [Products ...in Fiori](#)
- [Statuses ...in Fiori](#)
- [Vendors ...in Fiori](#)

/status / \$metadata

- [Status ...in Fiori](#)

/vendor / \$metadata

- [Vendors ...in Fiori](#)

OData Service

```
<edmx:Edmx xmlns:edmx="http://docs.oasis-open.org/odata/ns/edmx" Version="4.0">
  <edmx:Reference Uri="https://oasis-tcs.github.io/odata-vocabularies/vocabularies/Org.OData.Capabilities.V1.xml">
    <edmx:Include Alias="Capabilities" Namespace="Org.OData.Capabilities.V1"/>
  </edmx:Reference>
  <edmx:Reference Uri="https://wiki.scn.sap.com/wiki/download/attachments/448470974/Common.xml?api=v2">
    <edmx:Include Alias="Common" Namespace="com.sap.vocabularies.Common.v1"/>
  </edmx:Reference>
  <edmx:Reference Uri="https://oasis-tcs.github.io/odata-vocabularies/vocabularies/Org.OData.Core.V1.xml">
    <edmx:Include Alias="Core" Namespace="Org.OData.Core.V1"/>
  </edmx:Reference>
  <edmx:Reference Uri="https://wiki.scn.sap.com/wiki/download/attachments/448470968/UI.xml?api=v2">
    <edmx:Include Alias="UI" Namespace="com.sap.vocabularies.UI.v1"/>
  </edmx:Reference>
<edmx:DataServices>
  <Schema xmlns="http://docs.oasis-open.org/odata/ns/edm" Namespace="ProductService">
    <EntityContainer Name="EntityContainer">
      <EntityType Name="Domains" EntityType="ProductService.Domains"/>
      <EntityType Name="ProductDetails" EntityType="ProductService.ProductDetails">
        <NavigationPropertyBinding Path="Parent" Target="Products"/>
        <NavigationPropertyBinding Path="SiblingEntity" Target="ProductDetails"/>
      </EntityType>
      <EntityType Name="Products" EntityType="ProductService.Products">
        <NavigationPropertyBinding Path="SupportStatus" Target="Statuses"/>
        <NavigationPropertyBinding Path="Vendor" Target="Vendors"/>
        <NavigationPropertyBinding Path="Domain" Target="Domains"/>
        <NavigationPropertyBinding Path="Items" Target="ProductDetails"/>
        <NavigationPropertyBinding Path="SiblingEntity" Target="Products"/>
      </EntityType>
      <EntityType Name="Statuses" EntityType="ProductService.Statuses"/>
      <EntityType Name="Vendors" EntityType="ProductService.Vendors"/>
    </EntityContainer>
    <EntityType Name="Domains">
      <Key>
        <PropertyRef Name="ID"/>
      </Key>
      <Property Name="modifiedAt" Type="Edm.DateTimeOffset"/>
      <Property Name="createdAt" Type="Edm.DateTimeOffset"/>
      <Property Name="createdBy" Type="Edm.String" MaxLength="255"/>
      <Property Name="modifiedBy" Type="Edm.String" MaxLength="255"/>
      <Property Name="ID" Type="Edm.Guid" Nullable="false"/>
      <Property Name="Name" Type="Edm.String" Nullable="false"/>
      <Property Name="Description" Type="Edm.String" Nullable="false"/>
    </EntityType>
```

OData v4

Fiori annotations

```
using ProductService as product from './eol-service';
```

```
annotate product.Products with @()
```

```
    UI: {Identification: [{ Value:Name } ]},
```

```
UI.SelectionFields: [Name, Domain_ID, Vendor_ID, SupportStatus_ID ],
```

```
UI.LineItem: [
```

```
    {$Type: 'UI.DataField', Value: Name },  
    {$Type: 'UI.DataField', Value: ShortName },  
    {$Type: 'UI.DataField', Value: Vendor.Name },  
    {$Type: 'UI.DataField', Value: Domain.Name },  
    {$Type: 'UI.DataField', Value: SupportStatus.Name },
```

```
],
```

```
UI.HeaderInfo: {
```

```
    TypeName: '{i18n>productName}',
```

```
    Title: { Value: Name},
```

```
    Description: { Value: Homepage}
```

```
},
```

```
UI.Facets: [
```

```
    {$Type: 'UI.ReferenceFacet', Label: '{i18n>details}', Target:  
     '@UI.FieldGroup#ProductDetails'},
```

```
    {$Type: 'UI.ReferenceFacet', Label: '{i18n>releases}', Target:  
     'Items/@UI.LineItem'},
```

```
],
```

```
UI.FieldGroup#ProductDetails: {
```

```
    Label: '{i18n>productDetails}',
```

```
    Data: [
```

```
        {$Type: 'UI.DataField', Value: Name },
```

```
        {$Type: 'UI.DataField', Value: ShortName },
```

```
        {$Type: 'UI.DataField', Value: Homepage },
```

```
        {$Type: 'UI.DataField', Value: SupportStatus_ID, Label: '{i18n>status}' },
```

```
        {$Type: 'UI.DataField', Value: Vendor_ID },
```

```
        {$Type: 'UI.DataField', Value: Domain_ID },
```

```
    ]
```

```
}
```

```
);
```

