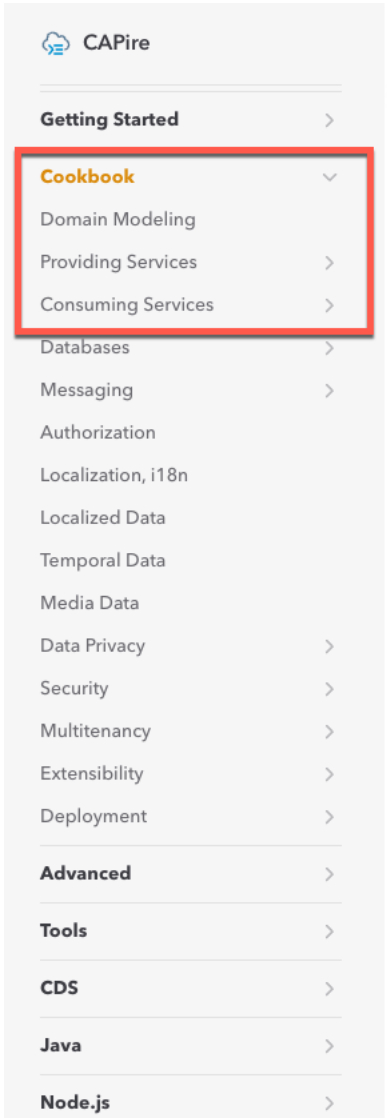
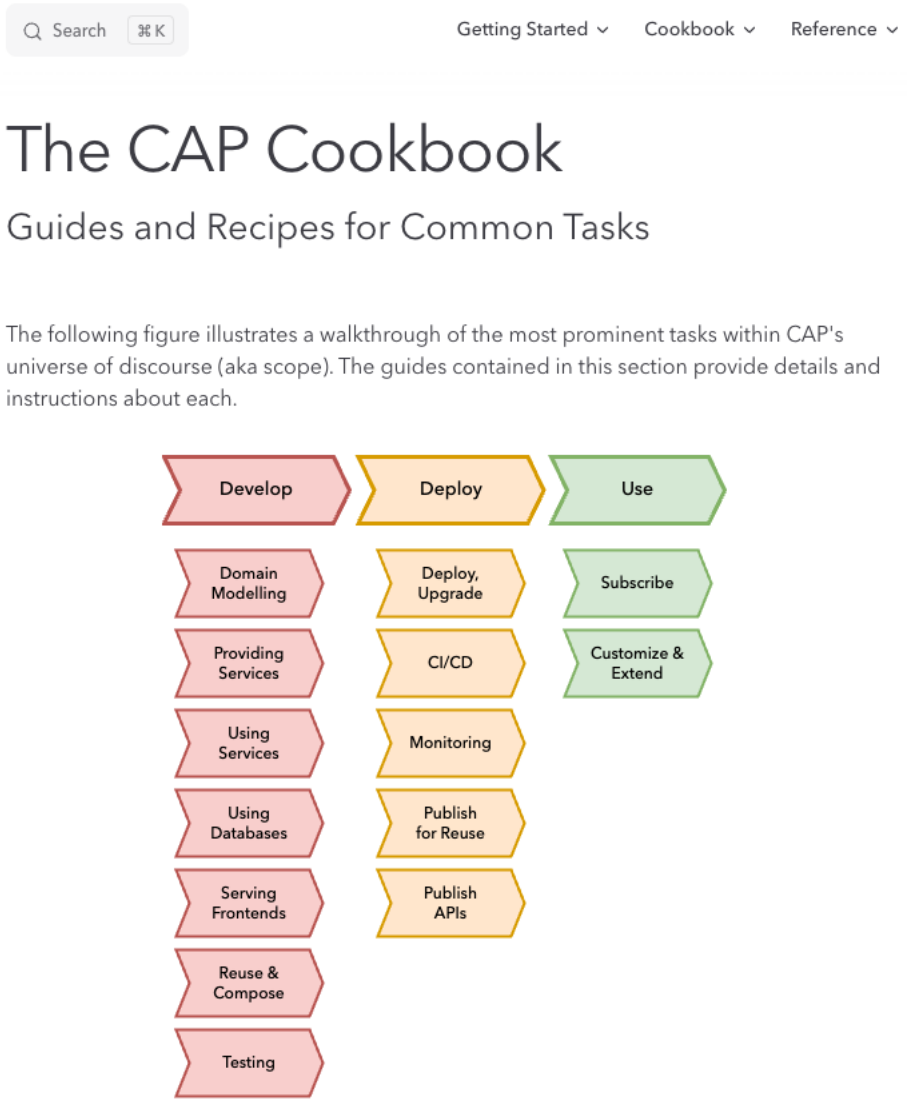
Cookbook 소개


