



CAP outside SAP

Tobias Hofmann

Lead SAP Architect, bridgingIT GmbH

25/06/2021

CAP

Agnostic Design -> For the architect

Open and Opinionated -> For the developer

	CDS/deploy	Node.js	Java
SAP HANA out of the box	✓	✓	✓
SAP HANA Cloud	✓	✓	✓
SQLite out of the box ¹	✓	✓	✓
MongoDB out of the box	n/a	planned	in prog.
Pluggable drivers architecture	in prog.	in prog.	in prog.
Out-of-the-box support for other DBs?	contrib?	contrib ²	contrib?

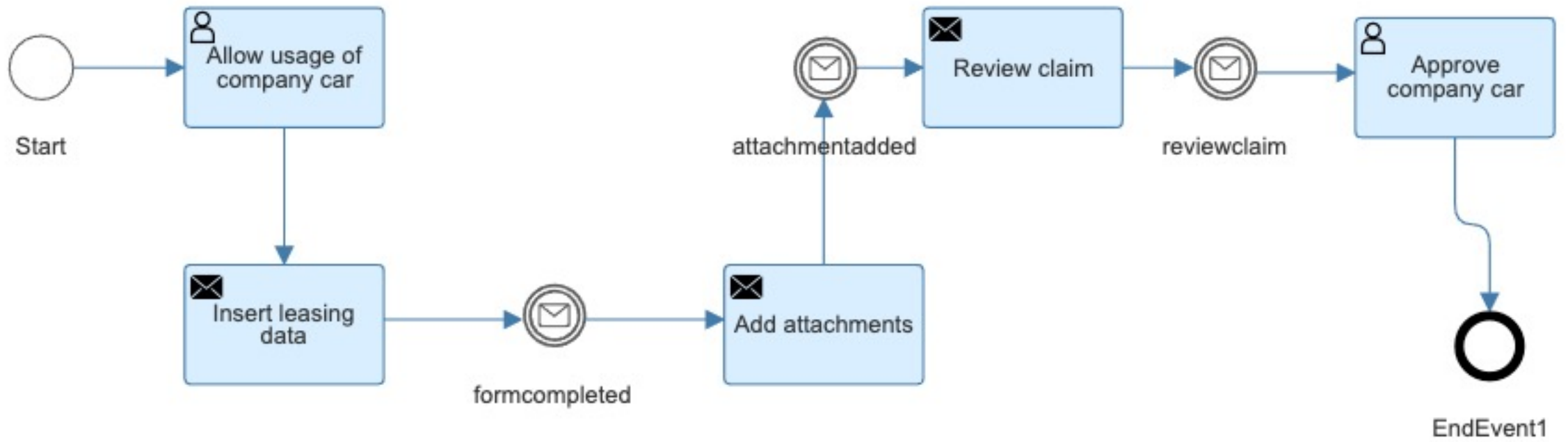
	Node.js	Java
Deploy to/run on SAP BTP, Cloud Foundry environment	✓	✓
Deploy to/run on Kubernetes ¹	planned	planned
Deploy to/run on Kyma ²	planned	planned
SaaS on-/off-boarding	✓	✓
Multitenancy	✓	✓
Health checks	planned	✓



Request a company car

Workflow

Example POC for a process to claim a company car

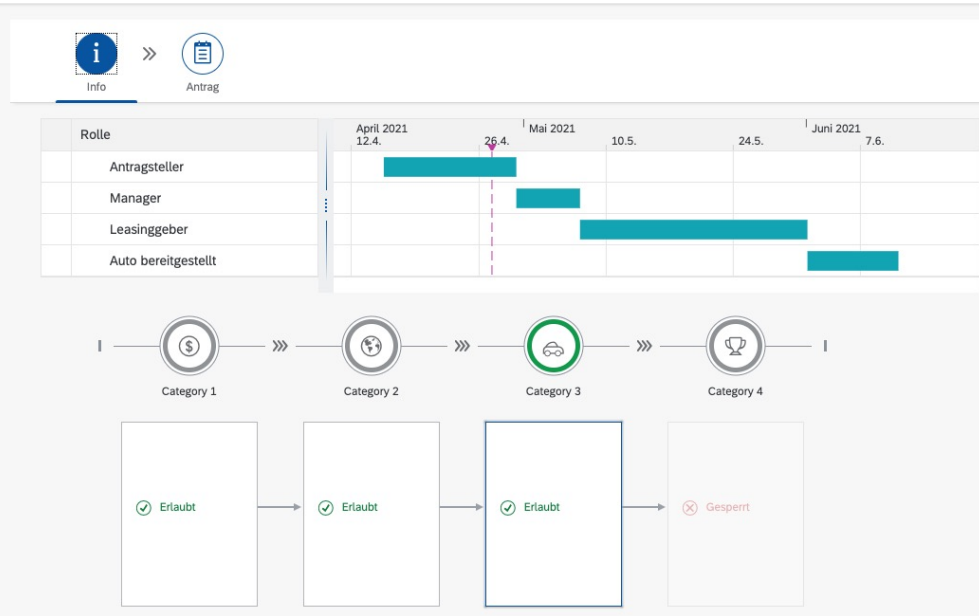


Antrag

Antragsteller: Tobias Hofmann
Manager: Martin Fischer

Skillstufe: Senior Consultant

Datum: 27 04 2021



Antrag

Antragsteller: Tobias Hofmann
Manager: Martin Fischer

Skillstufe: Senior Consultant

Datum: 27 04 2021

The form for 'Antrag' consists of three steps: 1. Kategorie auswählen, 2. Antriebsart, and 3. Persönliche Kalkulation.