Fiori annotations

Preview – List of ProductService.Products

The screenshot shows a Fiori application interface for a product list. At the top, there is a search bar, a dropdown for 'Editing Status' set to 'All', and several filter fields labeled 'Product', 'Domain', 'Vendor', and 'Status'. The 'Product' field is highlighted with a green background and the text 'UI.SelectionFields'. Below the header is a table titled 'Products (4)'. The table has columns for 'Product' (checkbox), 'Short Name', 'Vendor', 'Domain', and 'Status'. The 'Vendor' column is highlighted with a green background and the text 'UI.LineItem'. The table contains five rows of data:

Product	Short Name	Vendor	Domain	Status
SAP Mobile Platform	SAP	SAP	Mobile	GA
Business Suite	ERP	SAP	Portal	GA
IoT Enablement	IoT	Microsoft	IoT	GA
Fiori	Fiori	Oracle	Cloud	GA

At the bottom right of the table, there are 'Create' and 'Delete' buttons.

Fiori annotations

The screenshot shows the SAP Fiori annotations interface. At the top left, there is a green header bar with the text "SAP Mobile Platform" and "UI.HeaderInfo". On the right side of this bar are two buttons: "Bearbeiten" (Edit) and "Löschen" (Delete). Below this, there are two tabs: "Details" (selected) and "Releases". The main content area has a green header bar with the text "UI. Facets". Underneath, there are several data fields:

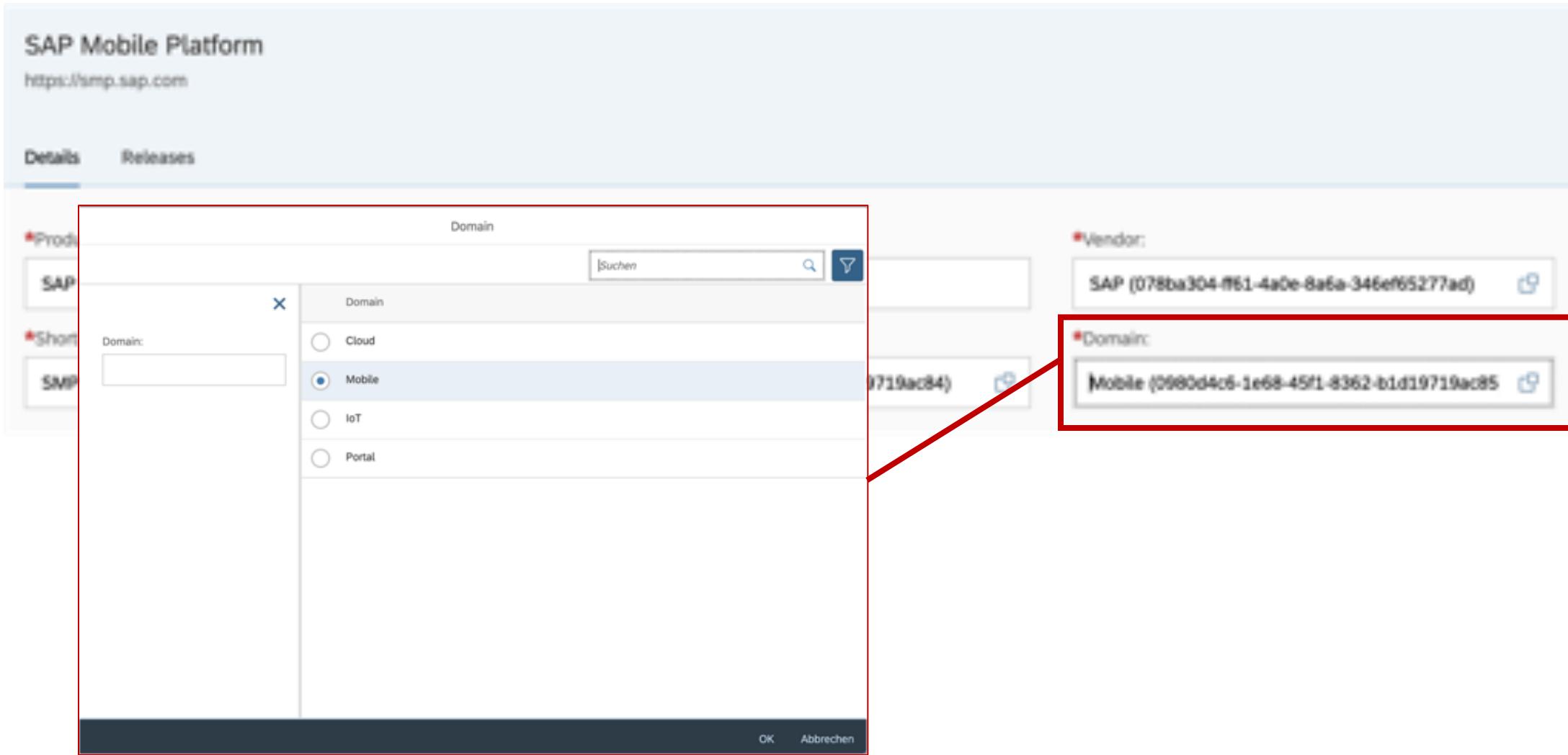
- Product: SAP Mobile Platform
- Homepage: <https://fmp.sap.com>
- Vendor: SAP (079be304-9603-4a0e-ba0a-346ef652777e)
- Short Name: SMP
- Status: GA (09800d01-1e00-4a0e-ba0a-346ef652777e)
- Domain: somewhere