CAPire
Getting Started
Cookbook
Domain Modeling
Providing Services
Consuming Services
Databases
Messaging
Authorization
Localization, i18n
Localized Data
Temporal Data
Media Data
Data Privacy
Security
Multitenancy
Extensibility
Deployment
Advanced
Tools
CDS
Java
Node.js


Search ⌘K
Getting Started
Cookbook
Reference
The CAP Cookbook
Guides and Recipes for Common Tasks
The following figure illustrates a walkthrough of the most prominent tasks within CAP's universe of discourse (aka scope). The guides contained in this section provide details and instructions about each.
Develop
Deploy
Use
Domain Modelling
Providing Services
Using Services
Using Databases
Serving Frontends
Reuse & Compose
Testing
Deploy, Upgrade
CI/CD
Monitoring
Publish for Reuse
Publish APIs
Subscribe
Customize & Extend

Domain Modeling & Providing Service Hands-on

Capture intent → What, not How

```
using { cuid, managed } from '@sap/cds/common’;
entity Books : cuid, managed
{
  title : localized String;
  descr : localized String;
  author : Association to Authors;
}
```

Entity-Relationship Modeling

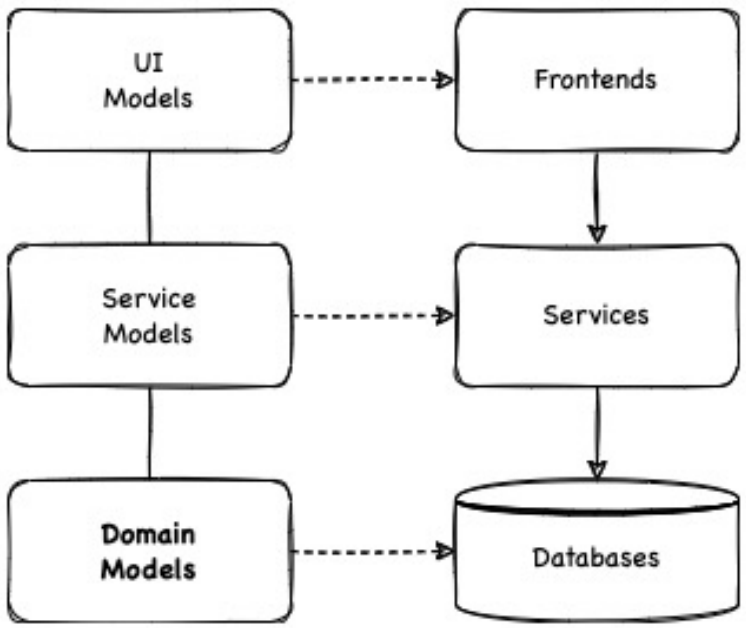
```
using { cuid } from '@sap/cds/common';

entity Books : cuid {
  title  : String;
  descr  : String;
  genre  : Genre;
  author : Association to Authors;
}

entity Authors : cuid {
  name   : String;
  books  : Association to many Books on books.author = $self;
}

type Genre : String enum {
  Mystery; Fiction; Drama;
}
```

