

# The Universe of BTP in a Nutshell

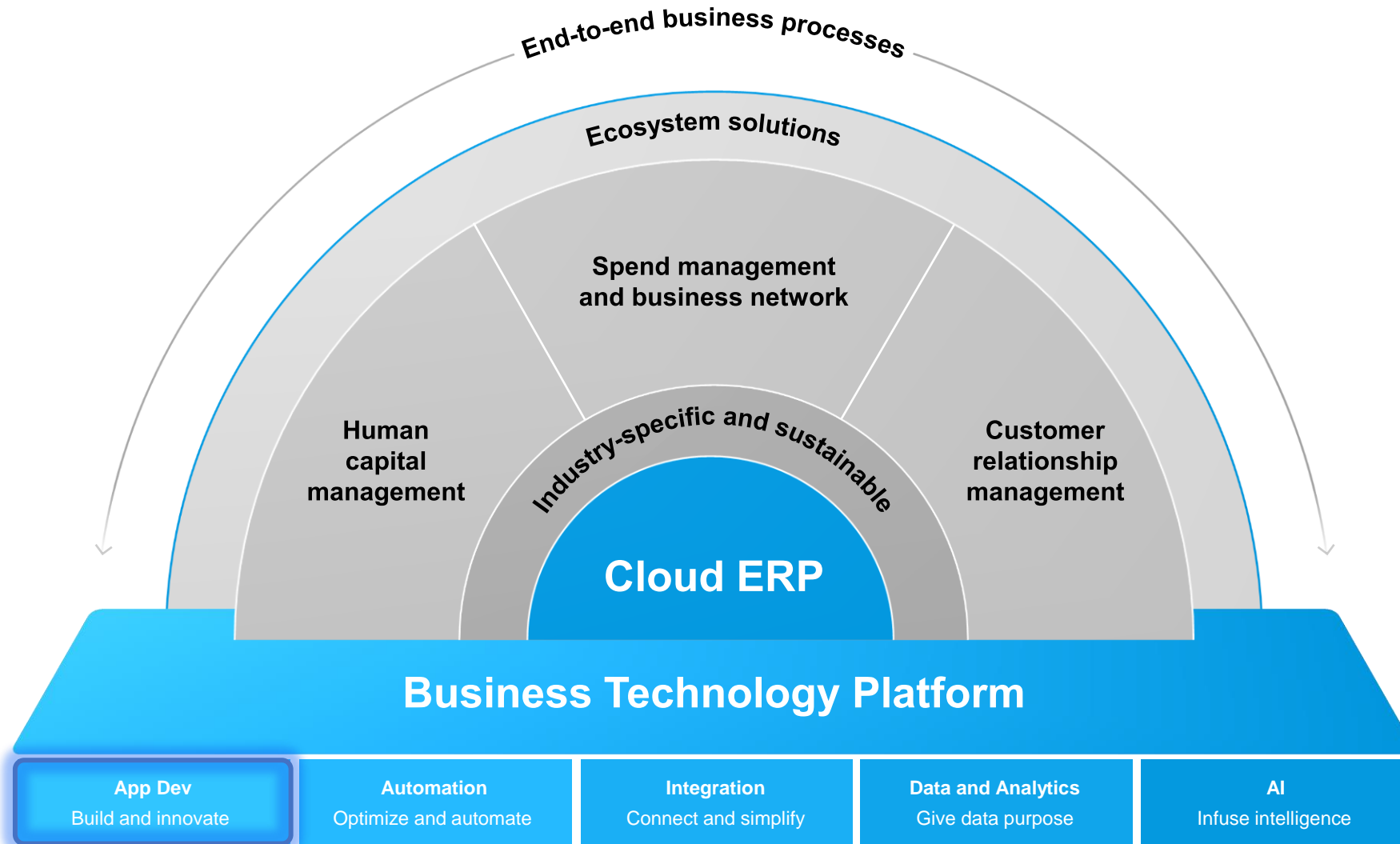
## Pro Code based Development with SAP BTP

Tim Back, Chief Development Architect, Application Development, Automation and Integration, SAP BTP  
02.05.2023

INTERNAL

# **Business Technology Platform Overview**

# The SAP Business Technology Platform makes our integrated modular suite fit for the uniqueness of any organization,,



# SAP BTP products can support your journey



## SAP BUILD

SAP Build Apps

SAP Build Process Automation

SAP Build Work Zone



## PRO-DEVELOPER TOOLS

SAP Business Application  
Studio

SAPUI5 / Fiori elements

SAP Cloud Application Programming Model  
*Java, JS*

ABAP RESTful Application Programming Model  
*ABAP*

SAP-Optimized Frameworks  
& Services



## SAP INTEGRATION SUITE

API & Event Integration  
(Cloud & Hybrid)

API Management

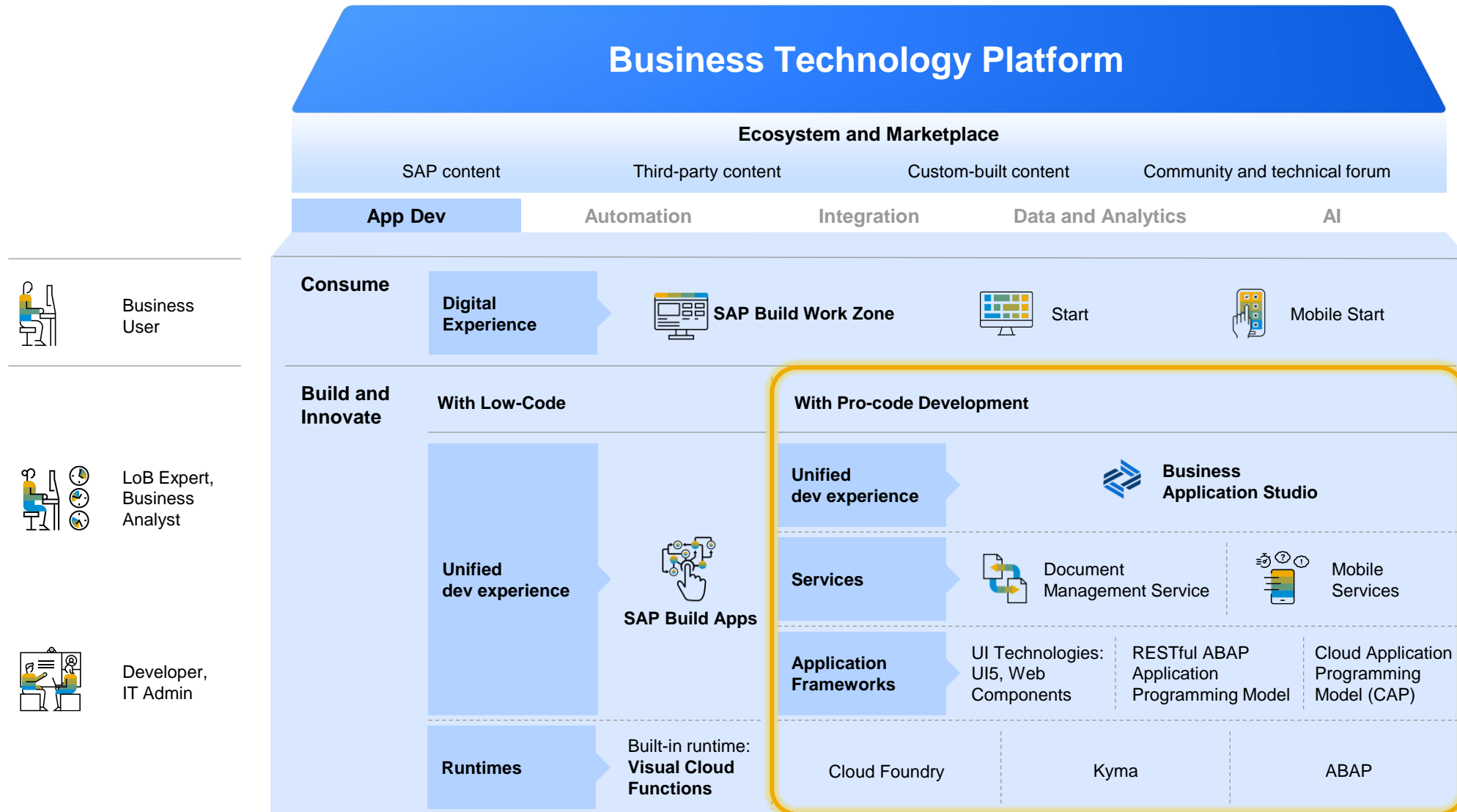
Integration Advisor

Trading Partner Management

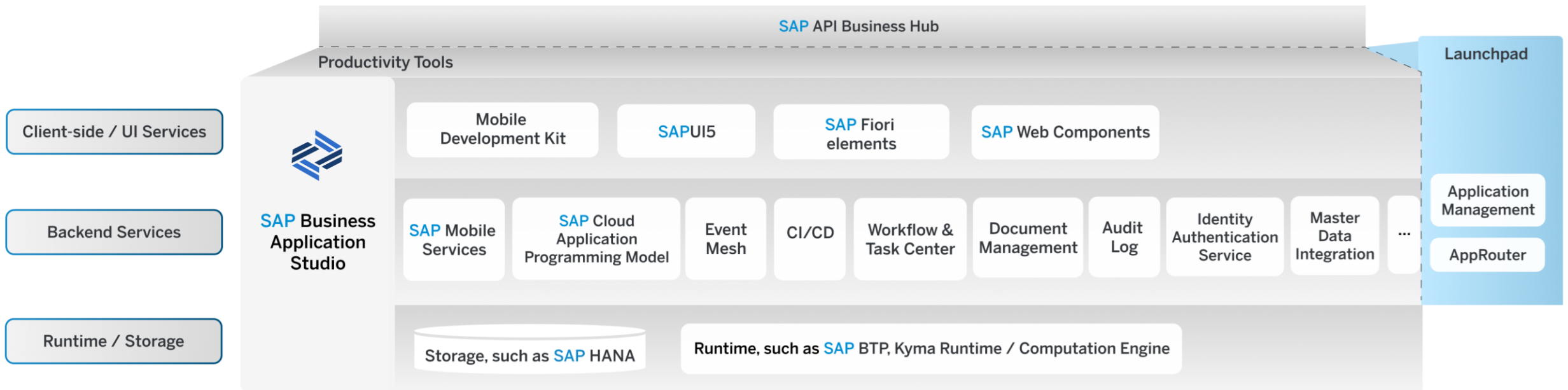
Pre-built content: API Business Hub

SAP Business Technology Platform

# SAP BTP for Application Development

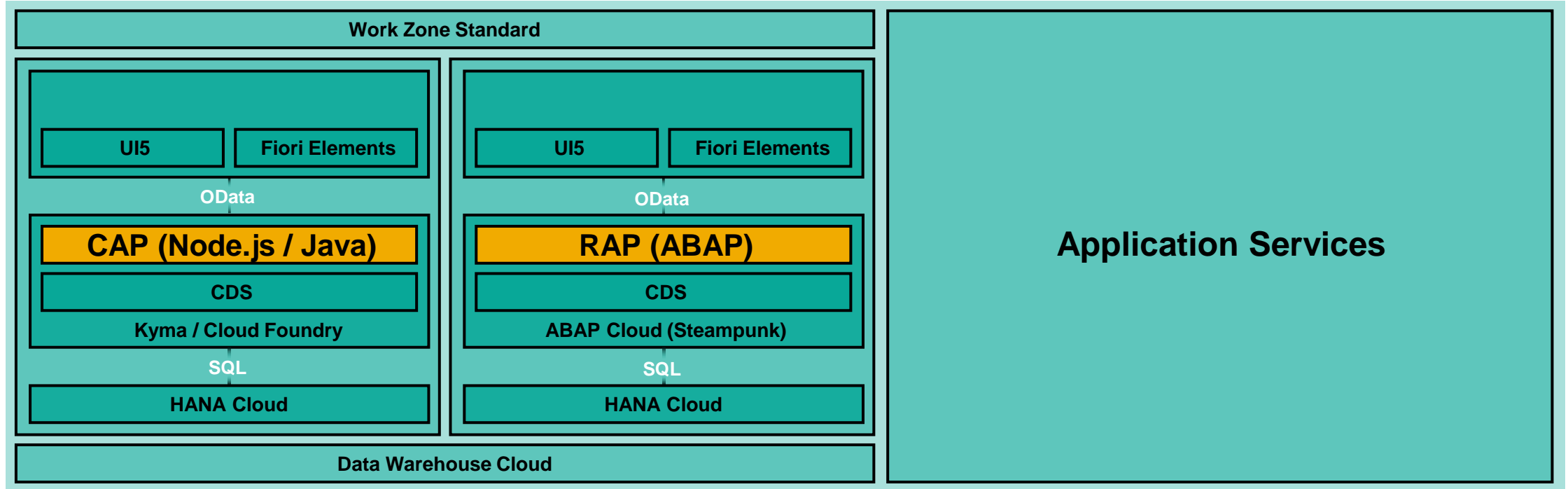


# Integrated Tools and Frameworks for full-stack App Development on BTP



- **SAP Business Application Studio** is the preferred development environment to maximize productivity on BTP for applications and extensions
- **SAP Cloud Application Programming Model** provides domain-driven modelling powered by CDS
- **SAP Mobile Services** can also be used for native development in XCode and Android Studio with SAP BTP SDKs for iOS and Android

