



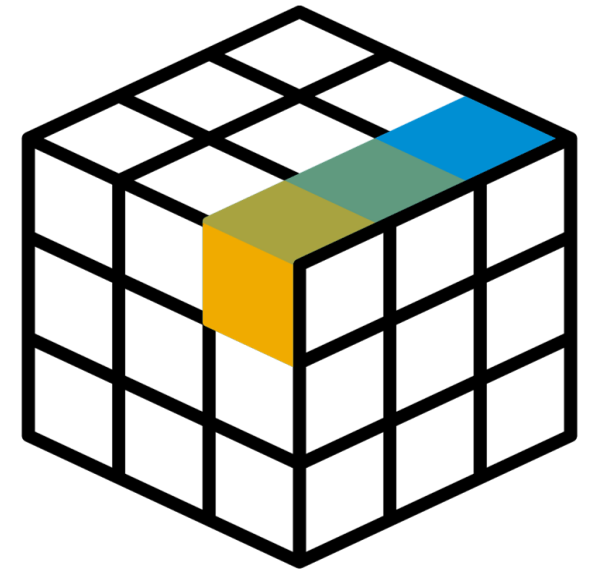
Week 2: New Programming Model

Unit 1: What Is the New Programming Model?

What Is the New Programming Model?

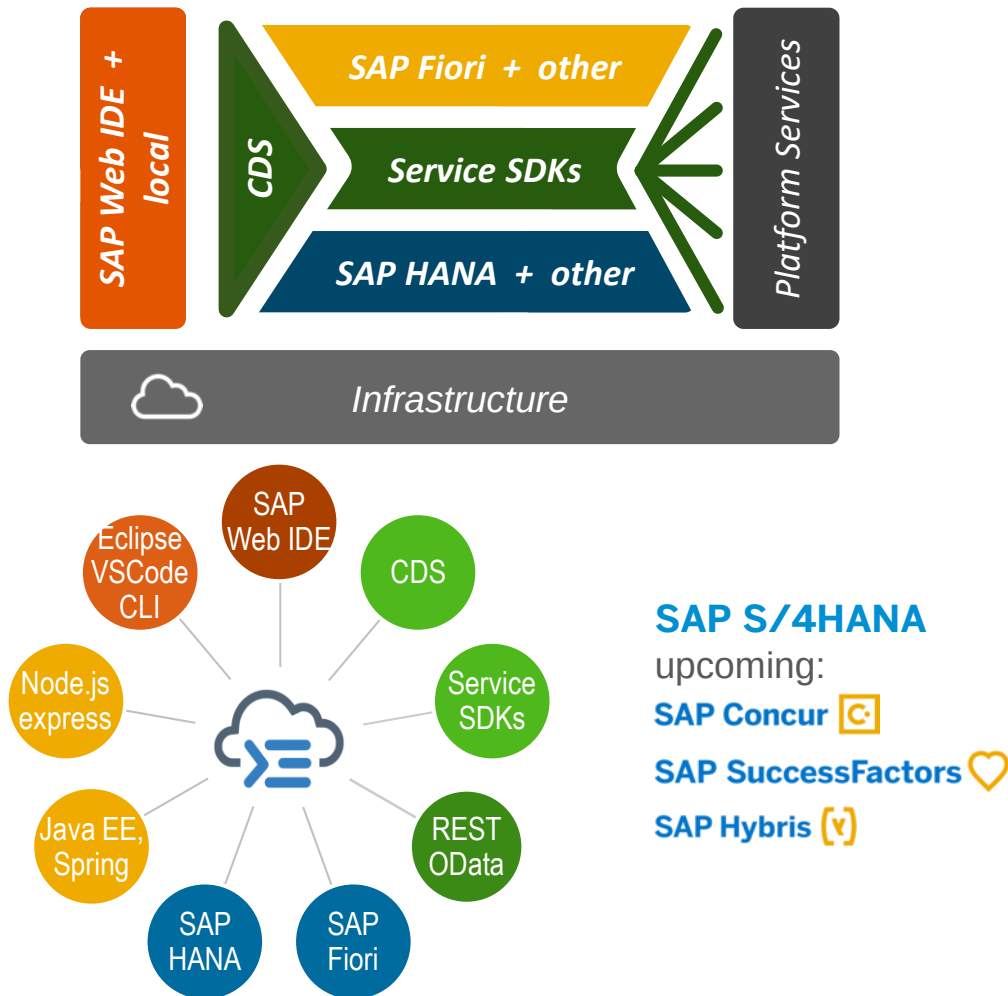
Challenges

- Customers ask for guidance and best practices
- High entry barrier for less tech-savvy developers
- Meet cloud qualities and enterprise qualities
- Focus on business domain logic
- Time to market
- Addressing costs of openness
- Verticalization for different industries
- Customization by SaaS clients



What Is the New Programming Model?