Fueling Generic Providers, Domain-Driven Design Keep it Simple, Stupid Aspect-oriented Modeling



CAP shares these goals and approaches with Domain-driven Design:

- Placing projects' primary focus on the core domain
- Close collaboration of developers and domain experts
- Iteratively refining domain knowledge

CAP Composition of many

CDS – Data modeling

Schema.cds

```
entity XTBL003 {
       key FLOWUUID : UUID @(Core.Computed : true);
       key FLOWCODE : String(5);
       key STATUS : String(2);
       DETAIL : Composition of many XTBL004
         on DETAIL.FLOWUUID = FLOWUUID
         and DETAIL.FLOWCODE = FLOWCODE
         and DETAIL.STATUS = STATUS;
       FIELD01 : String(100);
       FIELD02 : String(100);
}

entity XTBL004 {
       key FLOWUUID : UUID @(Core.Computed : true);
       key FLOWCODE : String(5);
       key NO1 : String(2);
       STATUS : String(2);
       APPROVAL_NAME : String(100);
}
```

Postman - POST

/hkmc/XTBL003


```
{
 "FLOWCODE": "MH001",
 "STATUS": "01",
 "FIELD01": "07199782",
 "FIELD02": "K0121120",
 "DETAIL" : [
  {
   "FLOWCODE": "MH001",
   "NO1": "2",
   "STATUS": "01",
   "APPROVAL_NAME": "JHAN"
  },{
   "FLOWCODE": "MH001",
   "NO1": "3",
   "STATUS": "01",
   "APPROVAL_NAME": "JHAN"
  }]
}
```


Postman – GET

/hkmc/XTBL003(FLOWUUID=50a79a4e-834f-4294-9d4c-c415fbc572c4,FLOWCODE='MH001',STATUS='01')?$expand=DETAIL


```
{
 "@odata.context": "$metadata#XTBL003(DETAIL())/$entity",
 "FLOWUUID": "79e8d3e8-ef0d-401f-a586-a15f59789b62",
 "FLOWCODE": "MH001",
 "STATUS": "01",
 "DETAIL": [
  {
   "FLOWUUID": "79e8d3e8-ef0d-401f-a586-a15f59789b62",
   "FLOWCODE": "MH001",
   "NO1": "2",
   "STATUS": "01",
   "APPROVAL_NAME": "JHAN"
  },
  {
   "FLOWUUID": "79e8d3e8-ef0d-401f-a586-a15f59789b62",
   "FLOWCODE": "MH001",
   "NO1": "3",
   "STATUS": "01",
   "APPROVAL_NAME": "JHAN"
  }
 ],
 "FIELD01": "07199782",
 "FIELD02": "K0121120"
}
```

CAP Composition

OData Metadata

```
<EntityType Name="XTBL003">
 <Key>
  <PropertyRef Name="FLOWUUID"/>
  <PropertyRef Name="FLOWCODE"/>
  <PropertyRef Name="STATUS"/>
 </Key>
 <Property Name="FLOWUUID" Type="Edm.Guid" Nullable="false"/>
 <Property Name="FLOWCODE" Type="Edm.String" MaxLength="5" Nullable="false"/>
 <Property Name="STATUS" Type="Edm.String" MaxLength="2" Nullable="false"/>
 <NavigationProperty Name="DETAIL" Type="Collection(hkmc.XTBL004)">
   <OnDelete Action="Cascade"/>
 </NavigationProperty>
 <Property Name="FIELD01" Type="Edm.String" MaxLength="100"/>
 <Property Name="FIELD02" Type="Edm.String" MaxLength="100"/>
</EntityType>

<EntityType Name="XTBL004">
 <Key>
   <PropertyRef Name="FLOWUUID"/>
   <PropertyRef Name="FLOWCODE"/>
   <PropertyRef Name="NO1"/>
 </Key>
 <Property Name="FLOWUUID" Type="Edm.Guid" Nullable="false"/>
 <Property Name="FLOWCODE" Type="Edm.String" MaxLength="5" Nullable="false"/>
 <Property Name="NO1" Type="Edm.String" MaxLength="2" Nullable="false"/>
 <Property Name="STATUS" Type="Edm.String" MaxLength="2"/>
 <Property Name="APPROVAL_NAME" Type="Edm.String" MaxLength="100"/>
</EntityType>
```