1. Kategorie auswählen

Frage deinen Manager aus welchen Kategorien du wählen darfst

Category 1 Zuzahlung 200 €	Category 2 Zuzahlung 125 €	Category 3 Zuzahlung 75 €	Category 4 Zuzahlung 50 €
CO2 159	CO2 152	CO2 142	Für alle die einen kleinen Wagen... BridgingIT GmbH

2. Antriebsart

Die gewählte Antriebsart bestimmt auch die Besteuerung!

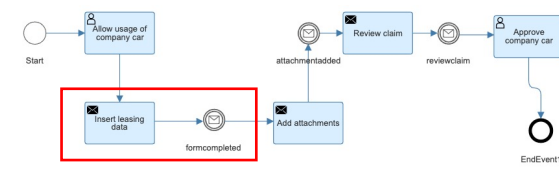
Antriebsart: * Hybrid

Ladestation: JA

3. Persönliche Kalkulation

Leasingrate:	300
Wartung:	150
Effizienzklasse:	A+
Verbrauch:	2
Preis:	50499

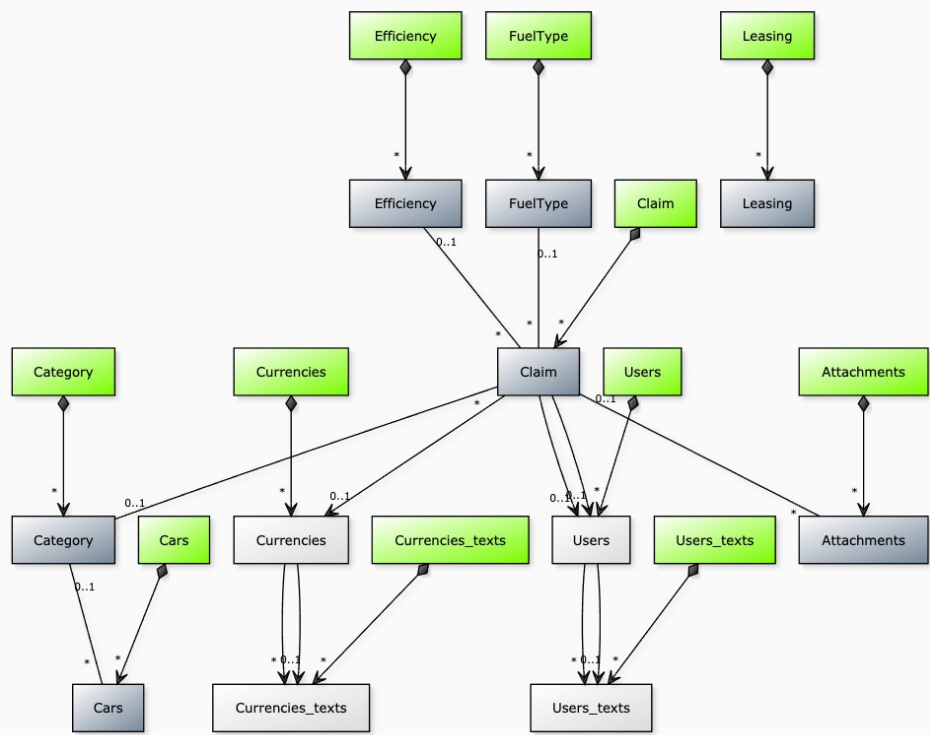
Überprüfen



OPEN API SUPPORT

Entity Data Model

Entity Data Model



CREATED WITH YUML

Attachments

Cars

Category

Claim

GET /Attachments({ID})/claim Retrieve claim of a attachment.

GET /Category({ID})/claim Retrieve a list of claim of a category.

POST /Category({ID})/claim Create a single claim of a category.

GET /Claim Retrieve a list of claim.

POST /Claim Create a single claim.

GET /Claim({ID}) Retrieve a single claim.

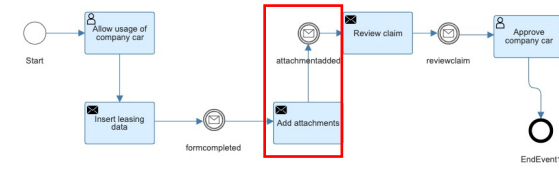
PATCH /Claim({ID}) Change a single claim.

DELETE /Claim({ID}) Delete a single claim.

GET /Claim({ID})/efficiency Retrieve efficiency of a claim.

GET /Claim({ID})/category Retrieve category of a claim.