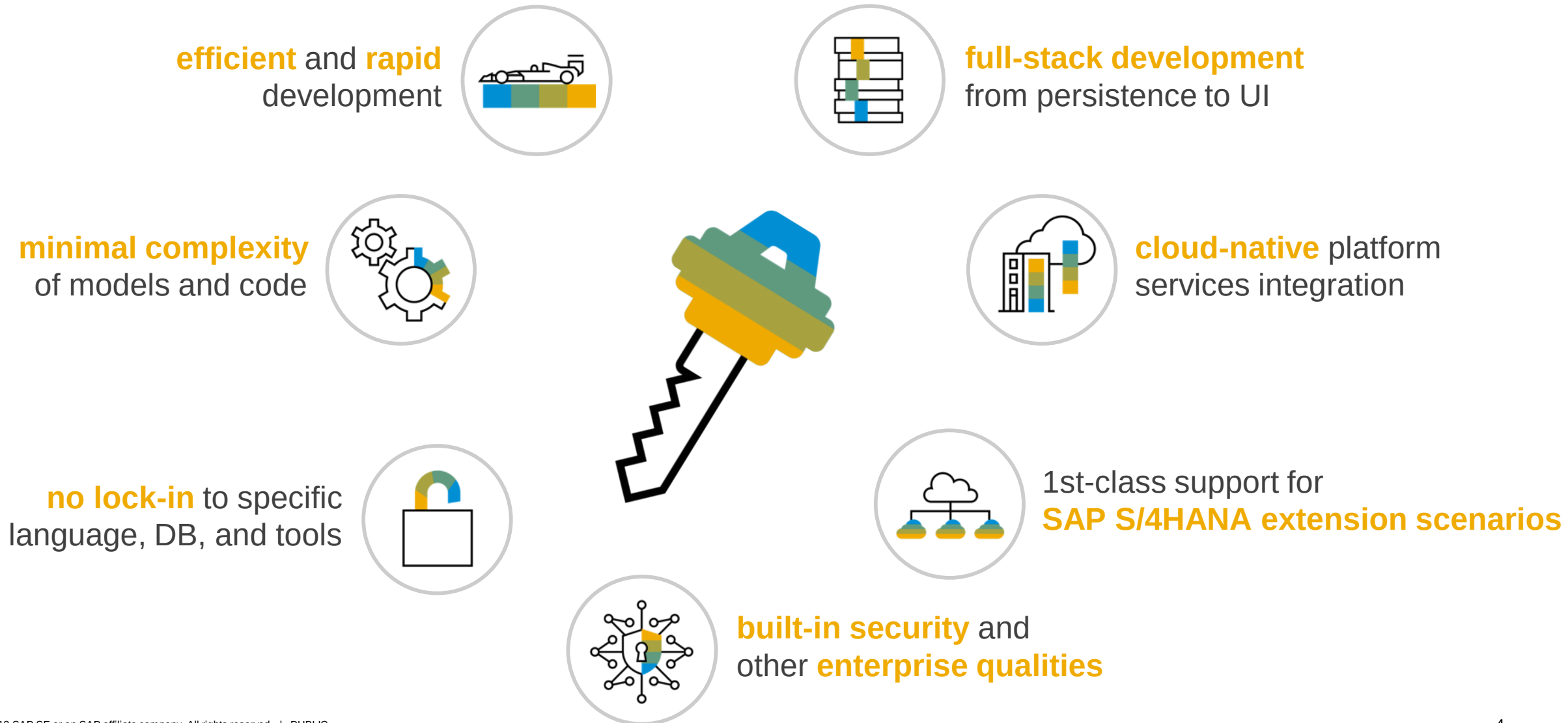
Overview



1. **The application programming model** is a framework comprising of...
 - **Core Data Services** which adds *domain focus* and acts as the *glue* for all below...
 - **Service SDKs** for providing and consuming services in *Java* and *Node.js*
 - **Cloud-native platform integration** w/ common custom handlers and abstraction
 - **SAP Web IDE** w/ tailored tools + **local** tools in *side-by-side* usage on same projects
2. ... with *out-of-the-box* support for...
 - **SAP Fiori + other** UI technologies
 - **SAP HANA + other** DBs
 - **SAP S/4HANA** extension scenarios
3. ... as *synergy* of **SAP technologies + Open Source**
→ A consistent end-to-end **programming model** for enterprise services and applications featuring a **full-stack development** experience

What Is the New Programming Model?

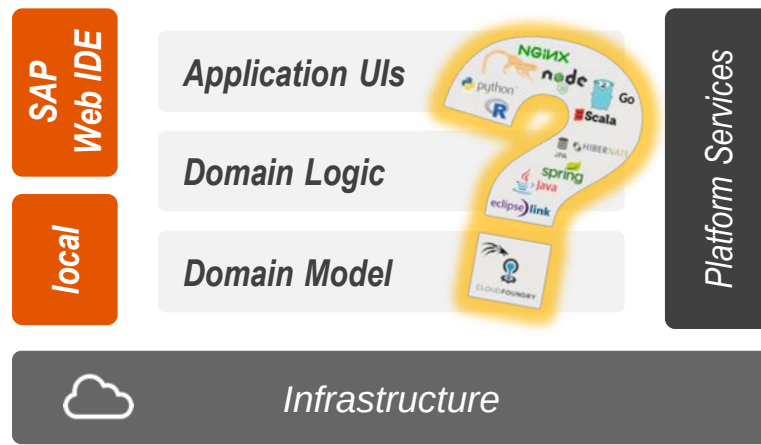
Key benefits



What Is the New Programming Model?