Postman – PUT

/hkmc/XTBL003(FLOWUUID=79e8d3e8-ef0d-401f-a586-a15f59789b62,FLOWCODE='MH001',STATUS='01')


```
{
"FLOWUUID": "79e8d3e8-ef0d-401f-a586-a15f59789b62",
"FLOWCODE": "MH001",
"STATUS": "01",
"FIELD01": "VVV",
"FIELD02": "SSS",
"DETAIL" : [{
"FLOWUUID": "79e8d3e8-ef0d-401f-a586-a15f59789b62",
"FLOWCODE": "MH001",
"NO1": "2",
"STATUS": "01",
"APPROVAL_NAME": "JHANS"
},{
"FLOWUUID": "79e8d3e8-ef0d-401f-a586-a15f59789b62",
"FLOWCODE": "MH001",
"NO1": "3",
"STATUS": "01",
"APPROVAL_NAME": "JHANQ"
}]
}
```


Postman – DELETE

/hkmc/XTBL003(FLOWUUID=79e8d3e8-ef0d-401f-a586-a15f59789b62,FLOWCODE='MH001',STATUS='01')

CAP Composition of one

CDS – Data modeling

Schema.cds

```
entity XTBL0030 {
       key FLOWUUID : UUID @(Core.Computed : true);
       key FLOWCODE : String(5);
       key STATUS : String(2);
       DETAIL : Composition of one XTBL004
         on DETAIL.FLOWUUID = FLOWUUID
         and DETAIL.FLOWCODE = FLOWCODE
         and DETAIL.STATUS = STATUS;
       FIELD01 : String(100);
       FIELD02 : String(100);
}

entity XTBL0040 {
       key FLOWUUID : UUID @(Core.Computed : true);
       key FLOWCODE : String(5);
       key NO1 : String(2);
       STATUS : String(2);
       APPROVAL_NAME : String(100);
}
```

Postman - POST

/hkmc/XTBL0030

```
{
 "FLOWCODE": "MH001",
 "STATUS": "01",
 "FIELD01": "07199782",
 "FIELD02": "K0121120",
 "DETAIL" :
  {
   "FLOWCODE": "MH001",
   "NO1": "2",
   "STATUS": "01",
   "APPROVAL_NAME": "JHAN"
  }
}
```

Postman – GET

XTBL0030(FLOWUUID=3f96b199-6f98-4160-844d-06795360e697,FLOWCODE='MH001',STATUS='01')?$expand=DETAIL


```
{
"@odata.context": "$metadata#XTBL0030(DETAIL())/$entity",
"FLOWUUID": "3f96b199-6f98-4160-844d-06795360e697",
"FLOWCODE": "MH001",
"STATUS": "01",
"FIELD01": "07199782",
"FIELD02": "K0121120",
"DETAIL": {
 "FLOWUUID": "3f96b199-6f98-4160-844d-06795360e697",
 "FLOWCODE": "MH001",
 "NO1": "2",
 "STATUS": "01",
 "APPROVAL_NAME": "JHAN"
 }
}
```