Antrag verwalten

Anträge (2)

Suchen 

Datum	Kategorie	Effizienzklasse	Antriebsart	Leasing/Monat	Listenpreis (brutto)	
27.April.2021	Category 3	A+	Hybrid	300	50499.00	>
27.April.2021	Category 3	A+	Diesel	504.34	56832.00	>

Wert: 50.499,00 €

Antragsteller: Tobias Hofmann
Manager: Martin Fischer

Angelegt am: 27. April 2021 Leasingunternehmen: VW
Geändert am: 27. April 2021

Kategorie:	Category 3
Antriebsart:	Hybrid
Effizienzklasse:	A+
Leasing/Monat:	300,00 €
Service/Monat:	150,00 €
Verbrauchsangabe:	2
Listenpreis (brutto):	50.499,00 €
Mehrkosten Leasing/Service:	50.499,00 €
Ersparnis Verbrauchsangabe:	50.499,00 €
Mehrkosten Summe:	50.499,00 €

Upload selected Hochladen

- ☐  Dummy 2.docx  
- ☐  Dummy.docx  

Freigeben



Manager approval

Anträge (2) [Create](#) [Delete](#) [↑↓](#) [...](#)

☐	Preis	Kategorie	
☐	56,832	Category 3	>
antragsteller: Tobias			
Wartung: 34			
☐	60,000	Category 1	>
antragsteller: Tobias			
Wartung: 346			

Tobias



[Daten](#) [Anhänge](#)

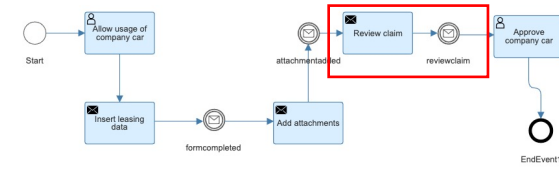
Daten

Preis:	Verbrauch:	Name:
56,832	4	Diesel
Leasingrate:	Name:	Ladestation:
504	A+	No
Wartung:	Kategorie:	
34	Category 3	

Anhänge

Anhänge

Datei:
Dummy 2.docx · SSHFS.docx



[Edit](#) [Delete](#) [Freigeben](#) [↗](#) [↖](#) [✕](#)



Make it enterprise ready

CLOUD NATIVE

CloudFoundry
Azure
Docker
OpenShift
SAP
DBaaS
PaaS
FaaS
Hybrid
AWS
GCP
App Services

DATA

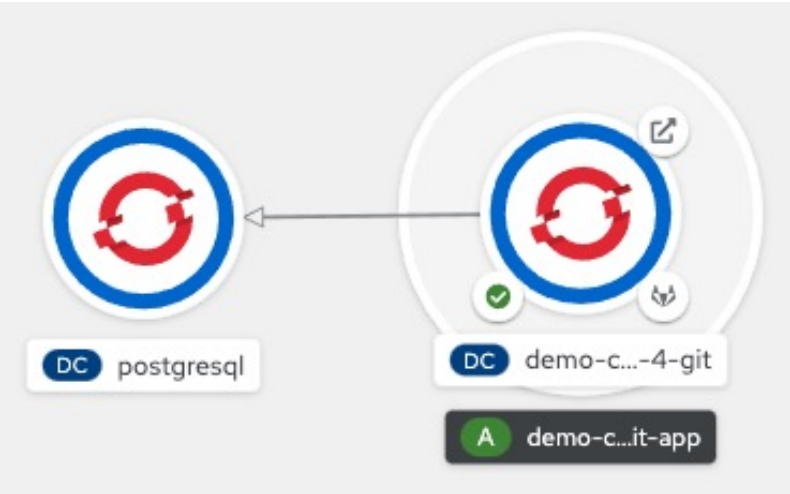
DBaaS
CRUD
Resilience
CosmosDB
DBaaS
Hybrid
HANA
Cluster
Stateful
MariaDB
MySQL

SINGLE SIGN-ON

Anonymous
CosmosDB
Authentication
SAML 2.0
Keycloak
SSO
X.509
2FA
Profiles
OpenID
Auth0
UIDPW



CAP ON OPENSSHIFT 4 WITH POSTGRESQL



Single values (env)

NAME	VALUE
POSTGRESQL_USER	postgresql - Secret database-user
POSTGRESQL_PASSWORD	postgresql - Secret database-password
POSTGRESQL_DATABASE	postgresql - Secret database-name

Single values (env)

NAME	VALUE
CDS_REQUIRES_DB_CREDENTIALS_DATABASE	postgresql - Secret database-name
CDS_REQUIRES_DB_CREDENTIALS_HOST	capire - Secret host
CDS_REQUIRES_DB_CREDENTIALS_USER	postgresql - Secret database-user
CDS_REQUIRES_DB_CREDENTIALS_PASSWORD	postgresql - Secret database-password
CDS_REQUIRES_DB_CREDENTIALS_PORT	capire - Secret port
CDS_REQUIRES_DB_CREDENTIALS_SCHEMA	capire - Secret schema



CAP ON OPENSIFT 4 WITH POSTGRESQL

<http://demo-cap-pg-os-4-git-cap-pg.apps-crc.testing/>

Welcome to @sap/cds Server

Serving cap-pg 1.0.0

These are the paths currently served ...