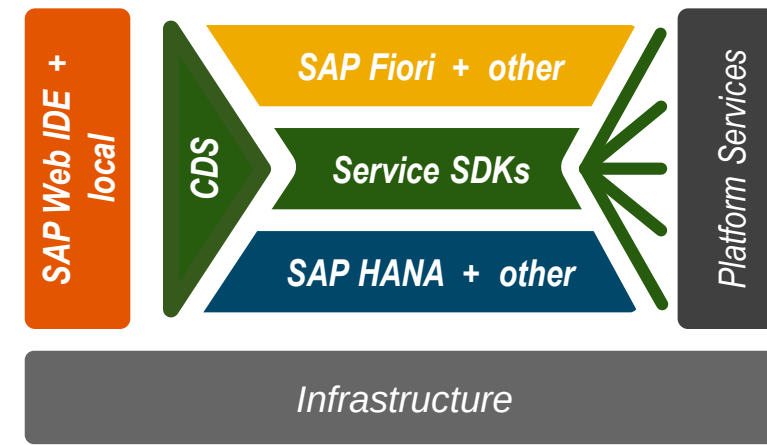
What we provide

complementing cloud-native platform...



- Large variety of open standards and SAP technologies
- How to combine between the technologies and solutions?
- Rather technical focus → *no quick start*
- Each project has to address *platform integration* and *enterprise qualities* individually
- High rates of boilerplate coding
→ *slow time to market and hard to fix/evolve*

with business domain focus

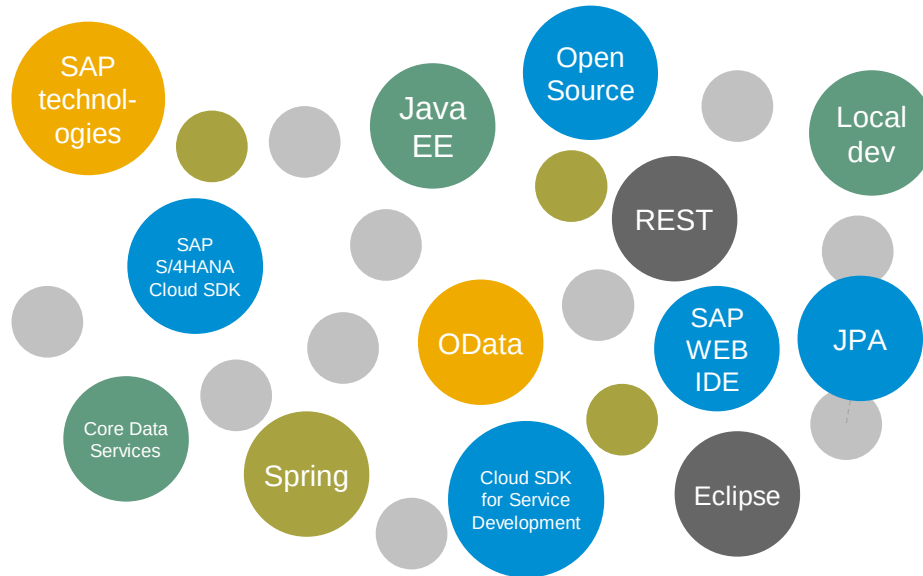


- Clear guidance to seamlessly integrated best practices...
- Common patterns and reuse models → *interoperability*
- Conceptual modeling → *re-focus on business domains*
- Capturing intent → *what not how; single points to fix*
- Automating tedious tasks → *multitenancy, extensibility, authorization, audit logging, data privacy, localization, platform integration, ...*

What Is the New Programming Model?

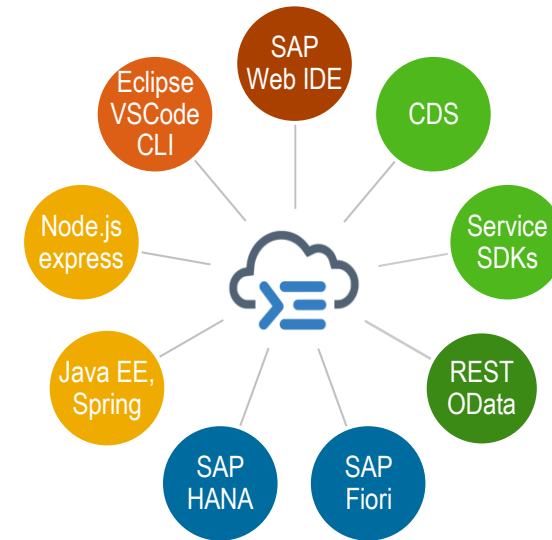
How we did that

from a variety of individual technologies...