CAP Composition

```
<EntityType Name="XTBL0030">
<Key>
<PropertyRef Name="FLOWUUID"/>
<PropertyRef Name="FLOWCODE"/>
<PropertyRef Name="STATUS"/>
</Key>
<Property Name="FLOWUUID" Type="Edm.Guid" Nullable="false"/>
<Property Name="FLOWCODE" Type="Edm.String" MaxLength="5" Nullable="false"/>
<Property Name="STATUS" Type="Edm.String" MaxLength="2" Nullable="false"/>
<NavigationProperty Name="DETAIL" Type="hkmc.XTBL0040">
  <OnDelete Action="Cascade"/>
  <ReferentialConstraint Property="FLOWUUID" ReferencedProperty="FLOWUUID"/>
  <ReferentialConstraint Property="FLOWCODE" ReferencedProperty="FLOWCODE"/>
  <ReferentialConstraint Property="STATUS" ReferencedProperty="STATUS"/>
</NavigationProperty>
<Property Name="FIELD01" Type="Edm.String" MaxLength="100"/>
<Property Name="FIELD02" Type="Edm.String" MaxLength="100"/>
</EntityType>
<EntityType Name="XTBL0040">
<Key>
<PropertyRef Name="FLOWUUID"/>
<PropertyRef Name="FLOWCODE"/>
<PropertyRef Name="NO1"/>
</Key>
<Property Name="FLOWUUID" Type="Edm.Guid" Nullable="false"/>
<Property Name="FLOWCODE" Type="Edm.String" MaxLength="5" Nullable="false"/>
<Property Name="NO1" Type="Edm.String" MaxLength="2" Nullable="false"/>
<Property Name="STATUS" Type="Edm.String" MaxLength="2"/>
<Property Name="APPROVAL_NAME" Type="Edm.String" MaxLength="100"/>
</EntityType>
```

CAP Association(to one)

CDS – Data modeling

Schema.cds

```
entity XYTBL003 {
key FLOWUUID : UUID @(Core.Computed : true);
key FLOWCODE : String(5);
key STATUS : String(2);
DETAIL : Association to XYTBL004
  on DETAIL.FLOWUUID = FLOWUUID
  and DETAIL.FLOWCODE = FLOWCODE
  and DETAIL.STATUS = STATUS;
FIELD01 : String(100);
FIELD02 : String(100);
}

entity XYTBL004 {
key FLOWUUID : UUID @(Core.Computed : true);
key FLOWCODE : String(5);
key NO1 : String(2);
STATUS : String(2);
APPROVAL_NAME : String(100);
}
```

Postman - POST

/hkmc/XYTBL003

```
{
"FLOWUUID": "170e3c5e-d3cf-4f0a-97ee-
bf68f971d8d9",
"FLOWCODE": "MH001",
"STATUS": "02",
"FIELD01": "07199782",
"FIELD02": "K0121120"
}
```

/hkmc/XYTBL004

```
{
"FLOWUUID": "170e3c5e-d3cf-4f0a-97ee-bf68f971d8d9",
"FLOWCODE": "MH001",
"NO1": "2",
"STATUS": "02",
"APPROVAL_NAME": "JHAN"
}
```

```
{
"FLOWUUID": "170e3c5e-d3cf-4f0a-97ee-bf68f971d8d9",
"FLOWCODE": "MH001",
"NO1": "2",
"STATUS": "02",
"APPROVAL_NAME": "JHAN9"
}
```

Postman – GET

/hkmc/XYTBL003(FLOWUUID= 170e3c5e-d3cf-4f0a-97ee-bf68f971d8d9,FLOWCODE='MH001',STATUS='02')?$expand=DETAIL

```
{
"@odata.context": "$metadata#XYTBL003(DETAIL())/$entity",
"FLOWUUID": "170e3c5e-d3cf-4f0a-97ee-bf68f971d8d9",
"FLOWCODE": "MH001",
"STATUS": "02",
"FIELD01": "07199782",
"FIELD02": "K0121120",
"DETAIL": {
 "FLOWUUID": "170e3c5e-d3cf-4f0a-97ee-bf68f971d8d9",
 "FLOWCODE": "MH001",
 "NO1": "2",
 "STATUS": "02",
 "APPROVAL_NAME": "JHAN"
}
}
```