Web Applications:

– none –

Service Endpoints:

/public / \$metadata

- **Projects** → Fiori preview
- **Votes** → Fiori preview

This XML file does not appear to have any style information associated with it. The document tree is shown below.

```
<?xml version="1.0"?>
<edmx:Edmx xmlns:edmx="http://docs.oasis-open.org/odata/ns/edmx" Version="4.0">
  <edmx:DataServices>
    <Schema xmlns="http://docs.oasis-open.org/odata/ns/edm" Namespace="PublicService">
      <EntityContainer Name="EntityContainer">
        <EntitySet Name="Projects" EntityType="PublicService.Projects">
          <NavigationPropertyBinding Path="votes" Target="Votes"/>
        </EntitySet>
        <EntitySet Name="Votes" EntityType="PublicService.Votes">
          <NavigationPropertyBinding Path="project" Target="Projects"/>
        </EntitySet>
      </EntityContainer>
      <EntityType Name="Projects">
        <Key>
          <PropertyRef Name="ID"/>
        </Key>
        <Property Name="ID" Type="Edm.Guid" Nullable="false"/>
        <Property Name="name" Type="Edm.String" MaxLength="150"/>
        <Property Name="language" Type="Edm.String" MaxLength="150"/>
        <Property Name="repository" Type="Edm.String"/>
        <NavigationProperty Name="votes" Type="Collection(PublicService.Votes)" Partner="project"/>
      </EntityType>
      <EntityType Name="Votes">
        <Key>
          <PropertyRef Name="ID"/>
        </Key>
        <Property Name="ID" Type="Edm.Guid" Nullable="false"/>
        <Property Name="username" Type="Edm.String" MaxLength="150"/>
        <Property Name="createdAt" Type="Edm.DateTimeOffset"/>
        <NavigationProperty Name="project" Type="PublicService.Projects" Partner="votes">
          <ReferentialConstraint Property="project_ID" ReferencedProperty="ID"/>
        </NavigationProperty>
        <Property Name="project_ID" Type="Edm.Guid"/>
      </EntityType>
    </Schema>
  </edmx:DataServices>
</edmx:Edmx>
```



CAP ON OPENSSHIFT 4 WITH POSTGRESQL

Postman

The image shows the Postman application interface. At the top, a POST request is configured to the URL `http://demo-cap-pg-os-4-git-cap-pg.apps-crc.testing/public/Projects`. The 'Body' tab is selected, and the data format is set to 'JSON'. The request body contains a JSON object with the following fields:

```
{
  "name": "lorem ipsum",
  "language": "dummy",
  "repository": "the cloud"
}
```

Below the request, the 'Body' tab of the response is selected, showing the received JSON data in 'Pretty' format:

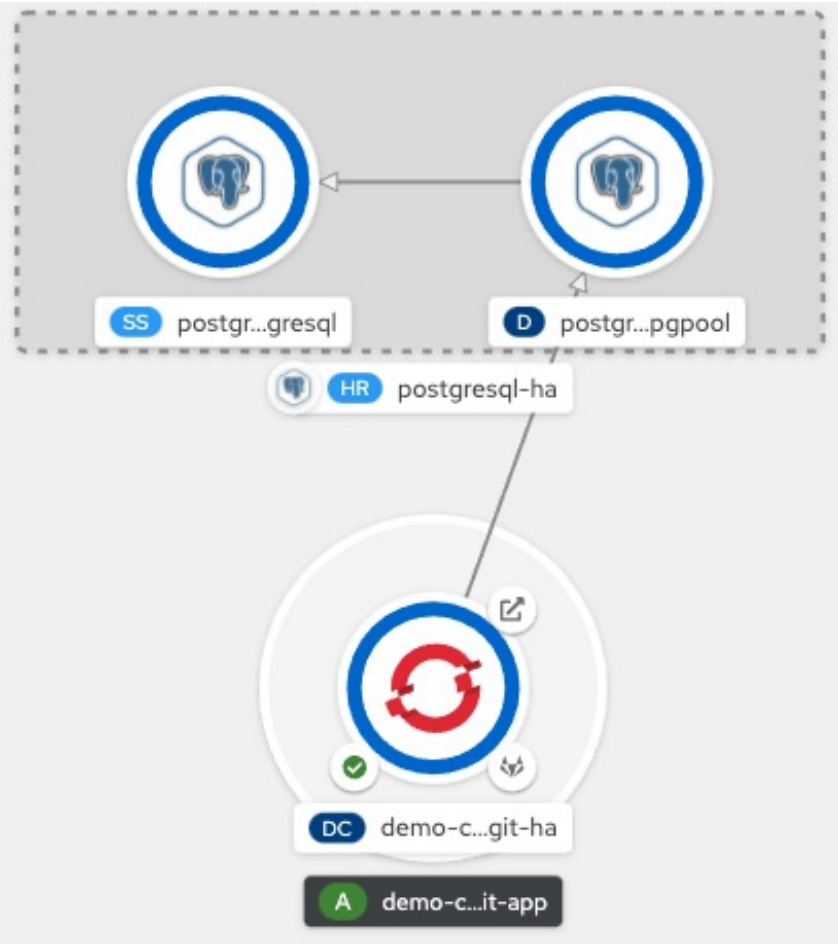
```
{
  "@odata.context": "$metadata#Projects/$entity",
  "ID": "a4be1ba7-b05b-446d-bd04-e6f82a472496",
  "name": "lorem ipsum",
  "language": "dummy",
  "repository": "the cloud"
}
```