# Building Blocks of BTP Applications



**SAP Standard Technologies**

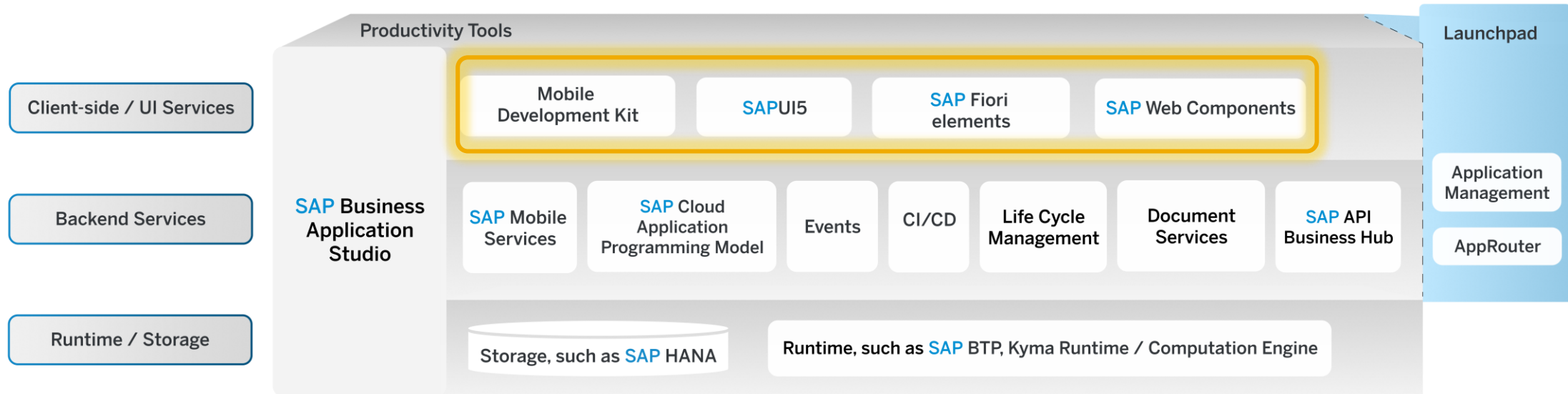
**Enterprise Qualities and Standards**

# **Client Side / UI Services**

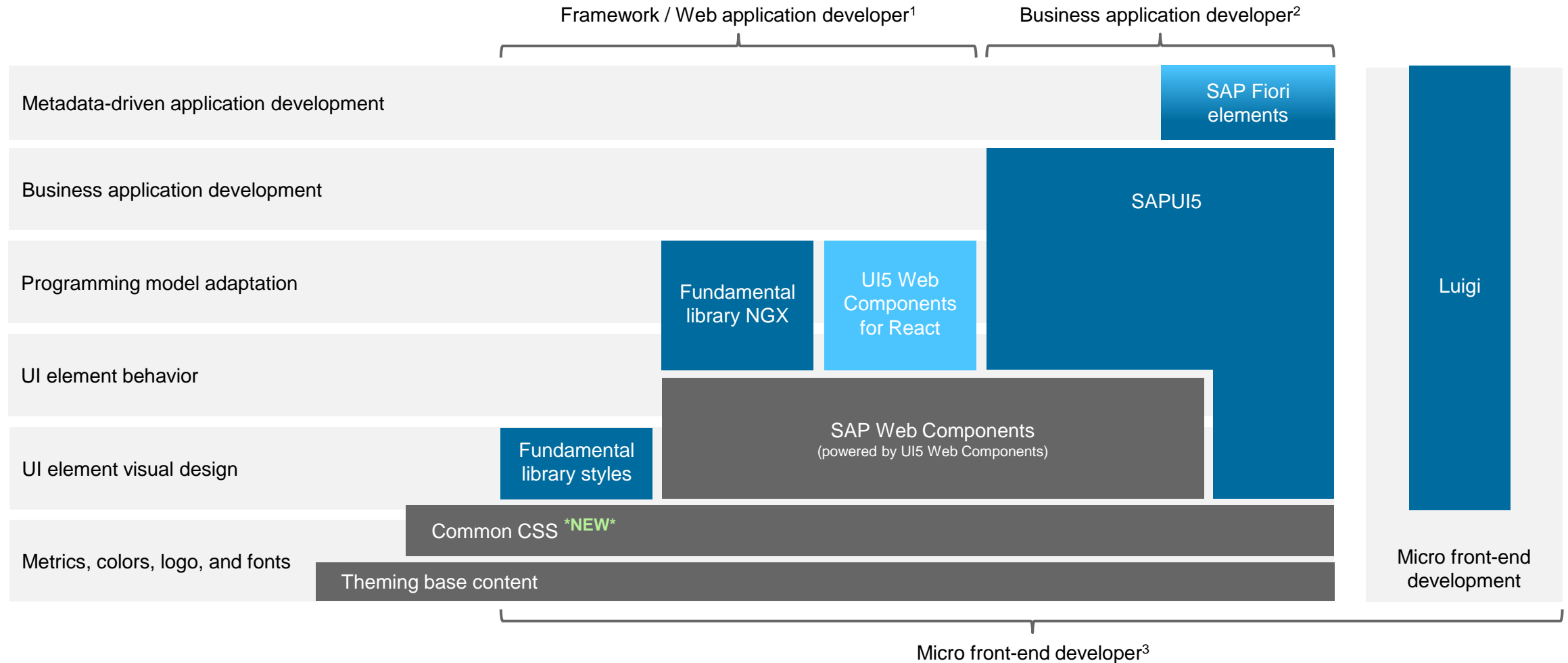


# Client Side / UI Services

Quickly build enterprise-ready applications based on opinionated technologies from SAP



# A tailored pro-code approach for everybody!



- UI technology option from SAP
- Reuse UI technologies
- Inner source UI technology options from SAP

1. Works with HTML/CSS, using component tool kits, free framework choice, special requirements, limited lifetime of apps
2. Works with UI components, using app framework, built-in connectivity, preconfigured UI flexibility, fulfill enterprise qualities
3. Works with multiple UI components, built in different technologies and put together in a micro front-end with Luigi

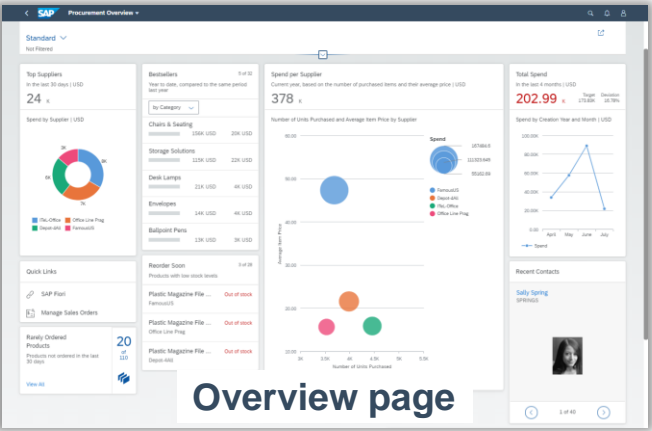
# **Client Side / UI Services**

## **SAP Fiori elements**

# SAP Fiori elements ensures consistent layout, navigation

Most use cases in the enterprise space involve providing an **overview**, **lists** of business objects and management of these business **objects**.

Provide **overview** on business relevant data



Overview page

Work on a **list** of business objects

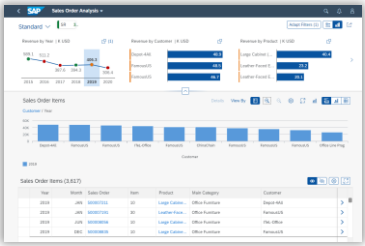
The screenshot shows the 'SAP Manage Product Catalog' Fiori application. It features a top navigation bar with 'Standard' and 'New Notes' tabs. The main content area is a list of products with columns for 'Product', 'Name', 'Category', 'Supplier', 'Star Ratings', and 'Price'. The list is filtered by 'All' and 'All' categories. The bottom of the page shows a 'SAP Fiori' logo and a 'Manage Product Catalog' button.

List report page

Manage individual business **objects**

The screenshot shows the 'SAP Product' Fiori application. It features a top navigation bar with 'Standard' and 'New Notes' tabs. The main content area is a detailed view of a product, 'Bookmark Sticky Notes - assorted'. It includes sections for 'General Information', 'Reviews', and 'Technical Data'. The 'General Information' section shows the product name, category, and price. The 'Reviews' section shows a list of reviews with star ratings and text. The 'Technical Data' section shows the product's dimensions and weight.

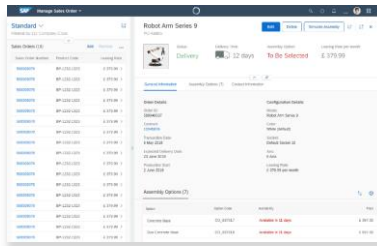
Object page



Analytical list page

The screenshot shows the 'SAP My Worklist' Fiori application. It features a top navigation bar with 'Standard' and 'New Notes' tabs. The main content area is a list of worklist items with columns for 'Worklist Item', 'Worklist Item Description', 'Worklist Item Status', and 'Worklist Item Date'. The list is filtered by 'All' and 'All' categories. The bottom of the page shows a 'SAP Fiori' logo and a 'My Worklist' button.

Worklist page



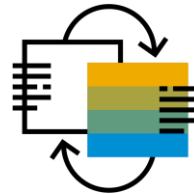
Flexible Column Layout

# SAP Fiori elements boosts SAPUI5 development efficiency



## Developer Productivity

- Focus on **business logic** and back-end services
- Write **less UI code**
- **Reduce** development and maintenance **costs**



## UX Consistency

- Comply with the **latest SAP Fiori design** specification
- Deliver a **centrally managed** user experience
- Include **uniform** fonts, colors, layout, navigation, actions, search, filtering, and more



## Enterprise Readiness

- Deliver **high quality** SAPUI5 applications to end-users
- Ensure **stable, optimized** UI code out of the box
- Include **standard enterprise features** (accessibility, mobile, translation support, ...)