Detail이 한건만 나옴

CAP Association to many

CDS – Data modeling

Schema.cds

```
entity XYZTBL003 {
key FLOWUUID : UUID @(Core.Computed : true);
key FLOWCODE : String(5);
key STATUS : String(2);
DETAIL : Association to many XYTBL004
  on DETAIL.FLOWUUID = FLOWUUID
  and DETAIL.FLOWCODE = FLOWCODE
  and DETAIL.STATUS = STATUS;
FIELD01 : String(100);
FIELD02 : String(100);
}

entity XYZTBL004 {
key FLOWUUID : UUID @(Core.Computed : true);
key FLOWCODE : String(5);
key NO1 : String(2);
STATUS : String(2);
APPROVAL_NAME : String(100);
}
```

Postman - POST

/hkmc/XYZTBL003

```
{
"FLOWUUID": "170e3c5e-d3cf-4f0a-97ee-
bf68f971d8d9",
"FLOWCODE": "MH001",
"STATUS": "02",
"FIELD01": "07199782",
"FIELD02": "K0121120"
}
```

/hkmc/XYZTBL004

```
{
"FLOWUUID": "170e3c5e-d3cf-4f0a-97ee-bf68f971d8d9",
"FLOWCODE": "MH001",
"NO1": "2",
"STATUS": "02",
"APPROVAL_NAME": "JHAN"
}
```

```
{
"FLOWUUID": "170e3c5e-d3cf-4f0a-97ee-bf68f971d8d9",
"FLOWCODE": "MH001",
"NO1": "2",
"STATUS": "02",
"APPROVAL_NAME": "JHAN9"
}
```

Postman – GET

/hkmc/XYZTBL003(FLOWUUID= 170e3c5e-d3cf-4f0a-97ee-bf68f971d8d9,FLOWCODE='MH001',STATUS='02')?$expand=DETAIL

```
{
"@odata.context": "$metadata#XYZTBL003(DETAIL())/$entity",
"FLOWUUID": "170e3c5e-d3cf-4f0a-97ee-bf68f971d8d9",
"FLOWCODE": "MH001",
"STATUS": "02",
"FIELD01": "07199782",
"FIELD02": "K0121120",
"DETAIL": [
 {
  "FLOWUUID": "170e3c5e-d3cf-4f0a-97ee-bf68f971d8d9",
  "FLOWCODE": "MH001",
  "NO1": "3",
  "STATUS": "02",
   "APPROVAL_NAME": "JHAN9"
 },
 {
  "FLOWUUID": "170e3c5e-d3cf-4f0a-97ee-bf68f971d8d9",
  "FLOWCODE": "MH001",
  "NO1": "4",
  "STATUS": "02",
  "APPROVAL_NAME": "JHAN90"
 }
]
}
```

Detail이 N건이 나옴

CAP Association

```
<EntityType Name="XYTBL003">
<Key>
<PropertyRef Name="FLOWUUID"/>
<PropertyRef Name="FLOWCODE"/>
<PropertyRef Name="STATUS"/>
</Key>
<Property Name="FLOWUUID" Type="Edm.Guid" Nullable="false"/>
<Property Name="FLOWCODE" Type="Edm.String" MaxLength="5" Nullable="false"/>
<Property Name="STATUS" Type="Edm.String" MaxLength="2" Nullable="false"/>
<NavigationProperty Name="DETAIL" Type="hkmc.XYTBL004">
<ReferentialConstraint Property="FLOWUUID" ReferencedProperty="FLOWUUID"/>
<ReferentialConstraint Property="FLOWCODE" ReferencedProperty="FLOWCODE"/>
</NavigationProperty>
<Property Name="FIELD01" Type="Edm.String" MaxLength="100"/>
<Property Name="FIELD02" Type="Edm.String" MaxLength="100"/>
</EntityType>

<EntityType Name="XYTBL004">
<Key>
<PropertyRef Name="FLOWUUID"/>
<PropertyRef Name="FLOWCODE"/>
<PropertyRef Name="NO1"/>
</Key>
<Property Name="FLOWUUID" Type="Edm.Guid" Nullable="false"/>
<Property Name="FLOWCODE" Type="Edm.String" MaxLength="5" Nullable="false"/>
<Property Name="NO1" Type="Edm.String" MaxLength="2" Nullable="false"/>
<Property Name="STATUS" Type="Edm.String" MaxLength="2"/>
<Property Name="APPROVAL_NAME" Type="Edm.String" MaxLength="100"/>
</EntityType>
```