In the top right corner, a browser window snippet shows the URL `demo-cap-pg-os-4-git-cap-pg.apps-crc.testing/public/Projects?$top=11` and a JSON response:

```
{ "@odata.context": "$metadata#Projects", "value": [ { "ID": "a4be1ba7-b05b-446d-bd04-e6f82a472496", "name": "lorem ipsum", "language": "dummy", "repository": "the cloud" } ] }
```



CAP ON OPENSSHIFT 4 WITH POSTGRESQL HA



Single values (env)

NAME	VALUE
CDS_REQUIRES_DB_CREDENTIALS_DATABASE	capire - Secret
CDS_REQUIRES_DB_CREDENTIALS_DATABASE	CDS_REQUIRES_DB_CREDENTIALS_DATABASE
CDS_REQUIRES_DB_CREDENTIALS_HOST	capire - Secret
CDS_REQUIRES_DB_CREDENTIALS_HOST	CDS_REQUIRES_DB_CREDENTIALS_HOST
CDS_REQUIRES_DB_CREDENTIALS_PORT	capire - Secret
CDS_REQUIRES_DB_CREDENTIALS_PORT	CDS_REQUIRES_DB_CREDENTIALS_PORT
CDS_REQUIRES_DB_CREDENTIALS_USER	capire - Secret
CDS_REQUIRES_DB_CREDENTIALS_USER	CDS_REQUIRES_DB_CREDENTIALS_USER
CDS_REQUIRES_DB_CREDENTIALS_PASSWORD	postgresql-ha-postgresql - Secret
CDS_REQUIRES_DB_CREDENTIALS_PASSWORD	postgresql-password



CAP ON OPENSSHIFT 4 WITH POSTGRESQL HA

<http://demo-cap-pg-os-4-git-ha-cap-pg-ha.apps-crc.testing/>

Welcome to @sap/cds Server

Serving cap-pg 1.0.0

These are the paths currently served ...

Web Applications:

– none –

Service Endpoints:

/public / \$metadata

- **Projects** → Fiori preview
- **Votes** → Fiori preview

This XML file does not appear to have any style information associated with it. The document tree is shown below.

```
<?xml version="1.0"?>
<edmx:Edmx xmlns:edmx="http://docs.oasis-open.org/odata/ns/edmx" Version="4.0">
  <edmx:DataServices>
    <Schema xmlns="http://docs.oasis-open.org/odata/ns/edm" Namespace="PublicService">
      <EntityContainer Name="EntityContainer">
        <EntitySet Name="Projects" EntityType="PublicService.Projects">
          <NavigationPropertyBinding Path="votes" Target="Votes"/>
        </EntitySet>
        <EntitySet Name="Votes" EntityType="PublicService.Votes">
          <NavigationPropertyBinding Path="project" Target="Projects"/>
        </EntitySet>
      </EntityContainer>
      <EntityType Name="Projects">
        <Key>
          <PropertyRef Name="ID"/>
        </Key>
        <Property Name="ID" Type="Edm.Guid" Nullable="false"/>
        <Property Name="name" Type="Edm.String" MaxLength="150"/>
        <Property Name="language" Type="Edm.String" MaxLength="150"/>
        <Property Name="repository" Type="Edm.String"/>
        <NavigationProperty Name="votes" Type="Collection(PublicService.Votes)" Partner="project"/>
      </EntityType>
      <EntityType Name="Votes">
        <Key>
          <PropertyRef Name="ID"/>
        </Key>
        <Property Name="ID" Type="Edm.Guid" Nullable="false"/>
        <Property Name="username" Type="Edm.String" MaxLength="150"/>
        <Property Name="createdAt" Type="Edm.DateTimeOffset"/>
        <NavigationProperty Name="project" Type="PublicService.Projects" Partner="votes">
          <ReferentialConstraint Property="project_ID" ReferencedProperty="ID"/>
        </NavigationProperty>
        <Property Name="project_ID" Type="Edm.Guid"/>
      </EntityType>
    </Schema>
  </edmx:DataServices>
</edmx:Edmx>
```



CAP ON OPENSIFT 4 WITH POSTGRESQL HA

Postman

POST ▼ http://demo-cap-pg-os-4-git-ha-cap-pg-ha.apps-crc.testing/public/Projects

Params Authorization Headers (10) **Body** Pre-request Script Tests (●) Settings

☐ none ☐ form-data ☐ x-www-form-urlencoded ☒ raw ☐ binary ☐ GraphQL **JSON** ▼

```
1 {
2   ... "name": "cluster test",
3   ... "language": "EN",
4   ... "repository": "pg cluster in open shift"
5 }
```

Body Cookies (1) Headers (7) Test Results (●)