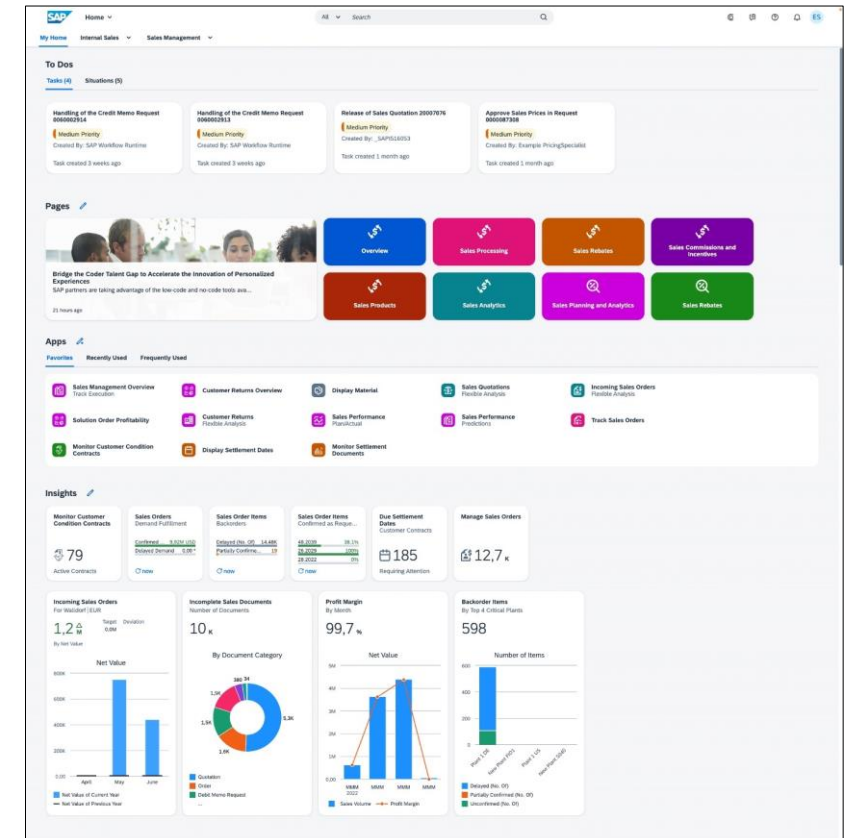
# **Client Side / UI Services**

## **SAPUI5 and SAP Web Components**

# Building modern Web applications with SAPUI5 and Web components

SAPUI5 is an HTML5 framework for creating cross-platform, enterprise-grade applications

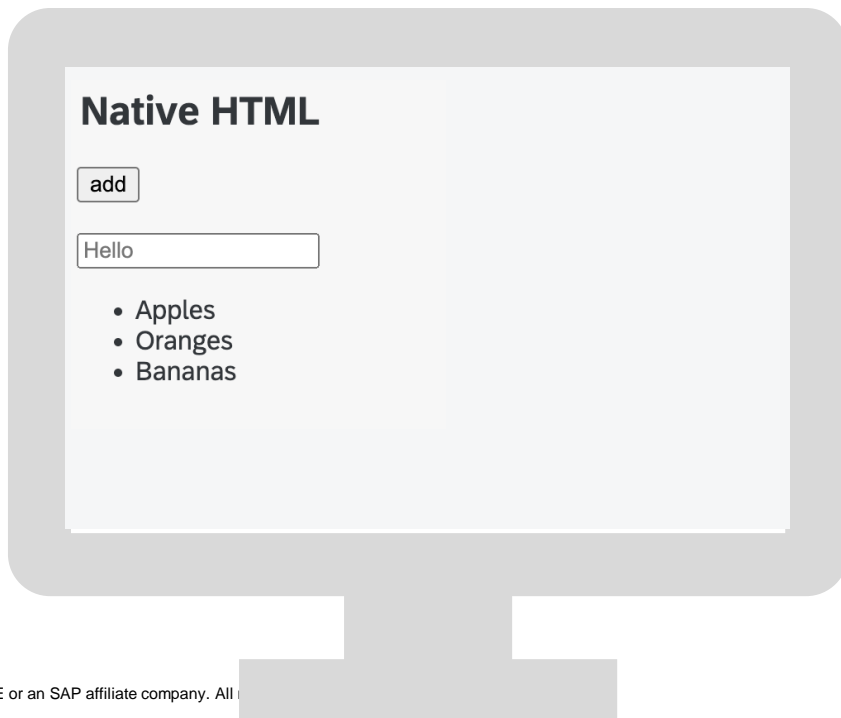
- **Consistent UX:** SAPUI5 enables the SAP Fiori design system evolution across SAP solutions (and beyond).
- **Anywhere use:** SAPUI5 allows a single, responsive app implementation for all browsers, platforms, and devices.
- **Hundreds of enterprise UI elements** to build professional UIs: UI5 Web Components extend UI5 to all Web technology stacks.
- **Powerful extension options:** You can adapt SAP standard apps and customize UIs without coding.
- **Flexible tools for any developer:** SAPUI5 comes with tools to efficiently build, test, and deploy apps, including SAP Business Application Studio, which is our recommended IDE.
- **Innovations:** SAPUI5 brings a separation of apps and framework to centrally innovate while staying upgrade compatible.
- **Open source:** SAPUI5 qualities are available as OpenUI5.



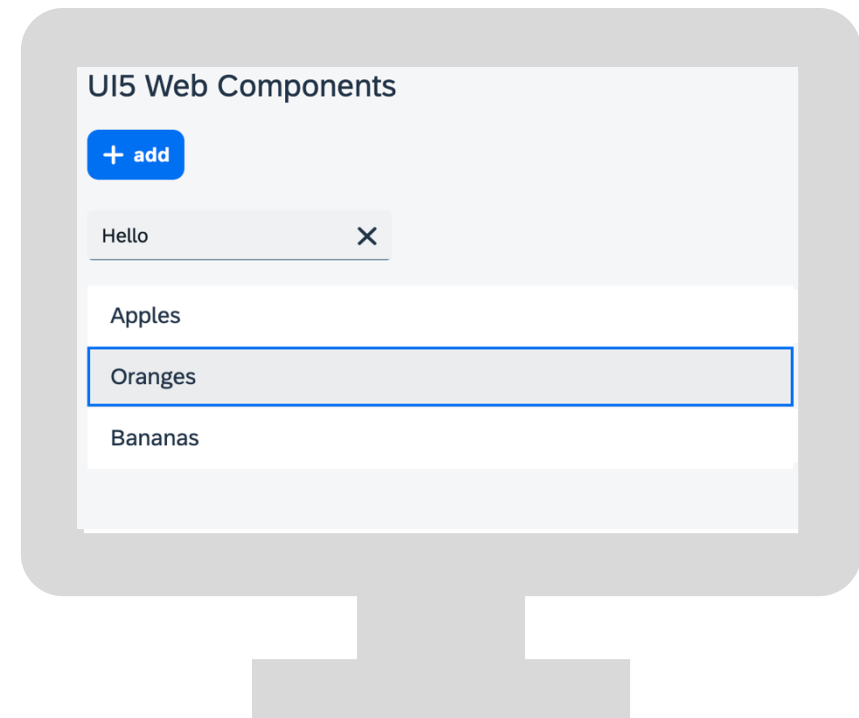
[Video: What is SAPUI5?](#)

# UI5 Web Components in a nutshell

```
28 <body class="sapUiBody">
29   <h2>UI5 Web Components</h2>
30
31   <button>add</button><br>
32   <input placeholder="Hello"></input>
33   <ul>
34     <li>Apples</li>
35     <li>Oranges</li>
36     <li>Bananas</li>
37   </ul>
38 </body>
```



```
28 <body class="sapUiBody">
29   <h2>UI5 Web Components</h2>
30
31   <ui5-button>add</ui5-button><br>
32   <ui5-input placeholder="Hello"></ui5-input>
33   <ui5-list>
34     <ui5-li>Apples</ui5-li>
35     <ui5-li>Oranges</ui5-li>
36     <ui5-li>Bananas</ui5-li>
37   </ui5-list>
38 </body>
```





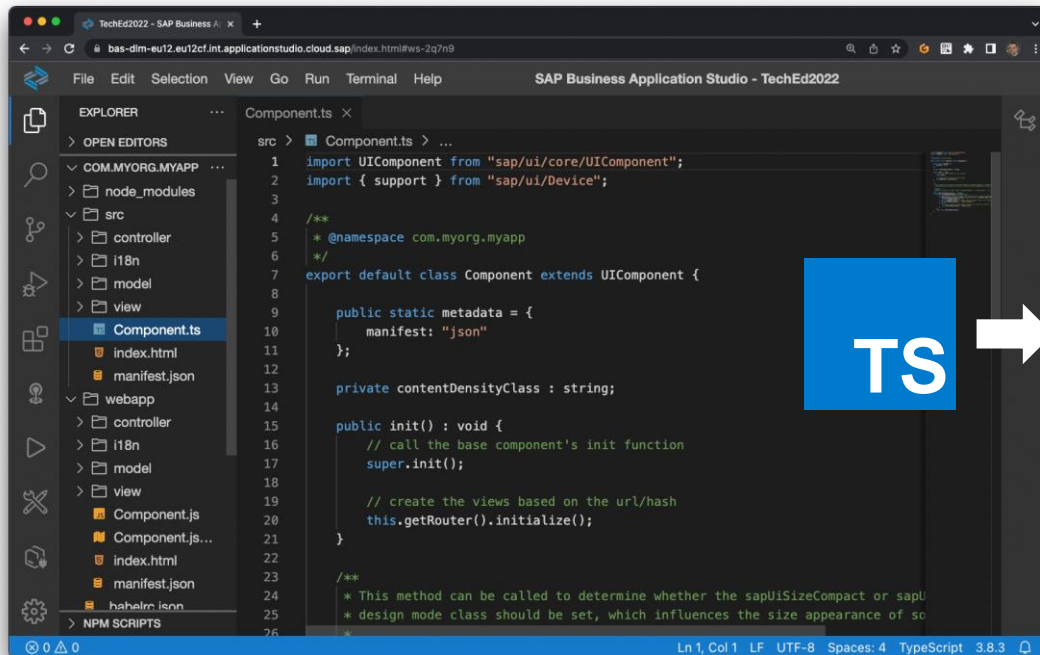
# TypeScript for SAPUI5: Professional application development

Modern ES language features for SAPUI5

Support in **SAP Business Application Studio** and **Visual Studio Code**

Transpiled into SAPUI5 runtime code

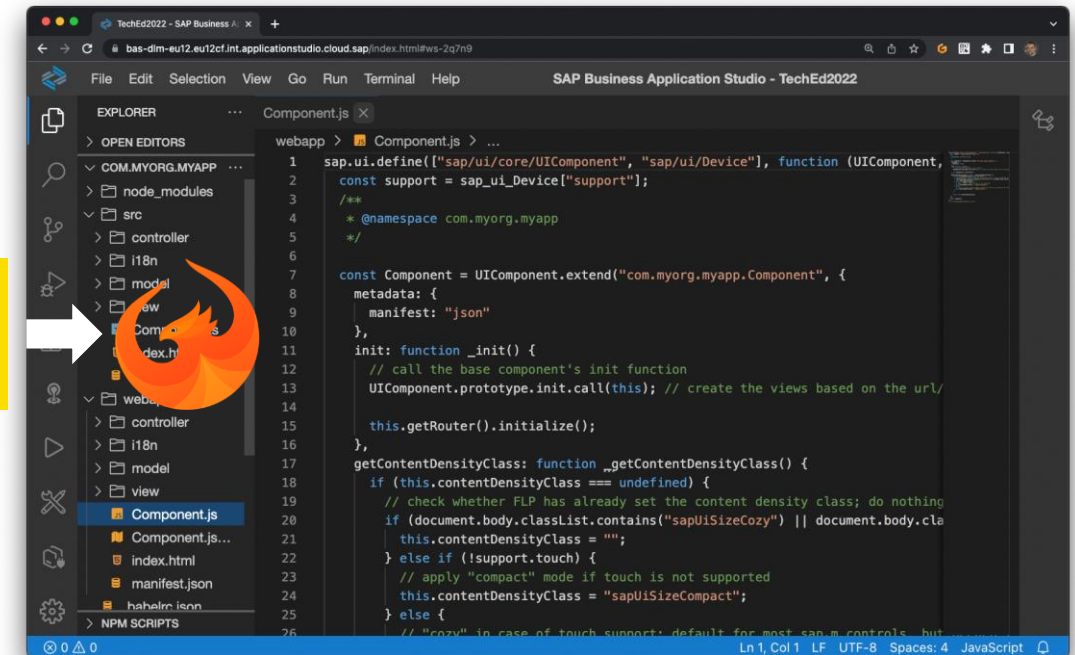
TypeScript as a fully optional layer on top