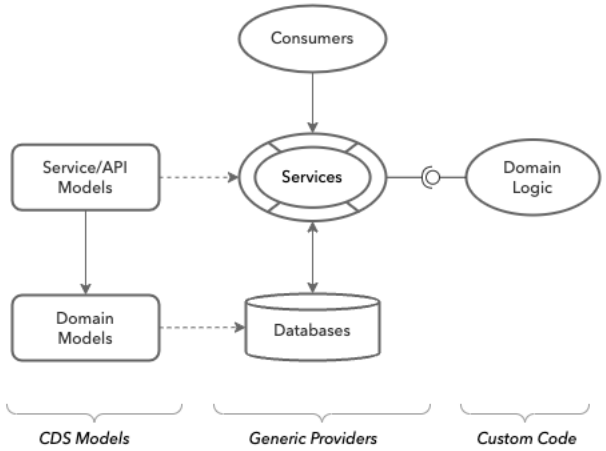
```
<EntityType Name="XYZTBL003">
<Key>
<PropertyRef Name="FLOWUUID"/>
<PropertyRef Name="FLOWCODE"/>
<PropertyRef Name="STATUS"/>
</Key>
<Property Name="FLOWUUID" Type="Edm.Guid" Nullable="false"/>
<Property Name="FLOWCODE" Type="Edm.String" MaxLength="5" Nullable="false"/>
<Property Name="STATUS" Type="Edm.String" MaxLength="2" Nullable="false"/>
<NavigationProperty Name="DETAIL" Type="Collection(hkmc.XYZTBL004)"/>
<Property Name="FIELD01" Type="Edm.String" MaxLength="100"/>
<Property Name="FIELD02" Type="Edm.String" MaxLength="100"/>
</EntityType>

<EntityType Name="XYZTBL004">
<Key>
<PropertyRef Name="FLOWUUID"/>
<PropertyRef Name="FLOWCODE"/>
<PropertyRef Name="NO1"/>
</Key>
<Property Name="FLOWUUID" Type="Edm.Guid" Nullable="false"/>
<Property Name="FLOWCODE" Type="Edm.String" MaxLength="5" Nullable="false"/>
<Property Name="NO1" Type="Edm.String" MaxLength="2" Nullable="false"/>
<Property Name="STATUS" Type="Edm.String" MaxLength="2"/>
<Property Name="APPROVAL_NAME" Type="Edm.String" MaxLength="100"/>
</EntityType>
```

Providing Services

Service-Centric Paradigm

A CAP application commonly provides services defined in CDS models and served by the CAP runtimes. Every active thing in CAP is a service. They embody the behavioral aspects of a domain in terms of exposed entities, actions, and events.