- Multitude of choices, such as...
 - SAP Web IDE vs local dev
 - REST vs OData
 - Java vs Node.js
 - Open Source vs proprietary
- Largely heterogeneous results
- Lack of integrated end-to-end experience

to a synergy of best practices



- Choice of seamlessly integrated tools and technologies...
- Open Source *and* SAP technologies
- SAP Web IDE *and* local development
- Tailored tools for the tasks
- CDS as the glue and declarative backbone
- Staying open to choose and combine with alternatives

What Is the New Programming Model?

Toolset – SAP Web IDE + local development

SAP Web IDE provides...

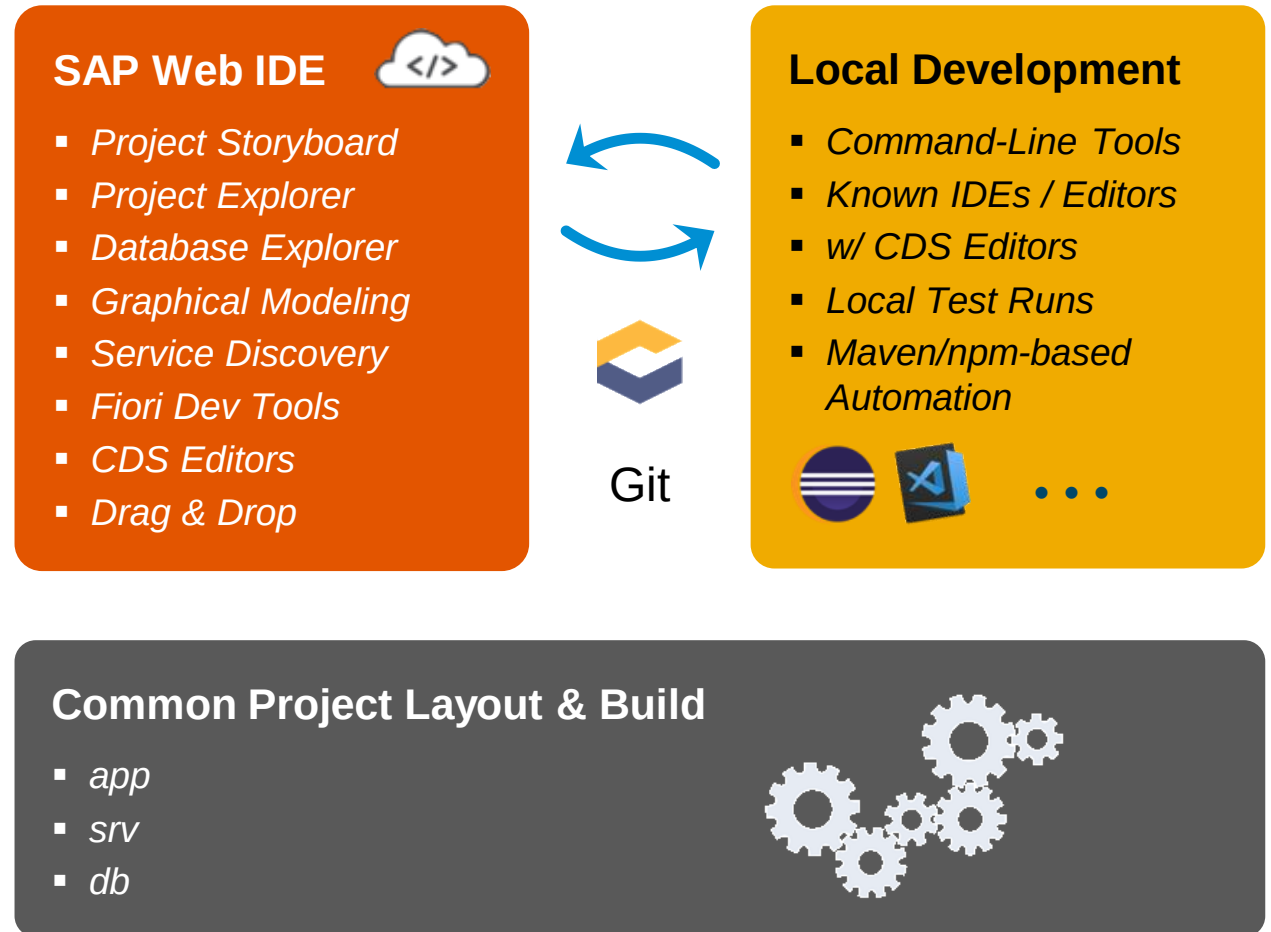
- Jumpstart development without local setup
- Highly integrated development experience
- Easy integration with other services (e.g. S/4)
- Advanced and tailored tools

Local development supported with...