Pretty Raw Preview Visualize **JSON** ▼ 

```
1 {
2   "@odata.context": "$metadata#Projects/$entity",
3   "ID": "82c42c89-182a-4e89-83fe-62b413350946",
4   "name": "cluster test",
5   "language": "EN",
6   "repository": "pg cluster in open shift"
7 }
```

← → ↻ ⚠ Nicht sicher | demo-cap-pg-os-4-git-ha-cap-pg-ha.apps-crc.testing/public/Projects?\$t... ☆ 🔍

```
{"@odata.context": "$metadata#Projects", "value": [{"ID": "82c42c89-182a-4e89-83fe-62b413350946", "name": "cluster test", "language": "EN", "repository": "pg cluster in open shift"}]}
```



CAP ON OPENSSHIFT 4 WITH POSTGRESQL HA

Connect to PgSQL HA Cluster

```
export POSTGRES_PASSWORD=$(oc get secret --namespace cap-pg-ha postgresql-ha-postgresql -o jsonpath="{.data.postgresql-password}" | base64 --decode)

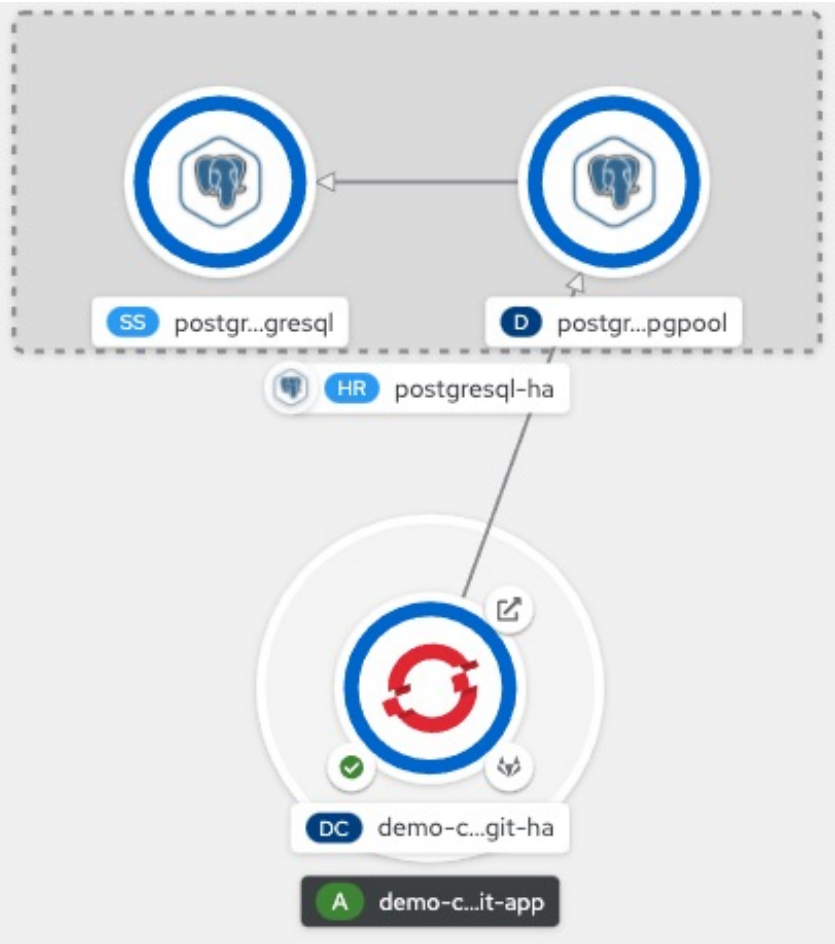
oc run postgresql-ha-client --rm --tty -i --restart='Never' --namespace cap-pg-ha --image docker.io/bitnami/postgresql-repmgr:11.12.0-debian-10-r23 --env="PGPASSWORD=$POSTGRES_PASSWORD" --command -- psql -h postgresql-ha-pgpool -p 5432 -U postgres -d postgres

select * from public.db_projects;
```

```
[postgres=# \dt
      List of relations
 Schema |          Name          | Type  | Owner
-----+-----+-----+-----
 public | databasechangelog      | table | postgres
 public | databasechangelock     | table | postgres
 public | db_projects            | table | postgres
 public | db_votes               | table | postgres
(4 rows)

postgres=# select * from public.db_projects;
      id      | name      | language | repository
-----+-----+-----+-----
 82c42c89-182a-4e89-83fe-62b413350946 | cluster test | EN      | pg cluster in open shift
(1 row)
```

CAP ON OPENSSHIFT 4 WITH POSTGRESQL HA & OIDC



Single values (env) ⓘ

NAME	VALUE
OIDC_DOMAIN	oidc
OIDC_ISSUER	oidc
OIDC_CLIENT_ID	oidc
OIDC_IDENTIFIER	oidc
OIDC_ALGORITHMS	oidc
CDS_REQUIRES_DB_CREDENTIALS_DATABASE	capire
CDS_REQUIRES_DB_CREDENTIALS_HOST	capire
CDS_REQUIRES_DB_CREDENTIALS_PORT	capire
CDS_REQUIRES_DB_CREDENTIALS_USER	capire
CDS_REQUIRES_DB_CREDENTIALS_PASSWORD	postgresql-ha-postgresql postgresql-password



CAP ON OPENSSHIFT 4 WITH POSTGRESQL HA & OIDC

<http://demo-cap-pg-os-4-git-ha-cap-pg-ha.apps-crc.testing/>

Welcome to @sap/cds Server
Serving cap-pg 1.0.0

These are the paths currently served ...