TS



ES



# Flexible programming model blends SAP Fiori elements and SAPUI5 freestyle

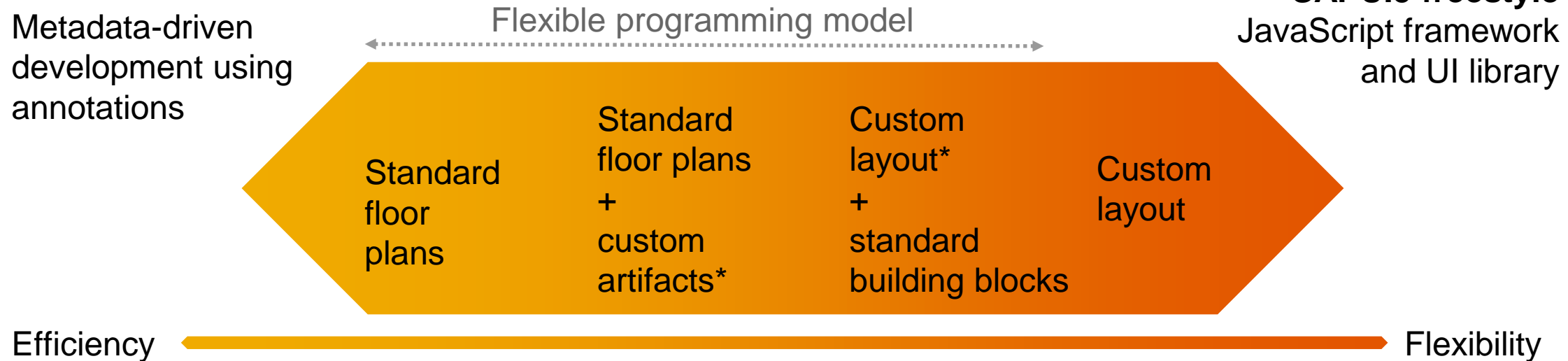
Developers have a continuum of options instead of an either/or decision

## SAP Fiori elements

Metadata-driven  
development using  
annotations

## SAPUI5 freestyle

JavaScript framework  
and UI library



\*Leveraging standard building blocks and/or SAPUI5 controls and custom code

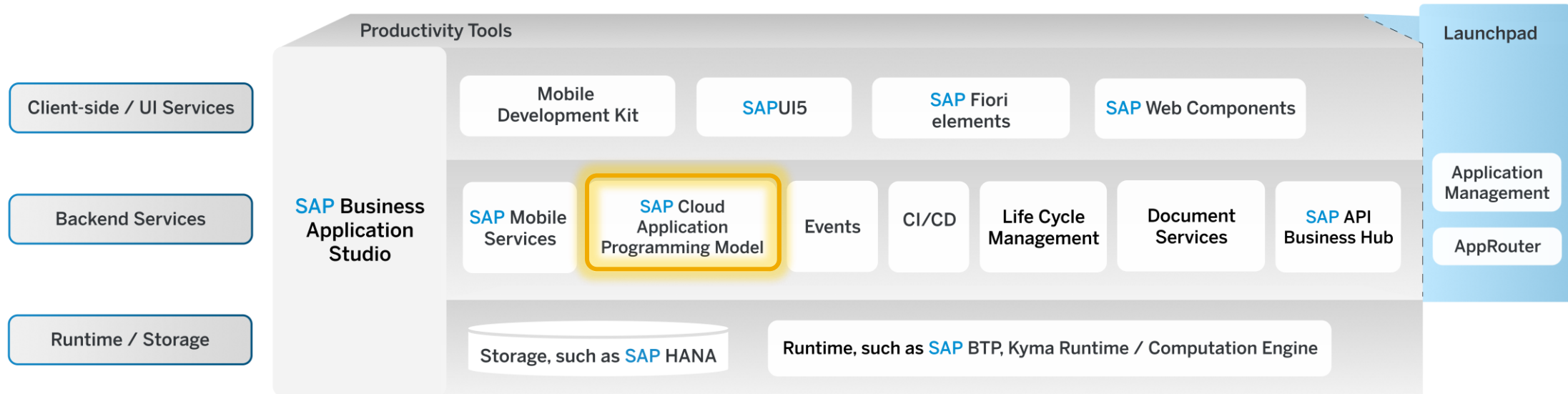
# **The Backend**

# **Cloud Application Programming Model**

# **(CAP)**

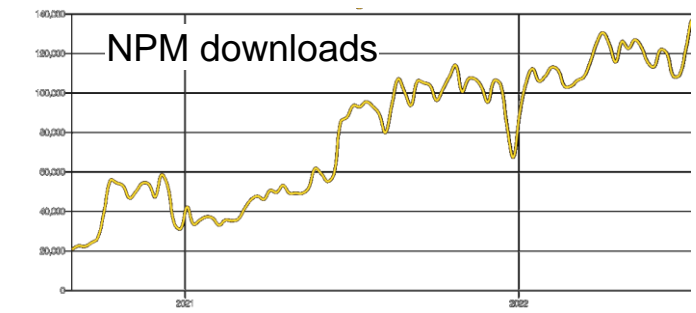
# Productivity tools for a comprehensive pro-code offering on SAP BTP

Quickly build enterprise-ready applications based on opinionated technologies from SAP



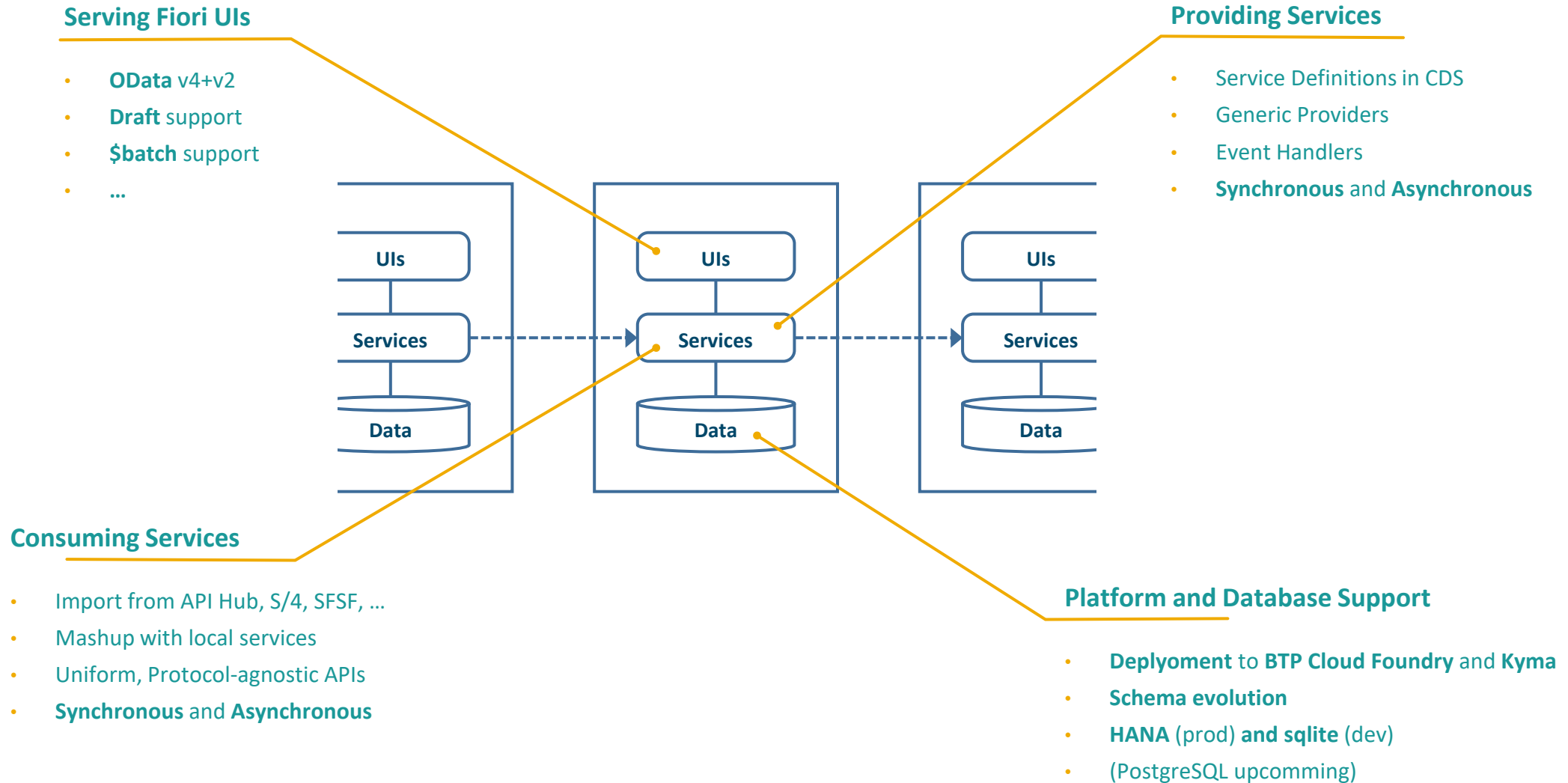
**SAP Cloud Application Programming Model** is the most adopted framework in SAP (non-ABAP hemisphere).

- **100+** internal stakeholders
  - **100+** customers and partners
  - **500+ blogs** featuring SAP Cloud Application Programming Model
- All growing steadily...



# CAP Cloud Application Programming Model

## Features Overview



# CAP Modeling through CDS: Define “what”, not “how”

```
Data Model Editor  schema.cds ×
namespace user001;

using { API_BUSINESS_PARTNER } from './srv/external/API_BUSINESS_PARTNER.cds';

using
{
    Country,
    Currency,
    Language,
    User,
    cuid,
    extensible,
    managed,
    temporal
}
from '@sap/cds/common';

entity Capex
{
    key ID : UUID
        @Core.Computed;
    description : String(100);
    totalcost : Integer;
    category : Association to one Category;
    BusinessPartner : Association to one API_BUSINESS_PARTNER.A_BusinessPartner;
}

entity Category
{
    key ID : Integer;
    name : String(100);
}
```

- **Data Model Definition** creating or importing entities

```
service.cds ×
using { API_BUSINESS_PARTNER } from './external/API_BUSINESS_PARTNER.cds';

using { user001 as my } from './db/schema';

using user001 from './db/schema';

@path : 'service/user001'
service user001Service
{
    @odata.draft.enabled
    entity Capex as
        projection on my.Capex
        {
            *
        };

    entity Category as
        projection on my.Category
        {
            *
        };

    entity BusinessPartner as
        projection on API_BUSINESS_PARTNER.A_BusinessPartner
        {
            BusinessPartner,
            FirstName,
            LastName
        };
}

annotate user001Service with @requires :
[
    'authenticated-user'
];
```

- **Service Definition** creating projections on the data model

# CAP basics

- **Describe the “what” and not the “how” via CDS**
- **Initialize CAP Projects via *cds init***
  - Provides only the bare bone things that you need. No unnecessary files
- **“Grow as you go”: *cds add***
  - Add additional parts only when you actually need them
    - HANA support
    - Deployment (mta, helm)
    - sample / mock data
- **Fast development cycles with *cds watch***
  - Automatic spin up of SQLite locally, automatic switch to HANA after deployment
  - Includes UIs from UI5 and Fiori elements
  - Mock data for remote service calls
  - Dummy authorizations
  - Optionally with hybrid mode (e.g. CAP app local, DB HANA)

## Accelerated development

- Jump-start: **x hrs**    **1 min**
- Turnaround: **x min**    **1 sec**
- First app:    **x days**    **25 min**
- *Grow-as-you-go* with mocked cloud environment    **Minimized costs**

## Focus on domain

- *What, not how* → through CDS
- Capturing **intent** in close collaboration of **developers** and **domain experts**
- Rapid prototyping for reqs analysis (from **months** → **days**)

## Proven support for **best practices**

- Served **preconfigured** by generic runtimes, eliminating recurring tasks
- Clear guidance along **golden paths**
- **Minimized code**  
(**100s** → **0–10** lines of code)

## Cloud scale by design

- **Services** and **event**-centric paradigm
- Memory            **~100 mb**
- Response times **~10 ms**
- Requests           **~5.000**

## Intrinsic **extensibility**

- **Customizations** by SaaS tenants
- **Verticalization** by partners
- **Feature-toggled** extensions by SaaS providers

## Safeguarded investments

- **Minimized lock-ins** to platform, protocols, and databases
- **Higher-level APIs**, shielding users from utterly technical disciplines