- CDS editors for VS Code and Eclipse
- Command-line tools (CLIs)
- Local test runs

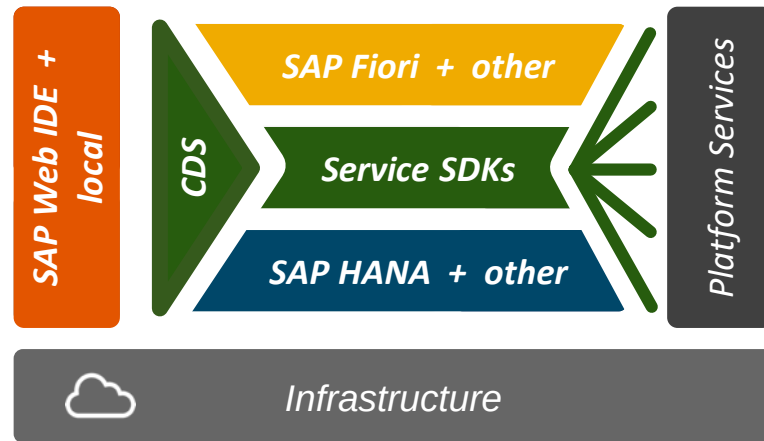
Switching between both using

- Shared workspaces in Che6
- Shared Git repositories



What Is the New Programming Model?

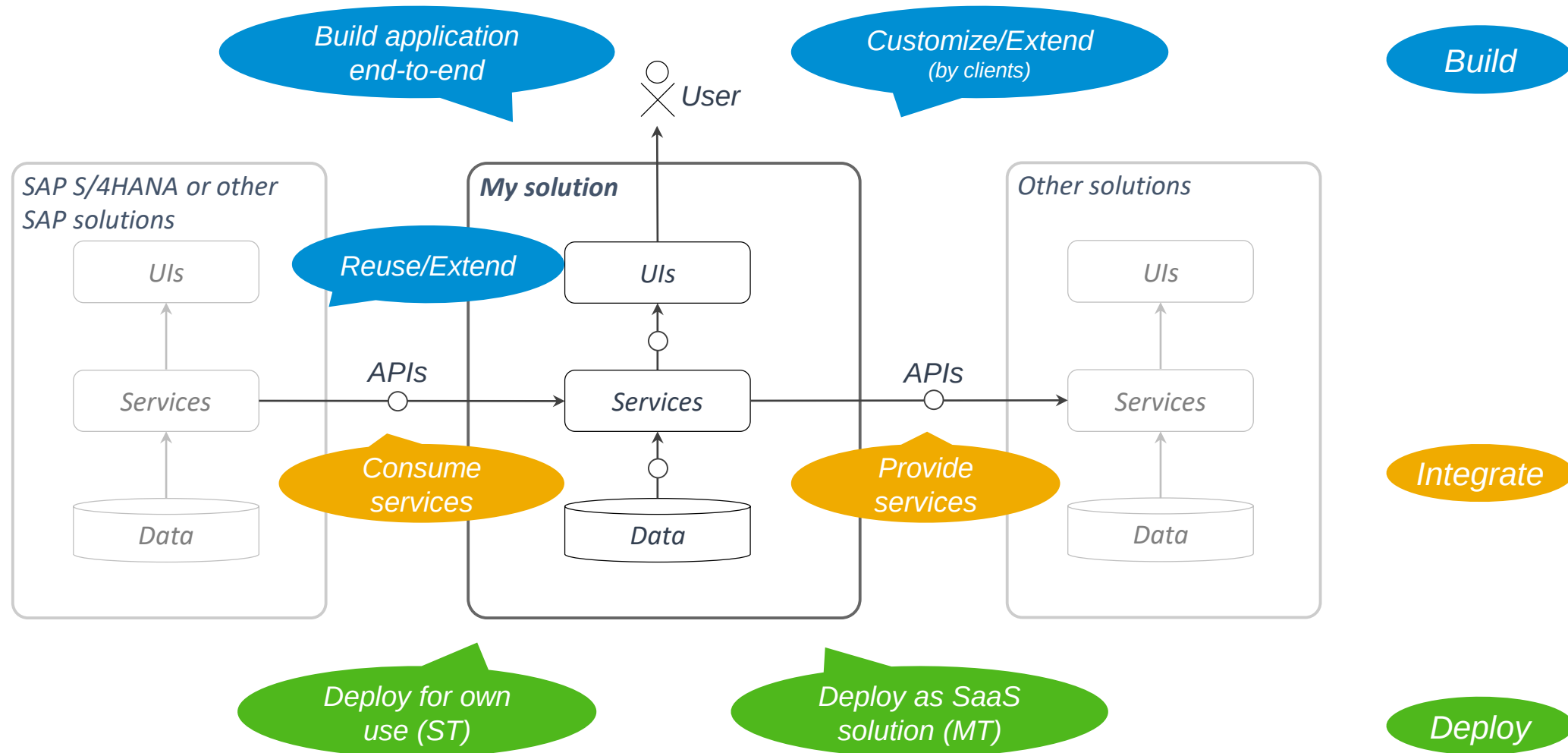
Best practices – basic usage of the programming model



Task	Using	And let the framework take care of...
Define Data Models	CDS	- Mapping to physical schemas (e.g. SQL) - Deployment to SAP HANA + others - Foreign keys and joins - Localized data - Extensibility - Language-embedded data access
Define Services	CDS	- Model translation (e.g. to OData, Open API) - Serving requests out-of-the-box - Standard validations - Auto-filling audit data - Trace and audit logging - Authorization enforcement
Add Custom Logic	Service SDKs for Java or Node.js	- Multitenancy and isolation - Connection pooling - Validating input - Message handling - Serializing responses - Platform integration
Consume Services (e.g. from S/4)		- Providing APIs in Java / Node.js - Mashups with local data - Handling http requests - Resilience, caching, ...
Add UIs	CDS Annotations	- Translating to OData vocabularies - Generating SAP Fiori app skeletons

What Is the New Programming Model?