In the center of the screen, there is a large bold text element: "UI.FieldGroup#ProductDetails".

Below the "UI. Facets" section, there is another green header bar with the text "Releases". Underneath, there is a table with one row:

Version	Label	Description
3.0.0	{\$Type: 'UI.ReferenceFacet', Label: '{i18n>releases}', Target: 'Items/@UI.LineItem'}	GA somewhere

Fiori annotations



Fiori annotations

```
annotate product.Products with {  
    Vendor @(  
        Common: {  
            Text: Vendor.Name,  
            FieldControl: #Mandatory  
        },  
        ValueList.entity:'Vendors',  
    );  
    Domain @(  
        Common: {  
            Text: Domain.Name,  
            FieldControl: #Mandatory  
        },  
        ValueList.entity:'Domains',  
    );  
};
```

Fiori annotations

```
service ProductService {  
    entity Products as projection on eol.Product;  
    entity ProductDetails as projection on eol.ProductDetail;  
    entity Vendors @readonly as projection on eol.Vendors;  
    entity Domains @readonly as projection on eol.Domains;  
    entity Statuses @readonly as projection on eol.Status;  
annotate Products with @odata.draft.enabled;  
}
```



Docker

```
.dockerignore  
.git  
node_modules  
npm-debug.log
```

```
Dockerfile  
FROM node:8  
  
LABEL website="www.bridging-it.de"  
  
WORKDIR /usr/src/app  
  
COPY . .  
  
RUN npm config set @sap:registry https://npm.sap.com  
RUN npm install  
  
EXPOSE 4004  
  
CMD [ "npm", "start" ]
```

Docker

```
.dockerignore  
.git  
node_modules  
npm-debug.log
```

```
Dockerfile  
FROM node:8  
  
LABEL website="www.bridging-it.de"  
  
WORKDIR /usr/src/app  
  
COPY . .  
  
RUN npm config set @sap:registry https://npm.sap.com  
RUN npm install  
  
EXPOSE 4004  
  
CMD [ "npm", "start" ]
```

Docker

```
docker build -t technight:ma .
```

```
docker run -p 5004:4004 technight:ma
```

<http://localhost:5004/>

Welcome to cds.services

These are the paths currently served ...