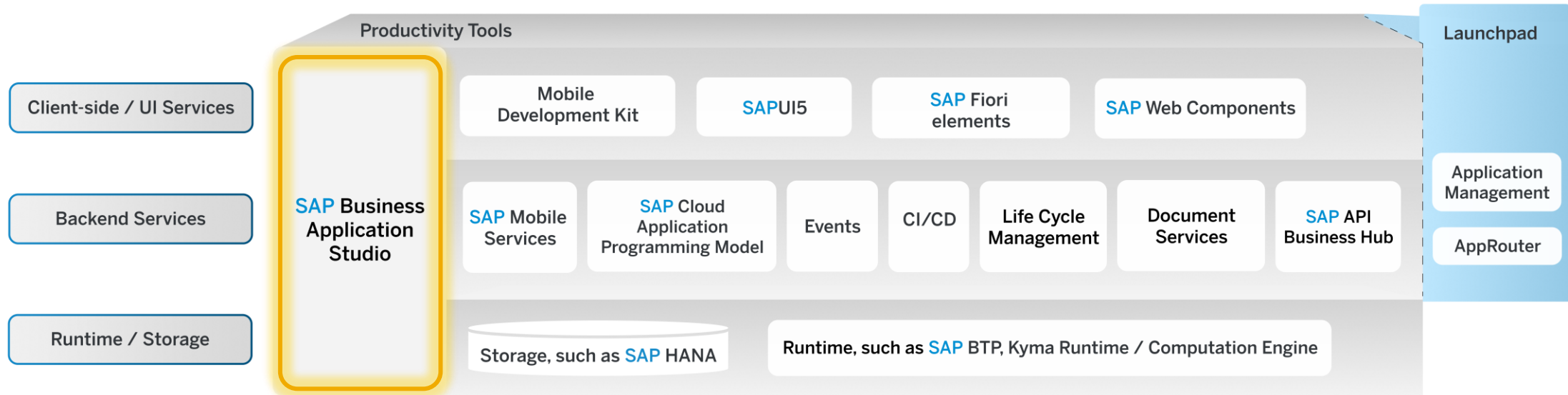
# **SAP Business Application Studio**

The high-productivity approach for  
professional developers



# Productivity tools for a comprehensive pro-code offering on SAP BTP

Quickly build enterprise-ready applications based on opinionated technologies from SAP



- **SAP Business Application Studio**, the preferred IDE to develop applications and extensions on SAP BTP
- Core technologies from SAP in one development environment with all necessary design-time and runtime services
- Fully flexible developer experience – from application composition through graphical productivity tools to command line; code-exit possible at any time
- Optimal integration to SAP data for consumption and deployment (cloud and on premise)
- Provision of content, support for best practices, and delivery of tools for guiding developers
- Complement to ABAP and SAP BTP, Kyma runtime for UI and mobile development



# SAP Business Application Studio: The high-productivity approach for professional developers

Scale your development and build smarter enterprise apps



- **Visual development environment**, tailored for efficient development of business applications for the Intelligent Enterprise
- Intuitive path **from visual tools to professional development tools**



- **Modern and intuitive** Web user interface
- **Simplified, faster setup** of the development environment
- **Tight integration** with SAP services, technologies, and systems



- **Accelerated time to market** with high-productivity development tools
- Based on **industry best-of-breed IDE and tools**, providing full, tool-independent, developer flexibility

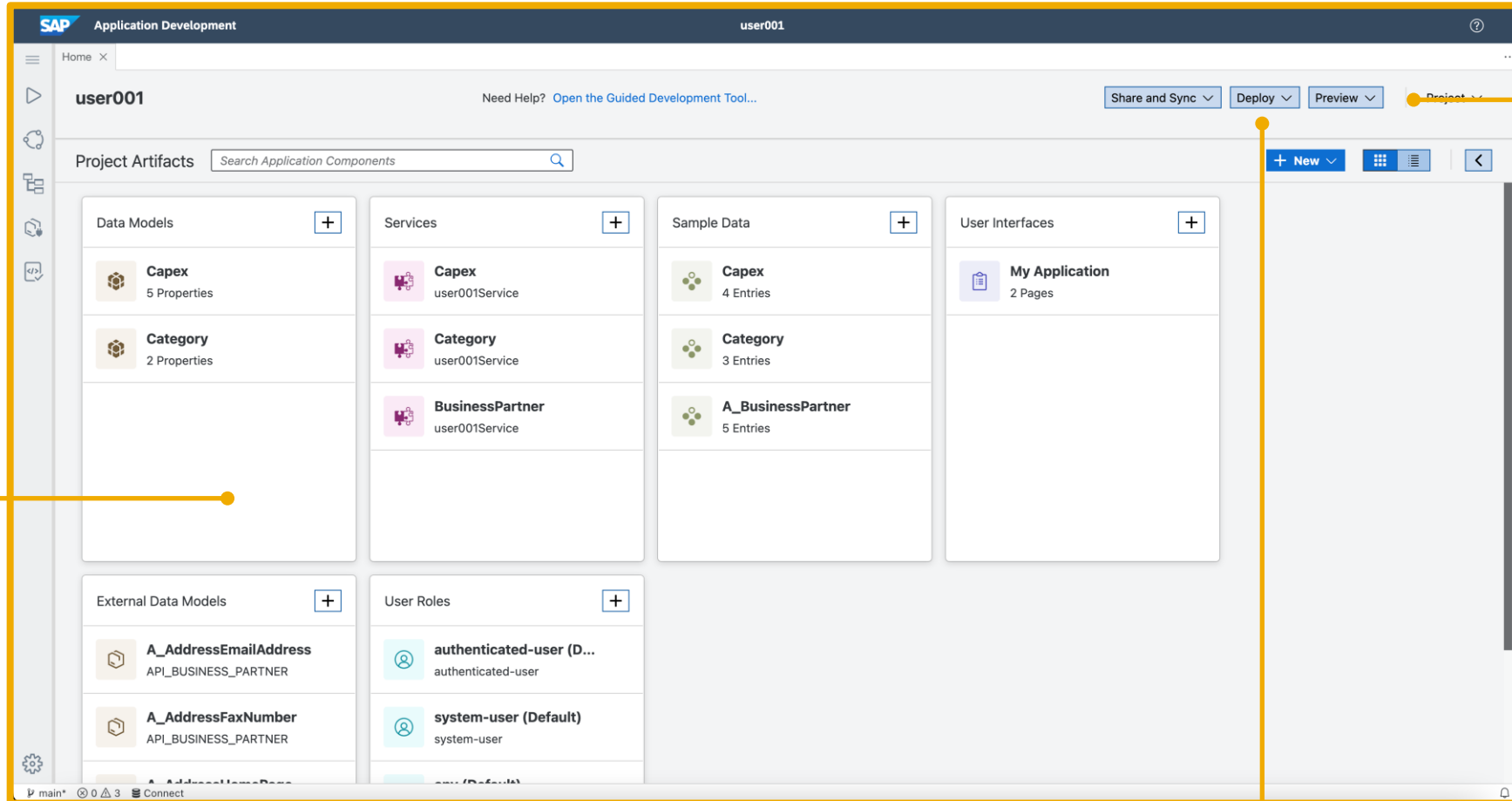


- **Highly adopted SAP BTP service**, across all channels, with thousands of paying customers
- Serves as the official tool of choice for **thousands of internal SAP developers**

# Simplify Application Development for Professional Developers

Enhanced development environment and user experience

- **Visual overview of all assets of a project** combining
  - CAP data models and services
  - Fiori elements applications
  - Mobile Development (MDK) Applications
  - External Services like S/4 HANA APIs

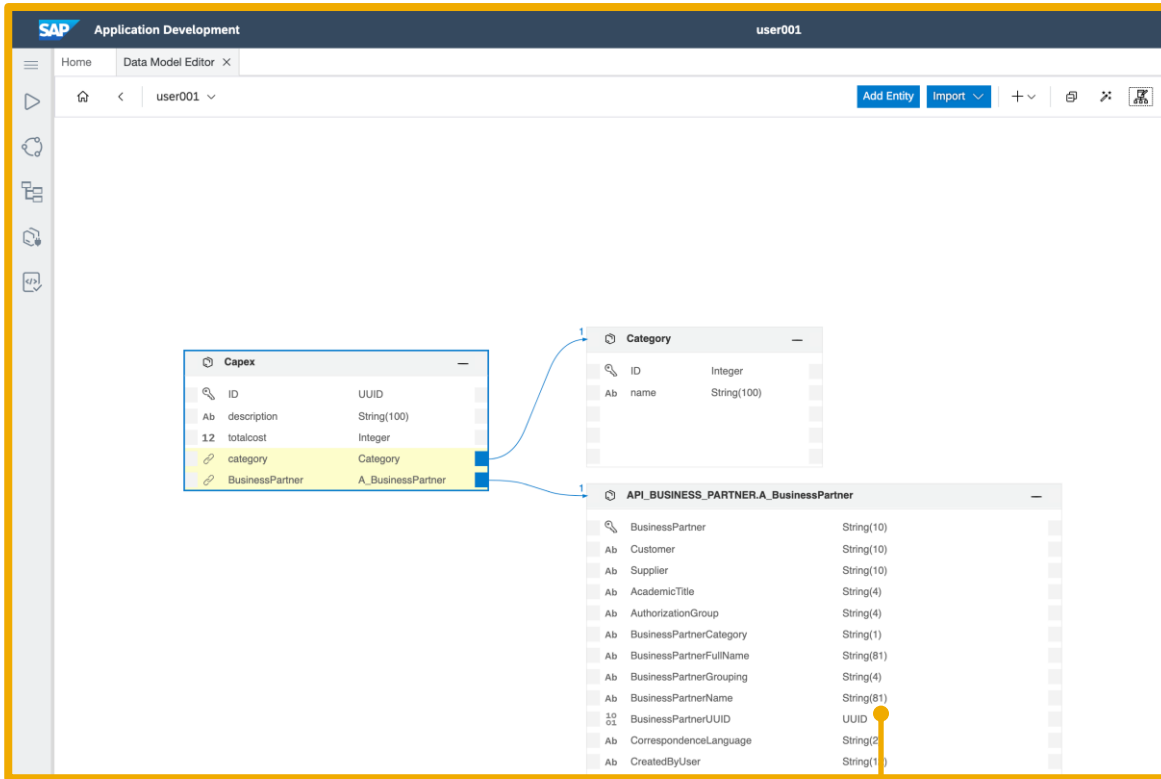


- **Preview** for fast development turnaround times

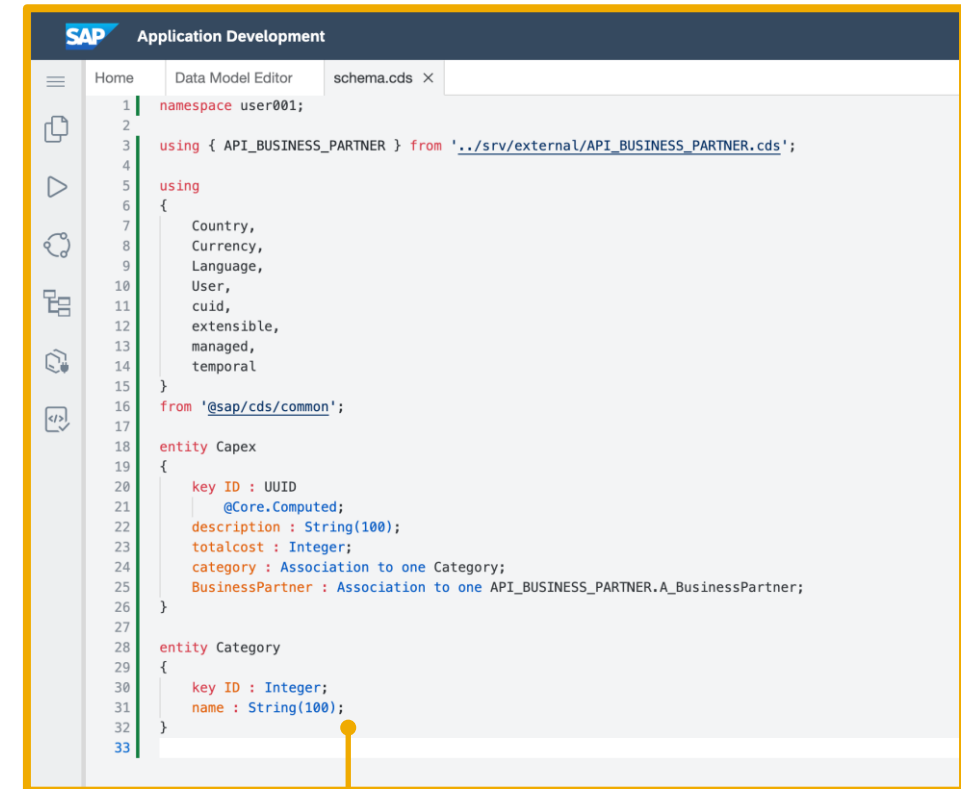
- **One Click Deployment** to the BTP by generating all the necessary configurations

# Simplify Application Development for Professional Developers

Visual Data Modeling: Switch from visual tools to textual editors



• **Visual editors** provide high productivity but create standard code with 0 additional meta data



• **Switch** possible for all editors: CDS/CAP, Fiori elements, MDK + Deployment Config

• **Textual code editors** to allow developers to continue with normal coding for the last mile of the app

# Simplify Application Development for Professional Developers

## Application Business Logic: CAP Handlers

The screenshot shows the 'Application Logics' configuration interface. On the left, a sidebar lists 'Application Logics(1)' with a plus icon, followed by 'NoDevService.Capex' and 'capex-logic' (selected, with 'No description' below it). The main area is titled 'capex-logic' with 'NoDevService.Capex' and 'No description' below it. It features two tabs: 'CONFIGURATION' (active) and 'DETAILS'. Under 'CONFIGURATION', there are three sections: 'Phase' with a description and radio buttons for 'Before', 'On', and 'After' (selected); 'Standard Event' with a description and checkboxes for 'Create', 'Read' (checked), 'Update', and 'Delete'; and 'Custom Event' with a description and an 'Add' button. An orange line with a dot at the end points from the 'Add' button to the bullet point below.