Web Applications:

– none –

Service Endpoints:

/public / \$metadata

- Projects → Fiori preview
- Votes → Fiori preview

test-git-cap-oidc.apps-crc.testing/public/\$metadata

```
Error: Error: No auth token
    at /app/srv/auth.js:72:25
    at allFailed (/app/node_modules/passport/lib/middleware/authenticate.js:107:18)
    at attempt (/app/node_modules/passport/lib/middleware/authenticate.js:180:28)
    at JwtStrategy.strategy.fail (/app/node_modules/passport/lib/middleware/authenticate.js:302:9)
    at JwtStrategy.authenticate (/app/node_modules/passport-jwt/lib/strategy.js:96:21)
    at attempt (/app/node_modules/passport/lib/middleware/authenticate.js:366:16)
    at authenticate (/app/node_modules/passport/lib/middleware/authenticate.js:367:7)
    at module.exports (/app/srv/auth.js:95:7)
    at Layer.handle [as handle_request] (/app/node_modules/express/lib/router/layer.js:95:5)
    at trim_prefix (/app/node_modules/express/lib/router/index.js:317:13)
```

Valid CAP user needed!

```
service PublicService {
  @requires: 'authenticated-user'
  entity Projects as projection on db.Projects;
}
```



CAP ON OPENS SHIFT 4 WITH POSTGRES SQL HA & OIDC

<http://test-git-cap-oidc.apps-crc.testing/webapp/index.html>

de.bit.recap.UI5Auth0

Login

Load CAP view



Welcome

Log in to dev-a2qo7c1e to continue to SPA-App.

Email address

Password

[Forgot password?](#)

Continue

Don't have an account? [Sign up](#)

OR

Continue with Google

Name	Method
login?state=hKFo2SBCa01fSy1ESEdtUFNk...	POST
auth?response_type=code&redirect_uri=htt...	GET
callback?state=E7H3IY78a3Gl_Lqas1mSP...	GET
callback?state=E7H3IY78a3Gl_Lqas1mSP...	GET
resume?state=e-_KzUSvKE10_i7sO7qqfgpj...	GET
index.html?code=ALwGmBg7Drah3-EA&st...	GET
sap-ui-core.js	GET

▼ Query String Parameters view source view URL-encoded

code: ALwGmBg7Drah3-EA

state: RlE3Z2s5S204NnB0Mmdyek0tYXhtLVNNdEZ0Zi4uaU5ZNULvZFPnfm5pQQ==



CAP ON OPENSIFT 4 WITH POSTGRESQL HA & OIDC

<http://test-git-cap-oidc.apps-crc.testing/webapp/index.html>

× Headers Preview Response Initiator Timing Cookies

▼ General

Request URL: <https://dev-a2qo7c1e.eu.auth0.com/oauth/token>

Request Method: POST

Status Code: ● 200

▼ Request Payload [view source](#)

```
{client_id: "c8qFhPSwYRCY0S3KB4R7ljemsfXbplaY",...}
  client_id: "c8qFhPSwYRCY0S3KB4R7ljemsfXbplaY"
  code: "ALwGmBg7Drah3-EA"
  code_verifier: "Tylw32ip3tJE6ZzpeQ5IoV33AloqvcAjitdwkSeZBIk"
  grant_type: "authorization_code"
  redirect_uri: "http://test-git-cap-oidc.apps-crc.testing/webapp/index.html"
```

de.bit.recap.UI5Auth0

 Logout

Load CAP view



CAP ON OPENSSHIFT 4 WITH POSTGRESQL HA & OIDC

<https://jwt.io/>

Encoded PASTE A TOKEN HERE

```
eyJhbGciOiJSUzI1NiIsInR5cCI6IkpXVCIsImt  
pZCI6InIwZkNRWWY0aWppWlFrdEpuZ1RIMyJ9.e  
yJpc3MiOiJodHRwczovL2Rldi1hMnFvN2MxZS5l  
dS5hdXRoMC5jb20vIiwic3ViIjoiz29vZ2x1LW9  
hdXRoMnwxMDM0MDg1OTk0ODQ2MTU1NzI5NDAiLC  
JhdWQiOi0lsiaHR0cHM6Ly9jYXAtYXV0aCIsImh0d  
HBzOi8vZGV2LWEycW83YzFlLmV1LmF1dGgwLmNv  
bS91c2VyaW5mbYJdLCJpYXQiOjE2MjQzNzY2Mjc  
sImV4cCI6MTYyNDQ2MzAyNywiYXpwIjoizYzhxRm  
hQU3dZUkNZT1MzS0I0UjdsamVtc2ZYnBsYVkiL  
CJzY29wZSI6Im9wZW5pZCBwcm9maWxlIGVtYWls  
IHJlYWQ6YWxsIn0.B0l6UgqmbjEg0aVmttsbV0_  
Z_WDL0kIxWlHyWfRJDaw_F-  
4NW18RZo44x8w6ZH2QTFDCQjaU8UPKkGG5Au0Oo  
T2atQtRYn6VoAAC3NXZhPUudydW0bj6-  
wpA_J5tZgyRpS_A2qe0lqsZ9evM2WLJUdDqT9f0  
ViLWNZom-  
HTMPj5_nUMUgghuJTwyQDu3o8n0HAw8YYpJEpsF  
b-7DsoSE7VFN6iNw61le0ZPBc-  
XSNt3EnAmljswUMT6LW0A9fYyrk5G42c_2gW4-  
zESXcB8SLIUxHM6I89WzSrV172MqhZ1j812WdK1  
VuzR8d9ceVMNpfK03suWMCXUgNoXr2V4EiA
```