[/domain / \\$metadata](#)

- [Domains](#) ...in Fiori

[/product / \\$metadata](#)

- [Domains](#) ...in Fiori
- [ProductDetails](#) ...in Fiori
- [Products](#) ...in Fiori
- [Statuses](#) ...in Fiori
- [Vendors](#) ...in Fiori

[/status / \\$metadata](#)

- [Status](#) ...in Fiori

[/vendor / \\$metadata](#)

- [Vendors](#) ...in Fiori

Kubernetes

YAML configuration for Azure

```
apiVersion: apps/v1beta1
kind: Deployment
metadata:
  name: sapcapm
spec:
  replicas: 1
  strategy:
    rollingUpdate:
      maxSurge: 1
      maxUnavailable: 1
  minReadySeconds: 5
  template:
    metadata:
      labels:
        app: sapcapm
    spec:
      containers:
        - name: scpcapm
          - name: scpcapm
            image: appcontiner.azurecr.io/scpcapm:v1
            imagePullPolicy: "Always"
            ports:
              - containerPort: 4004
            resources:
              requests:
                cpu: 250m
              limits:
                cpu: 500m
  ---  
  apiVersion: v1
  kind: Service
  metadata:
    name: sapcapm
  spec:
    type: LoadBalancer
    ports:
      - port: 80
    selector:
      app: sapcapm
```

LESSONS LEARNED

Odata v4

Events

Summary

Event	Description
parseError	The 'parseError' event is not supported by this model. <i>Since: 1.37.0.</i>
propertyChange	The 'propertyChange' event is not supported by this model. <i>Since: 1.37.0.</i>
requestCompleted	The 'requestCompleted' event is not supported by this model. <i>Since: 1.37.0.</i>
requestFailed	The 'requestFailed' event is not supported by this model. <i>Since: 1.37.0.</i>
requestSent	The 'requestSent' event is not supported by this model. <i>Since: 1.37.0.</i>
sessionTimeout	The 'sessionTimeout' event is fired when the server has created a session for the model and this session ran into a timeout due to inactivity. <i>Since: 1.66.0.</i>

The model does not support the methods `getData`, `getObject`, `getOriginalProperty`, `getProperty`. For data access, use the context API instead of methods on the model.

OData operations executed via binding: Model does not support the method `callFunction`.

No CRUD methods on model: Model does not support the methods `create`, `read`, `remove`, `update`.

No metadata access via model: Model does not support methods `getServiceAnnotations`, `getServiceMetadata`, `refreshMetadata` as well as methods corresponding to the events `metadataFailed`, `metadataLoaded`.

**NOT
SUPPORTED**

Odata v4

Goal

Simplification

Access data via bindings & controls



Incompatibility

Going from Odata v2 to v4 breaks code

Limitations for established patterns

Odata v2 != Odata v4

Do NOT use Odata v4 in UI5 app if you can avoid it

<https://openui5.hana.ondemand.com/#/api/sap.ui.model.odata.v4.ODatasModel>
<https://openui5.hana.ondemand.com/#/topic/abd4d7c7548d4c29ab8364d3904a6d74.html>

API Management



Make APIs available

Bundle APIs into a product
Manage & control access

APIKey in HTTP Header ← **Not with Odata v4!**
Alter requests
Transform API into a service

Enterprise Ready

Monitoring

Know if the app meets user expectation: based on numbers and real data

Important: faster is not a number

Example: Distributed tracing with Elastic APM



Kubernetes

Monitoring

Leverage multi cloud

Deploy & Run in Container on K8S

Select container runtime of your choice

Example: Azure K8S

Clusterkapazität gesamt

Kerne 2 vCPUs

Arbeitsspeicher 7 GiB



Clusterkapazität gesamt

Kerne 2 vCPUs → 200 vCPUs (neu)

Arbeitsspeicher 7 GiB → 700 GiB (neu)





Wir freuen uns auf Sie!

Mannheim
N7, 5-6
68161 Mannheim

Karlsruhe
Rüppurrer Str. 4
76137 Karlsruhe

Frankfurt
Solmsstraße 4
60486 Frankfurt

Stuttgart
Marienstraße 17
70178 Stuttgart

Köln
Martinstraße 3
50667 Köln

München
Riesstraße 12
80992 München

Zug (CH)
Baarerstraße 14
CH-6300 Zug

Nürnberg
Königstorgraben 11
90402 Nürnberg

Berlin
Friedrichstr. 68
10117 Berlin