Use cases



What Is the New Programming Model?

Summary

The **application programming model for SAP Cloud Platform** is SAP's programming model for building enterprise business applications by guiding developers with best practices for full-stack development.



Seamlessly integrates Open Source software and SAP technologies into one consistent **e2e programming model** and development experience



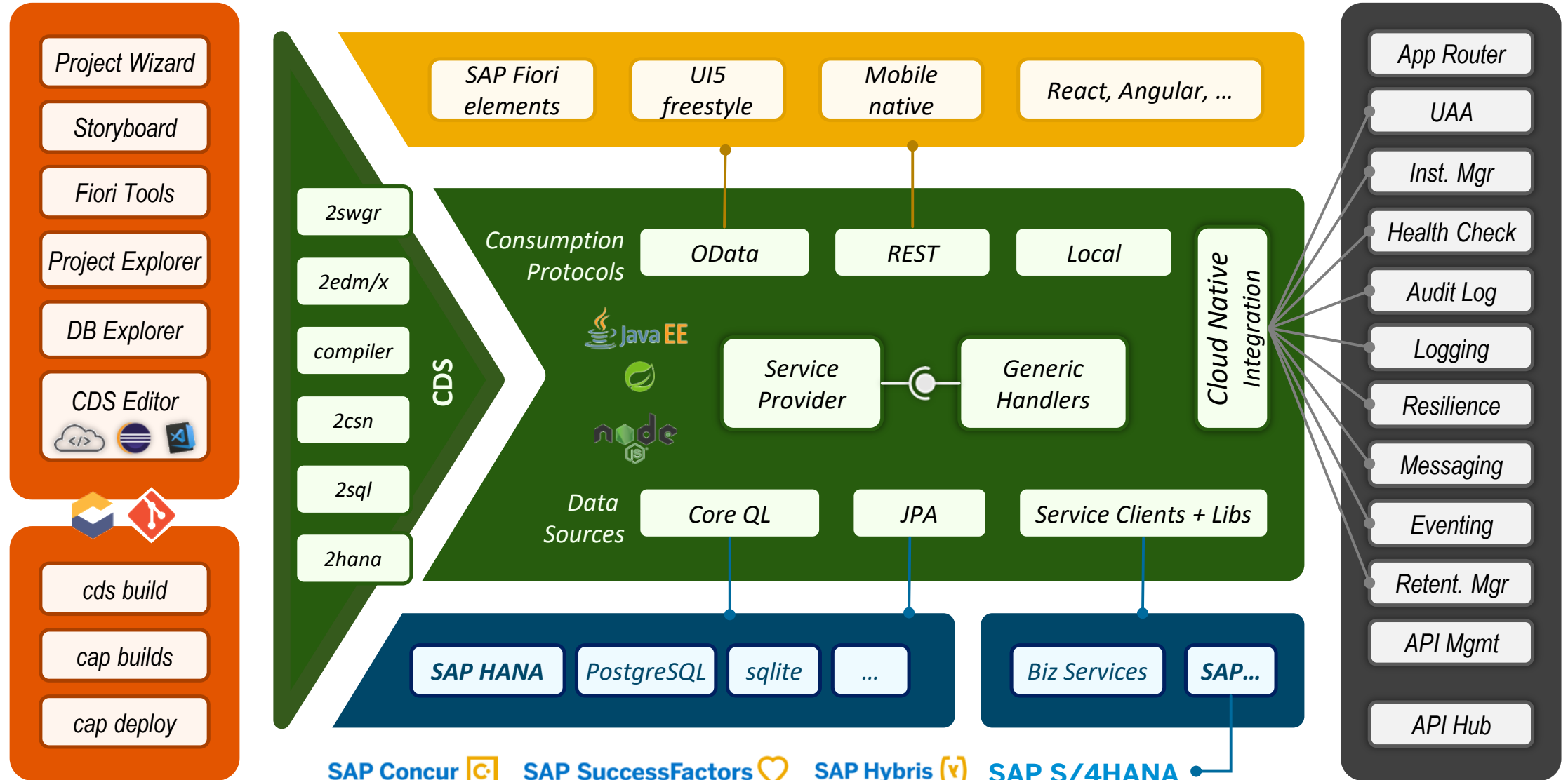
Lightweight and modular implementation of CDS allowing rapid development of persistency, business logic, and UI, tailored for SAP Cloud Platform



Guides developers by providing a set of enterprise-grade tools, languages, and libraries, allowing them to **focus only on their domain logic**

What Is the New Programming Model?