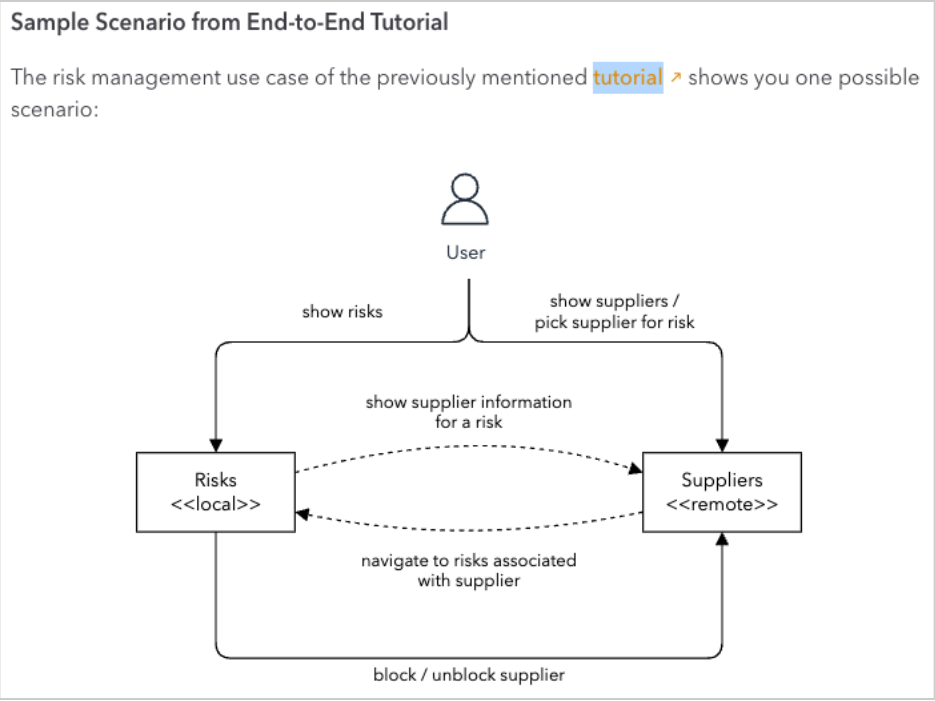


Ubiquitous Events

At runtime, everything happening is in response to events. CAP features a ubiquitous notion of events, which represent both, *requests* coming in through **synchronous** APIs, as well as **asynchronous** *event messages*, blurring the line between both worlds.

Modeling Services	Services Act as Facades
Generic Providers	Serve most requests automatically with OOTB solutions
Pagination & Sorting	Implicit Pagination & Reliable Pagination - @cds.query.xxxx
Input Validation	Implicit Pagination & Reliable Pagination - @mandatory, @assert
Managed Data	Using {managed} from '@sap/cds/common' $user, $now
Concurrency Control	Optimistic concurrency Etags - @odata.etag
Adding Custom Logic	Optimistic concurrency Etags - @odata.etag
Custom Actions & Functions	Action(input params) returns value Function (input params) returns value

Consuming Services



Import APIs	cds import <input file> --as cds Srv/external
Mocking	Srv/external/data cds watch
Querying	Cds.connect.to('cds_dest_in_pkg.json')
Mashups	Cds association between local and re /service/localsrv?$expand=remotesrv
Deployments	Create destination Dest_name in package.json

Databases

Messaging

Authorization

Localization, i18n

Temporal Data

Media Data

Security

Multitenancy

Extensibility

Deployment

CDS

Definition Language(CDL)

Define entity, define type…

Schema Notation(CSN)

…

Query Language(CQL)

CQL is based on standard SQL

```
SELECT from Authors { name, address.street }
SELECT from Books { *, author.name as author }
```

Query Notation(CQN)

CQN is a canonical plain object representation of CDS queries.

```
// Parsing CQL
let query = cds.parse.cql (`SELECT from Foo`)

// Query building
let query = SELECT.from('Foo’)

// Constructing CQN objects in your code
let query = {SELECT:{from:[{ref:['Foo']}]}}

cds.run (query)
```

Binary type

```
entity HR0010 {
key FLOWUUID : UUID @(Core.Computed : true);
key FLOWCODE : String(5);
key DOCUMENTID : String(36);
FILENAME : String(100);
FILESIZE : String(100);
FILEUPLOADDATE: Date;
FILEUPLOADTIME: Time;
URL: String(500);
@Core.MediaType: mimeType
CONTENT: LargeBinary @stream;
@Core.IsMediaType : true
mimeType : String(200);
}
```