Decoded EDIT THE PAYLOAD AND SECRET

HEADER: ALGORITHM & TOKEN TYPE

```
{  
  "alg": "RS256",  
  "typ": "JWT",  
  "kid": "r0fCQYf4ijiZQktJngTH3"  
}
```

PAYLOAD: DATA

```
{  
  "iss": "https://dev-a2qo7c1e.eu.auth0.com/",  
  "sub": "google-oauth2|103408599484615572940",  
  "aud": [  
    "https://cap-auth",  
    "https://dev-a2qo7c1e.eu.auth0.com/userinfo"  
  ],  
  "iat": 1624376627,  
  "exp": 1624463027,  
  "azp": "c8qFhPSwYRCY0S3KB4R7ljemsfXbplaY",  
  "scope": "openid profile email read:all"  
}
```

VERIFY SIGNATURE

```
RSASHA256(  
  base64UrlEncode(header) + "." +  
  base64UrlEncode(payload),  
  -----BEGIN PUBLIC KEY-----  
  MIIBIjANBgkqhkiG9w0BAQEFAAOCAQ
```



CAP ON OPENSIFT 4 WITH POSTGRESQL HA & OIDC

<http://test-git-cap-oidc.apps-crc.testing/webapp/index.html>

Access to CAP service with valid OpenID Connect token

Request URL: http://test-git-cap-oidc.apps-crc.testing/public/\$metadata

Request Method: GET

Authorization: Bearer eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCIsImtpZCI6InIwZkNRWY0aWppwLFdi1hMnFvN2MxZS5ldS5hdXRoMC5jb20vIiwic3ViIjoIjZ29vZ2xllW9hdXRoMnwxMDM0MDg1OTk0ODQ2MTUwOaCIsImh0dHBzOi8vZGV2LWEycW83YzFlLmV1LmF1dGgwLmNvbS91c2VyaW5mbYjdlCjYXQiOjE2MjQzNkYzYzhRmQU3dZUkNZT1MzS0I0UjdsamVtc2ZYnBsYVkiLCJzY29wZSI6Im9wZW5pZCBwcm9maWxIGVtYWwluUvIlaDXPeAazkd58oLVtgf6l81mybmCoBsdng-n7GNliaYtpFWExJfU0Di90syype4cLV2XcJyd1MMZFapV

Headers	Preview	Response	Initiator	Timing	Cookies
1	<?xml version="1.0" encoding="utf-8"?>				
2	<edmx:Edmx Version="4.0" xmlns:edmx="http://docs.oasis-open.org/odata/ns/edmx">				
3	<edmx:DataServices>				
4	<Schema Namespace="PublicService" xmlns="http://docs.oasis-open.org/odata/ns/edm">				
5	<EntityContainer Name="EntityContainer">				
6	<EntitySet Name="Projects" EntityType="PublicService.Projects"/>				
7	</EntityContainer>				
8	<EntityType Name="Projects">				
9	<Key>				
10	<PropertyRef Name="ID"/>				
11	</Key>				
12	<Property Name="ID" Type="Edm.Guid" Nullable="false"/>				
13	<Property Name="name" Type="Edm.String" MaxLength="150"/>				
14	<Property Name="language" Type="Edm.String" MaxLength="150"/>				
15	<Property Name="repository" Type="Edm.String"/>				
16	</EntityType>				
17	</Schema>				
18	</edmx:DataServices>				
19	</edmx:Edmx>				

de.bit.recap.UI5Auth0

					+	↺
Project Name	Language	Repository	Created at	Created by		
CAP Demo	EN	ReCAP				



ENTERPRISE READY?

CAP on Kubernetes

- Real multi-cloud: OpenShift on Azure, local DC, AWS.
- Source to Image (S2I)
- Pipelines
- Lift and shift: run complete VMs

Database

- Support for DBaaS: e.g. CosmosDB, or Postgres (see my [tweet](#))

Security

- CAP as API
- SSO Options: OIDC, X.509, SAML 2.0





[@tobiashofmann](#)



[LinkedIn](#)



[GitLab](#)



[GitHub](#)

TOBIAS
HOFMANN

Senior Consultant | Lead Architect SAP



[Personal website](#)



tobias.hofmann@bridging-it.de