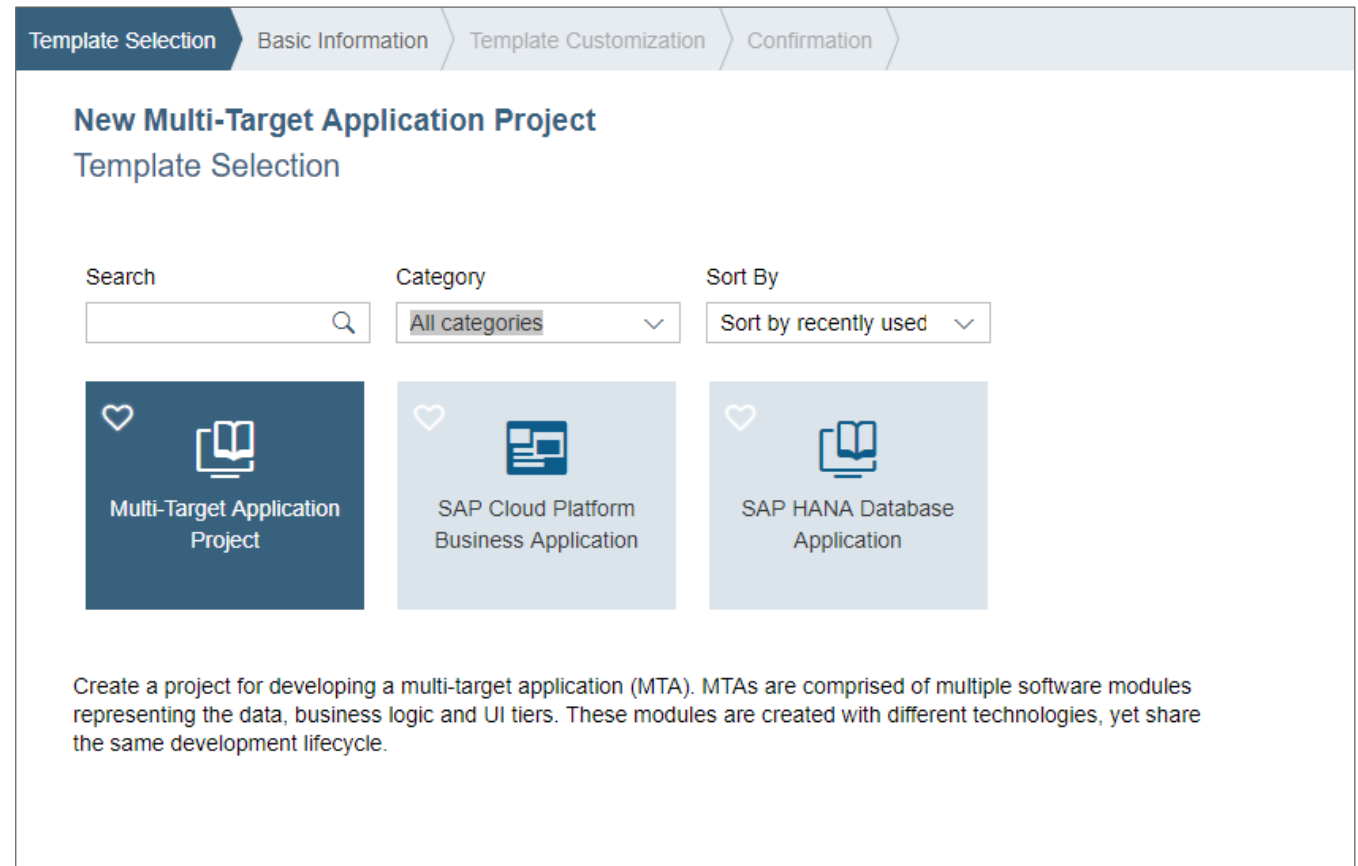
The big picture



What Is the New Programming Model?