- **Application Business Logic** stub generators for CAP

# Simplify Application Development for Professional Developers

## Choice of UI Application types in Productivity Perspective: Fiori elements and Mobile Development Kit

- **Fiori elements apps** responsive, works on all devices but optimized for Desktop

- **MDK apps** responsive, works on all devices but optimized for Mobile

UI Application Type

What kind of application do you want to create? \*

**Template-Based, Responsive Application**

Create a browser-based application with standard yet extensible floorplans, runs on desktop and mobile. It is derived from your data structures and metadata, automatically applies the latest SAP Fiori

[More Information](#)

**Mobile-Centric, Freestyle Application**

Create an application to run natively on mobile (Android & iOS) with device specific features such as biometric authentication and barcode scanning. It also runs on desktop browsers.

[More Information](#)

[Back](#) [Next](#)

Page Editor - MyApplication

Page Map

Show descriptions ☒ Search properties

CapexObjectPage

Header

Actions

Header Sections

Page Layout

Sections

General Information

Form

Fields

description

totalcost

category

BusinessPartner

Footer

Actions

**category**

Value: category\_ID Type: Int32

**Label** (Annotation)

Define text to be displayed as a field label.

category

**Text** (Annotation)

Choose string property describing field value. It will be displayed along or instead of the field value depending on the value of the Text Arrangement property.

None

**Text Arrangement** (Annotation)

Set text position relative to the field value. When set to none, the default positioning will be used as designed in SAP Fiori elements runtime.

**Display Type** (Annotation)

Define how the field is represented in the create and edit mode. Available options depend on the value type. For most value types, Value Help can be defined if service metadata contain entities with the list of eligible options. For string values, also Text Area representation is available.

Value Help

[Edit properties for Value Help](#)

**Restrictions** (Annotation)

Define whether the field input in create/edit mode is mandatory, optional or read-only. Depending on the field configuration in the service, some options might be not available. For example,

- **Page Map for Fiori elements apps** no manual writing of annotations, lots of configs e.g. for creating value helps

# Service Center – Discovery and Connectivity to External Services

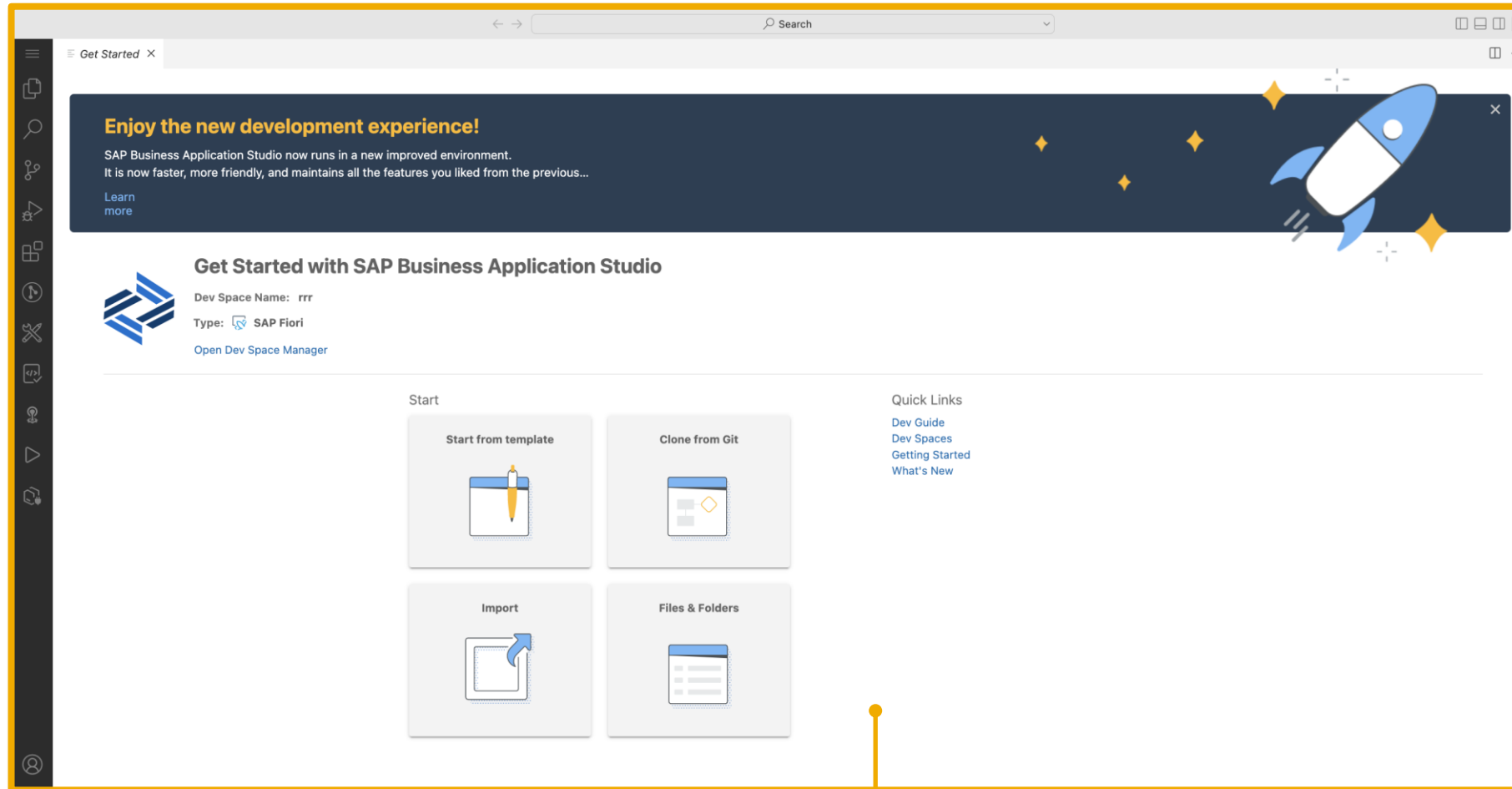
The screenshot displays the SAP Service Center interface. On the left, a sidebar lists various services under 'SAP API BUSINESS HUB', with 'Sales Order (A2X) (V2)' selected. The main area shows the details for 'Sales Order (A2X)' from 'SAP S/4HANA'. It includes a description: 'Create, read, update, and delete sales orders with this synchronous OData API'. A button 'Create Project from Service' is visible. Below, the 'Service Properties' section lists: Service Name: Sales Order (A2X), Protocol: OData V2, Status: Available, Product: SAP S/4HANA, Package: SAP S/4HANA, and URL: [https://api.sap.com/api/OP\\_API\\_SALES\\_ORDER\\_SRV\\_0001](https://api.sap.com/api/OP_API_SALES_ORDER_SRV_0001). At the bottom, the 'Entity: A\_SalesOr...' section has tabs for 'ENTITY DETAILS' and 'LIVE DATA'. The 'ENTITY DETAILS' tab shows a table with columns 'Property Name', 'Kind', and 'Data Type'. The 'LIVE DATA' tab is currently active and empty.

Property Name	Kind	Data Type
SalesOrder	key	string

- **Service center** provides the developer a single entry point for discovery and consumption of SAP APIs, originating from any SAP data sources (Destinations to S/4 HANA, Ariba, C4C, API Hub, SAP API Hub enterprise)



# BAS Platform



- Get VS Code's latest and greatest and better performance
- BAS extensibility – 100% VSCode compatibility, community extensions working out of the box
- **Business Application Studio** comes as a “turn key” solution with all the necessary extensions installed at the developers' fingertips

• **SAP Business Application Studio** includes **Visual Studio code** compatibility powered by Microsoft Code OSS Client

# **Embedded Analytics**

## **Business Application Studio, HANA Cloud, CAP, Fiori elements**

# Embedded Analytics in Fiori elements with OData

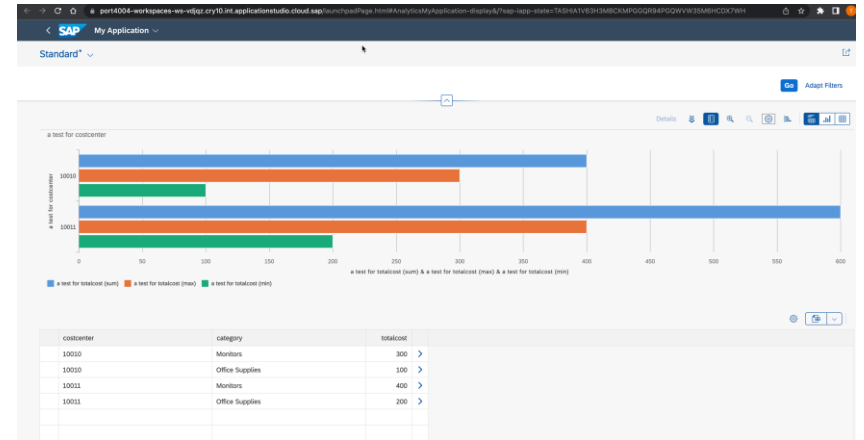
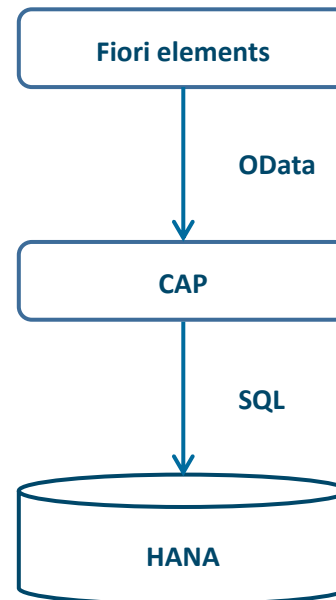
CAP and Fiori elements support analytical content based on OData Service analytical capabilities

- Based on just annotations for
  - Data model (dimensions and measures)
  - UI configuration
- Free of charge for any customer

Used to be a completely new floorplan (Analytical List Page, ALP), now just an additional feature in any List Report Object Page floorplan, so available to a large amount of applications

Only supports "basic analytics" that OData supports

- No multi-dimensional analytics, like Pivot tables



# Analytical Charts in Your List Application

In order to add basic embedded analytics in your list application, you can

- write CDS annotations like this to model dimensions and measures

```
service CapexService
{
  annotate Capex
  {
    totalcost
    @Aggregation.default : #sum;
  }

  annotate Capex with @Aggregation.ApplySupported :
  {
    $Type : 'Aggregation.ApplySupportedType',
    GroupableProperties :
    {
      category_ID,
      BusinessPartner_BusinessPartner
    },
    AggregatableProperties :
    {
      {
        Property : totalcost
      }
    }
  };

  @Aggregation.CustomAggregate#totalcost : 'Edm.Int32'
}
```

- And like this to configure the chart underneath the list

```
annotate service.Capex with @(
  UI.Chart #alpChart : {
    $Type : 'UI.ChartDefinitionType',
    ChartType : #Pie,
    Dimensions : [
      category_ID,
    ],
    Measures : [
      totalcost,
    ],
  }
);
```

... but also work in an editor that

Property Name	Data Type	None	Groupable	Aggregatable	Aggregation Method
ID	UUID	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	min
description	String	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	min
totalcost	Integer	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>	sum
category	Capex.Category	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	min
BusinessPartner	API_BUSINESS_PARTNER.A-BusinessPartner	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	min

Buttons: Update, Previous, Cancel

- Lets you define dimensions and measures next to your data model properties

- Lets you add a chart and its configuration

CapexList

Variant Management

Add Chart

Chart Type: Pie

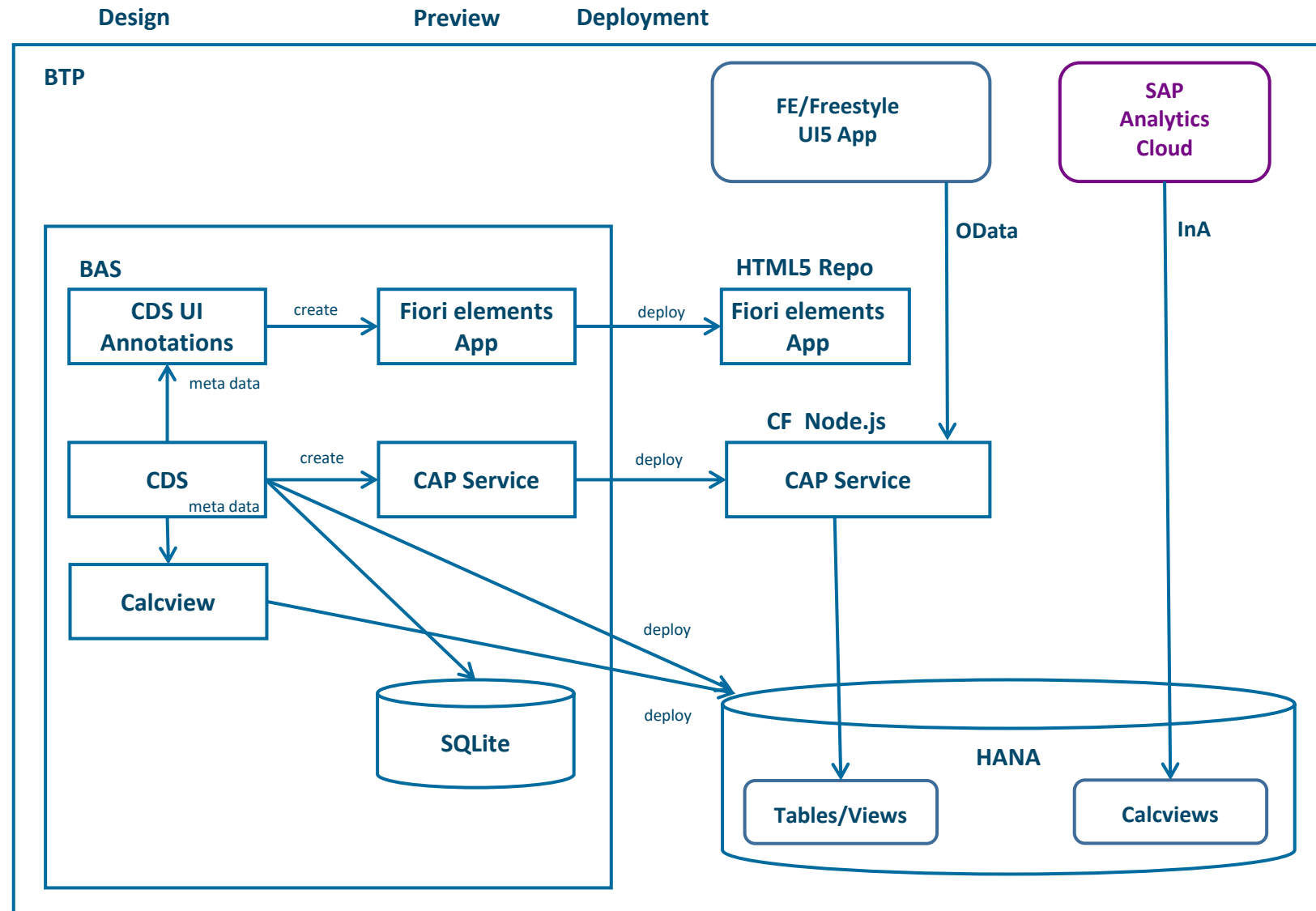
Dimension: category\_ID

Measure: Use existing measure (selected) | Create new measure

Name: totalcost

Buttons: Add, Cancel

# New: Design of the Calcview from CDS Entities Without Prior Deployment



# **DevOps with BTP**

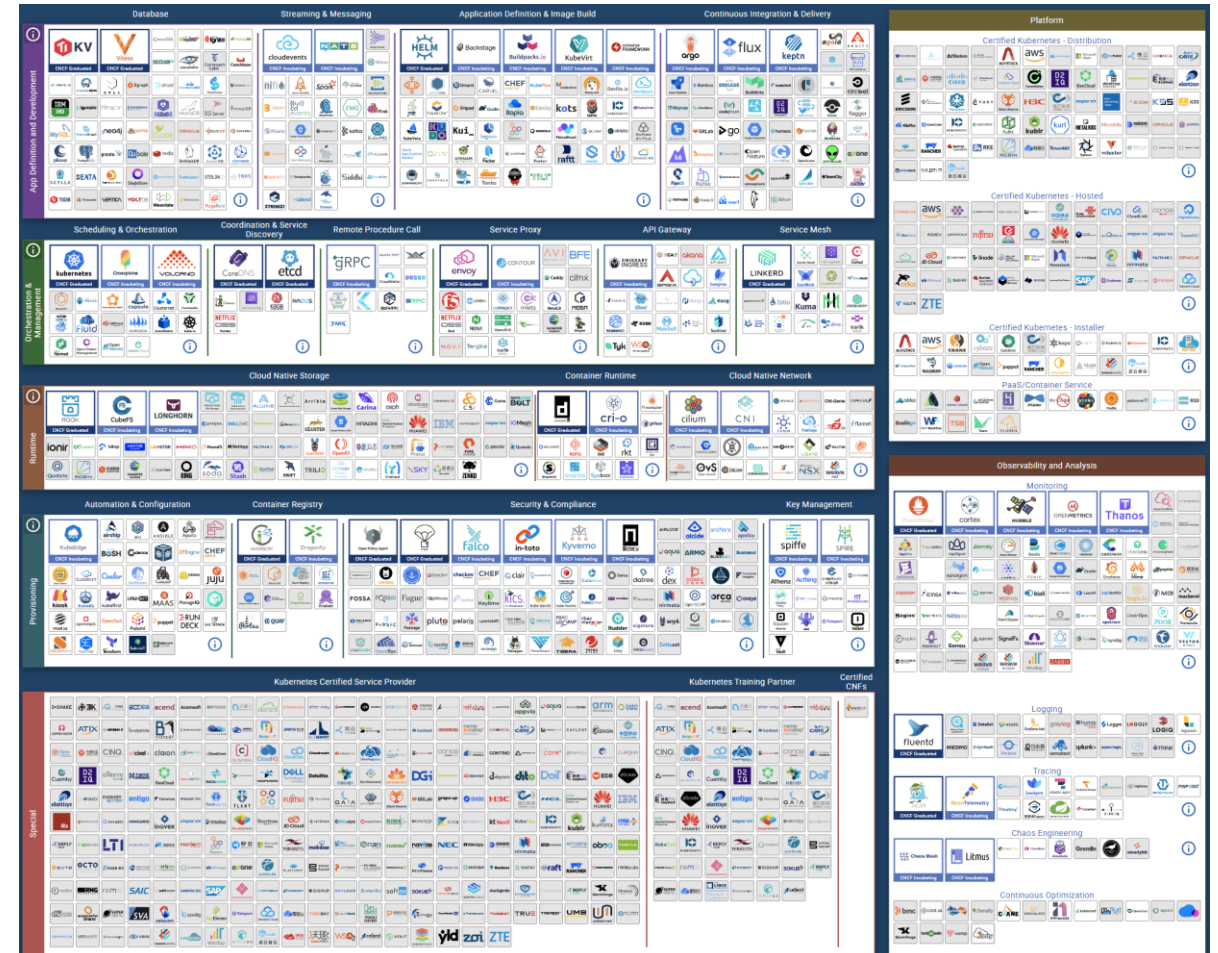
# Implied complexity for finding the right DevOps approach

Finding the right approach and toolset can be a long, complex and challenging task

- Manifold toolchain options and variants
- Where to find a good starting point – taking you where you are today?
- How to reduce cognitive load of your teams and avoid overburdening them right from the start?

Especially if approached as part of bigger cloud transformation, concentrate on quick wins first

- Confirm and verify the value an agile DevOps approach can bring – in your environment, for your boundary conditions
- Start opinionated for SAP use cases
- If required, adapt and evolve by time



[CNCF Cloud Native Interactive Landscape](#)

# DevOps with SAP BTP – easy start + highly integrated

SAP BTP makes it easy for you to benefit from DevOps principles for your apps running on SAP BTP, as part of your cloud transformation

- Provides opinionated approach focused especially for SAP-centric scenarios and use cases
- Respects existing ops processes – integrate into change management and operations, as needed
- If you should bring DevOps expertise and existing infrastructure, we support you to bring in SAP-specific aspects
- Examples:

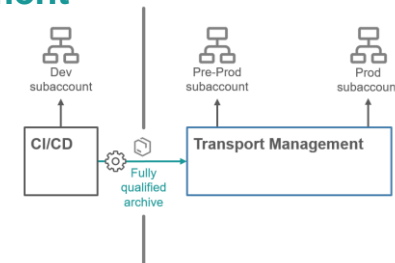
**‘Pipeline-as-a-Service’ + golden paths – running CI/CD pipeline for SAP-specific use cases in minutes, with direct support from SAP**



**Respects existing ops processes – integrate into change management and operations, as needed; such as running on SAP Solution Manager or SAP Cloud ALM**



**Highly integrated standard setup for agile development on SAP BTP – covering the complete lifecycle from development to operations; example: automated hand-over of qualified changes into transport management**





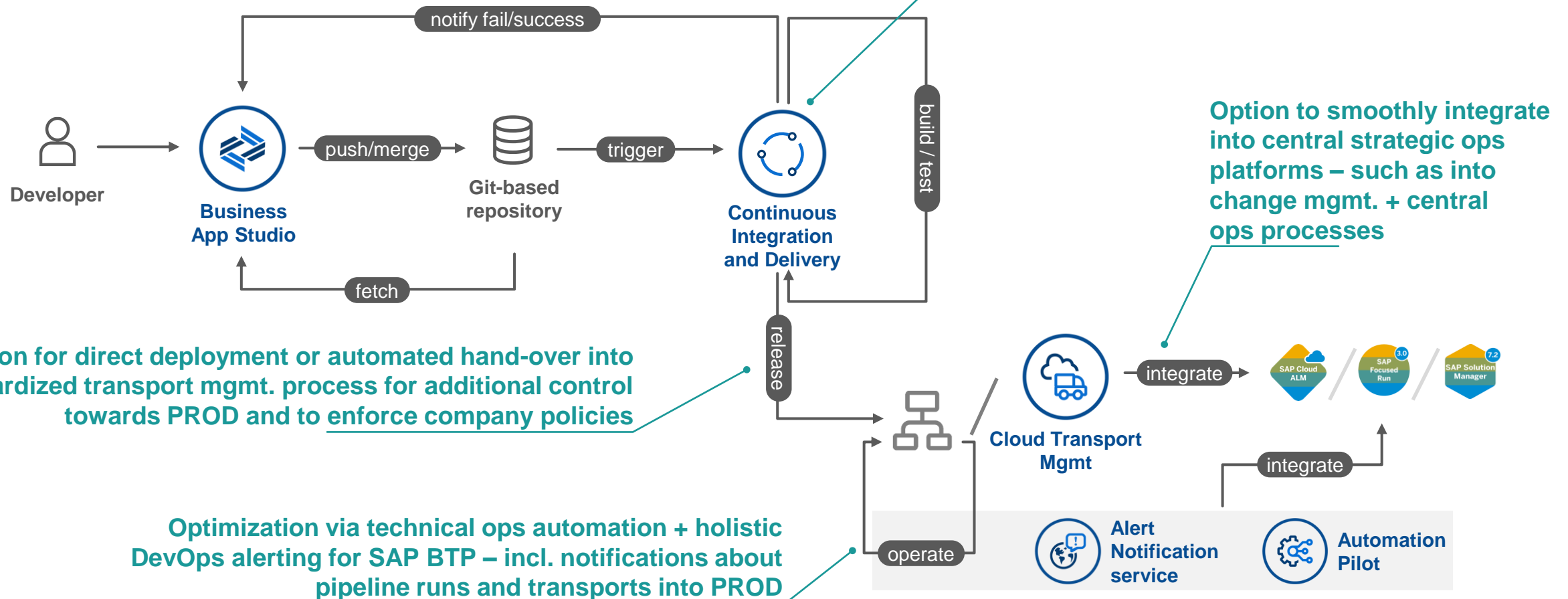
# How to enable DevOps with SAP BTP?

Benefit from highly integrated standard setup especially for SAP-centric dev use cases

**Tighter feedback loops**  
**Reduced risk, better code quality**

Container-Based Applications  
SAP Cloud Application Programming Model  
SAP Fiori in the Cloud Foundry environment  
SAP Fiori in the Neo environment  
SAP Integration Suite Artifacts

**'Pipeline-as-a-Service' + golden paths – running CI/CD pipeline for SAP-specific use cases in minutes**



**Option for direct deployment or automated hand-over into standardized transport mgmt. process for additional control towards PROD and to enforce company policies**

**Optimization via technical ops automation + holistic DevOps alerting for SAP BTP – incl. notifications about pipeline runs and transports into PROD**

# Out of the box CI/CD Integration in BAS' Productivity Tools

The screenshot displays the SAP Continuous Integration and Delivery (CI/CD) interface. On the left, a table lists various jobs under different categories. On the right, a detailed view of the 'bookshop' pipeline is shown, including its configuration and build retention settings.

Name	State	Branch	Timed Triggers	Pipeline
<b>cicd-api</b>				
bookshop	ON	master	0	SAP Cloud Application Programming Model 1.0
cicd-api	ON	main	0	SAP Cloud Application Programming Model 1.0
<b>cicdtest</b>				
cicdtest	ON	main	0	SAP Cloud Application Programming Model 1.0
<b>riskmanagement</b>				
riskmanagement	ON	main	0	SAP Cloud Application Programming Model 1.0

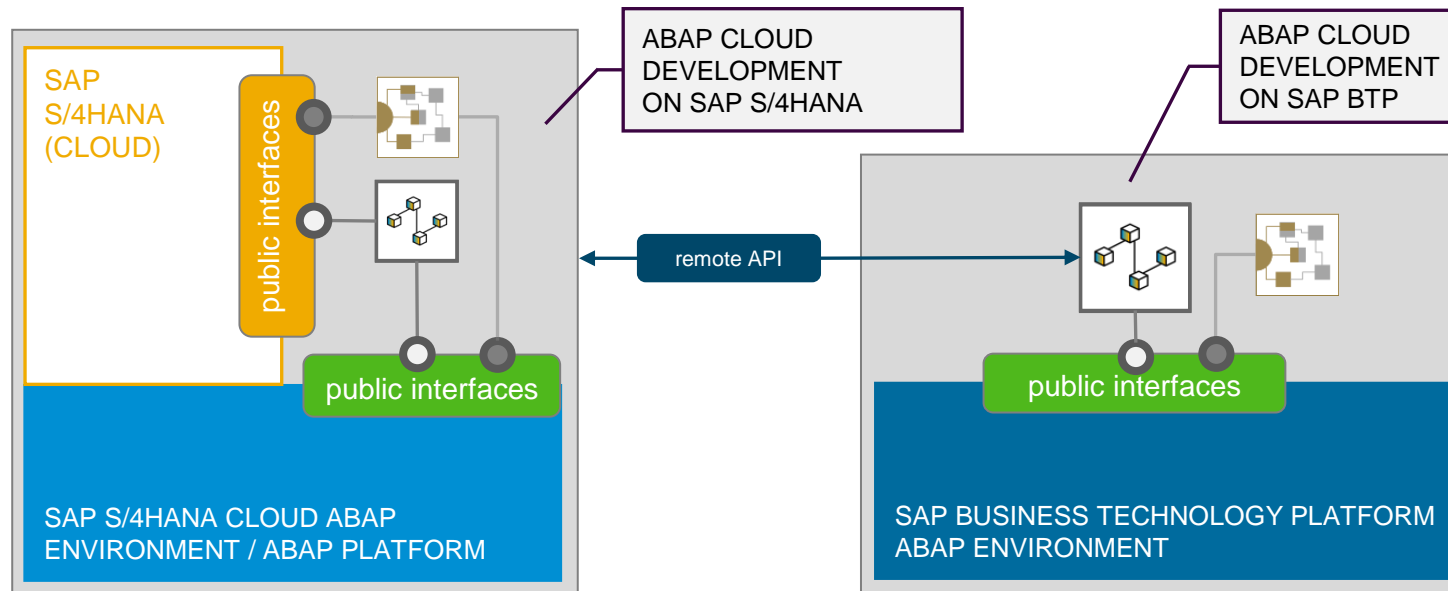
**bookshop** pipeline details:

- Repository: cicd-api
- Server: https://github.tools.sap
- Pipeline: SAP Cloud Application Programming Model
- Builds: No builds yet
- General Information: Description, Repository, Branch, Pipeline, Version, State
- Build Retention: Keep logs for: 7 days, Keep maximum: 50 build items
- Stages

- **Out of the box integration with SAP Continuous Integration and Delivery** for CAP based apps to assist with pipeline configuration and quick access to pipelines

# SAP S/4HANA CLOUD ABAP ENVIRONMENT (Embedded Steampunk) + UI Applications on BTP

# ABAP Cloud on SAP BTP and on SAP S/4HANA



## ABAP Cloud

- ❖ Public SAP APIs and extensions points
- ❖ Cloud-optimized ABAP language
- ❖ ABAP Development Tools
- ❖ ABAP RESTful Application Programming Model

# Use Cases


## New user interfaces or mobile apps on top of S/4

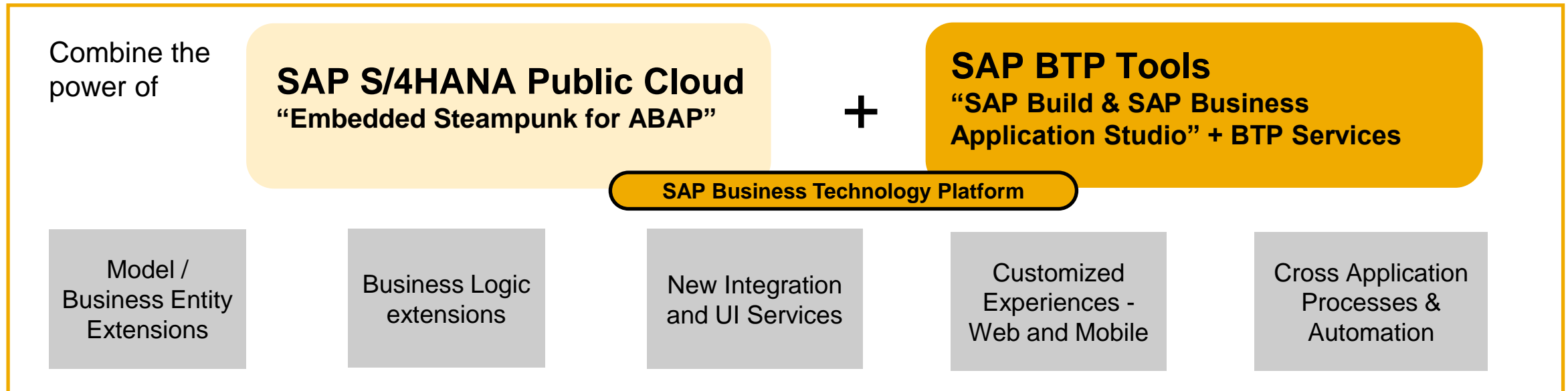
- To build modern Fiori user interfaces public remote APIs are needed
- While S/4 has a lot of remote public APIs, most of them are optimized for UI consumption (Example: The Business Partner API has 43 entities).
- To build optimized remote UI APIs, either just leveraging the rich existing S/4 functionality or extending it with additional customer entities and logic, the ABAP RESTful Programming Model (RAP) on Embedded Steampunk is well suited. To develop these API, developers would use the ABAP Development Tools (ADT).
- To build user interfaces developers use BTP tools and technology like the Business Application Studio (BAS) or Build Apps and deploy the final apps to BTP, Steampunk or Mobile

# Embedded Steampunk and SAP BTP tools

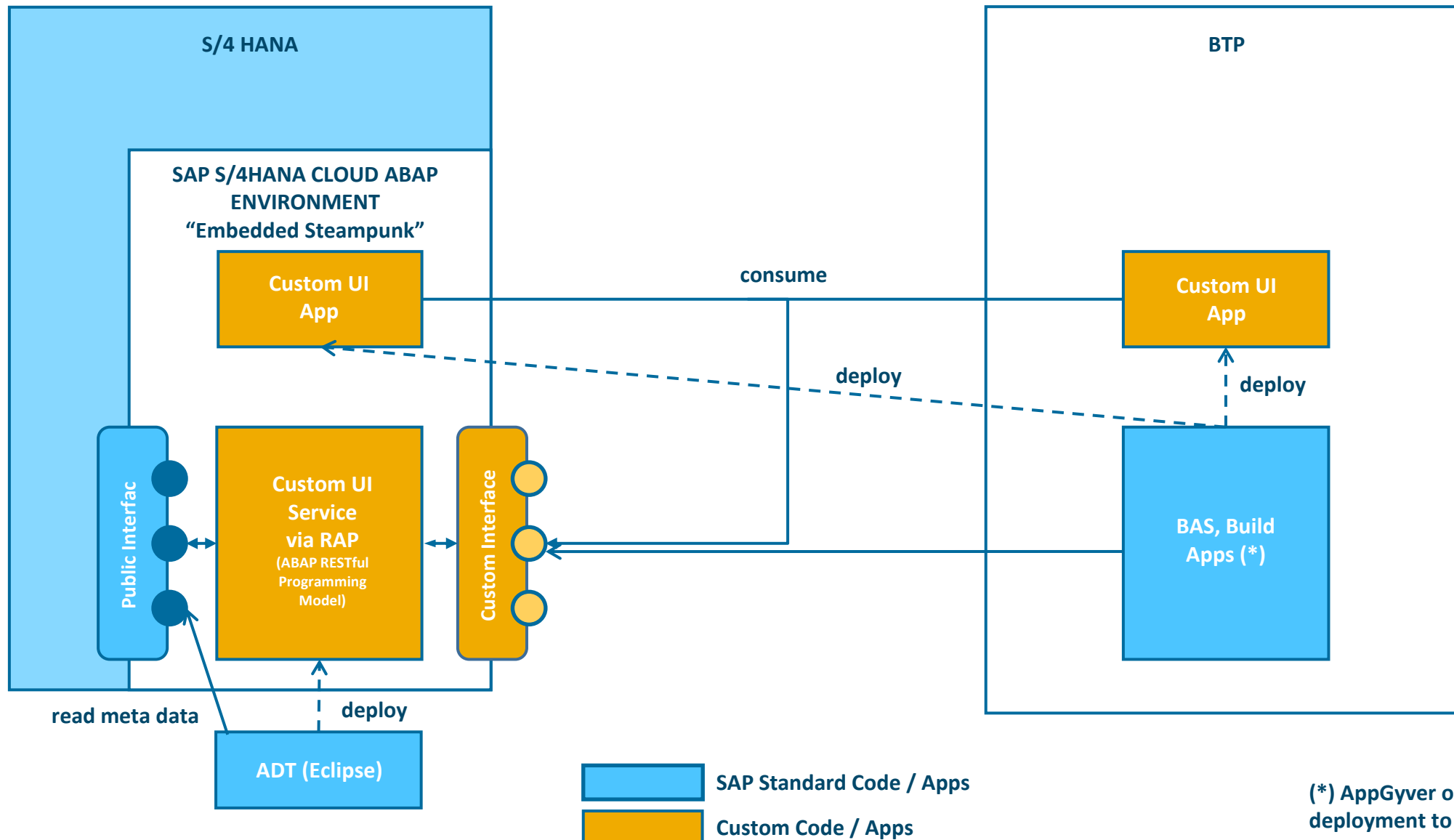
## SAP S/4HANA Extensibility Strategy

- **Keeping the core clean is a key priority for customers**
- Extending business entities is complex, if done outside the core
- Increasing need to deliver custom apps and processes fast

- 
- ▶ Leverage Embedded Steampunk for **business entity and logic extensions**
  - ▶ Leverage BTP Tools for **customized experiences /web/mobile**), as well as **automation and cross-application processes**



# New UI app (customized experiences) on existing S/4 business entities



New process on SAP S/4HANA / LoB business entities using **SAP Build Process Automation (SBPA)**