New project wizards

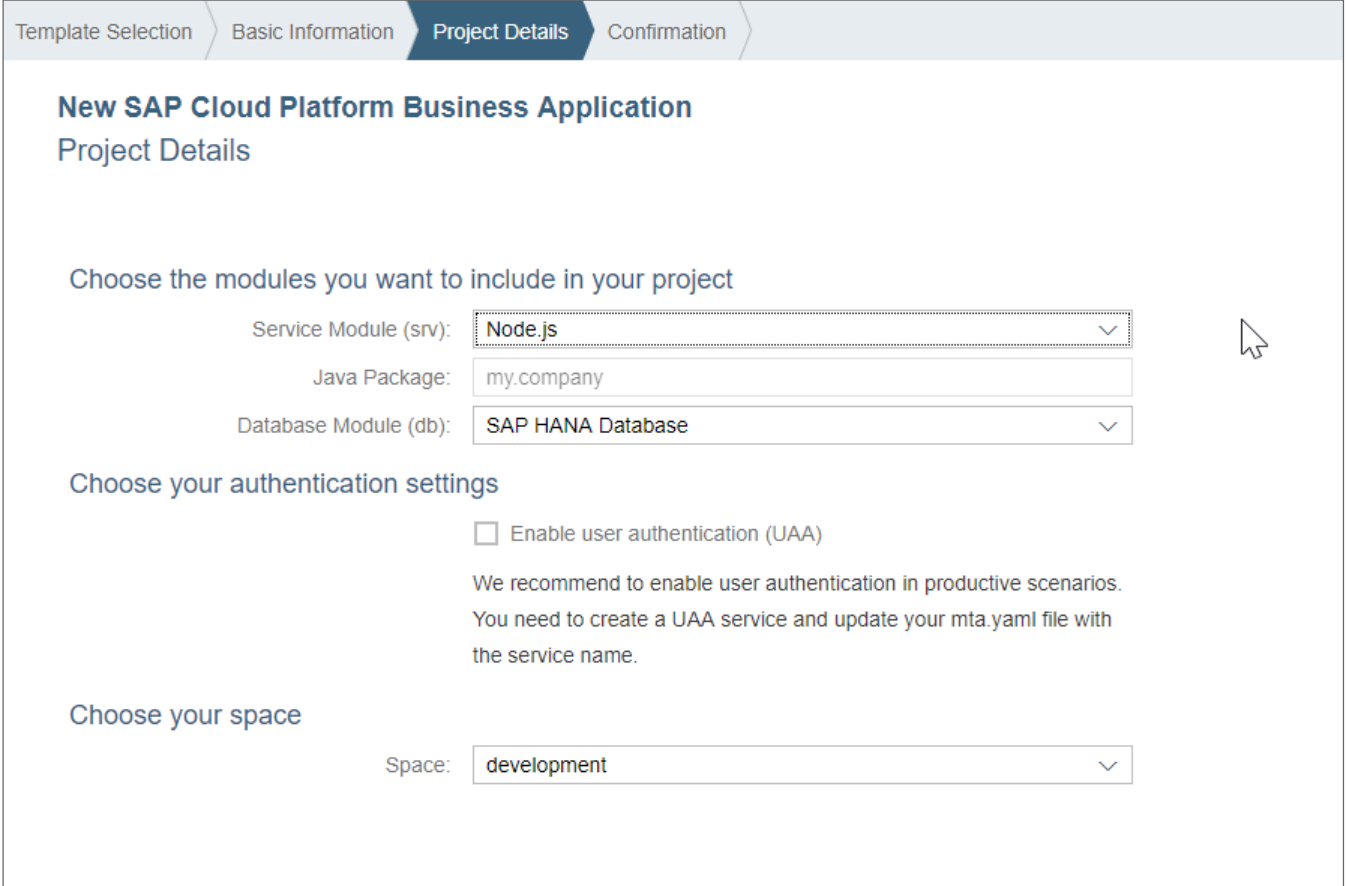
- Two new project wizards
 - SAP HANA Database Application (lightweight/simple project for DB-only development)
 - SAP Cloud Platform Business Application (new programming model for SAP Cloud Platform and on-premise development – see Week 2 for more details)



What Is the New Programming Model?

SAP Cloud Platform Business Application wizard

- New application wizard that creates an MTA project but also initializes it for CDS/new application programming model development
- Supports both Node.js (OData V4) and Java (OData V2) service module creation
- Creates the SAP HANA DB module
- Creates the UAA configuration in the mta.yaml and wiring to generated modules



The screenshot shows the 'Project Details' step of the 'New SAP Cloud Platform Business Application' wizard. The wizard has four steps: Template Selection, Basic Information, Project Details (current), and Confirmation. The 'Project Details' section is titled 'Project Details' and contains three main sections: 'Choose the modules you want to include in your project', 'Choose your authentication settings', and 'Choose your space'. In the first section, 'Service Module (srv)' is set to 'Node.js', 'Java Package' is 'my.company', and 'Database Module (db)' is 'SAP HANA Database'. In the second section, 'Enable user authentication (UAA)' is unchecked, with a note recommending it for productive scenarios. In the third section, 'Space' is set to 'development'.

Template Selection Basic Information **Project Details** Confirmation

New SAP Cloud Platform Business Application

Project Details

Choose the modules you want to include in your project

Service Module (srv):

Java Package:

Database Module (db):

Choose your authentication settings

☐ Enable user authentication (UAA)

We recommend to enable user authentication in productive scenarios.
You need to create a UAA service and update your mta.yaml file with the service name.

Choose your space

Space:

Thank you.

Contact information:

open@sap.com

Follow all of SAP



www.sap.com/contactsap

© 2018 SAP SE or an SAP affiliate company. All rights reserved.

No part of this publication may be reproduced or transmitted in any form or for any purpose without the express permission of SAP SE or an SAP affiliate company.

The information contained herein may be changed without prior notice. Some software products marketed by SAP SE and its distributors contain proprietary software components of other software vendors. National product specifications may vary.

These materials are provided by SAP SE or an SAP affiliate company for informational purposes only, without representation or warranty of any kind, and SAP or its affiliated companies shall not be liable for errors or omissions with respect to the materials. The only warranties for SAP or SAP affiliate company products and services are those that are set forth in the express warranty statements accompanying such products and services, if any. Nothing herein should be construed as constituting an additional warranty.

In particular, SAP SE or its affiliated companies have no obligation to pursue any course of business outlined in this document or any related presentation, or to develop or release any functionality mentioned therein. This document, or any related presentation, and SAP SE's or its affiliated companies' strategy and possible future developments, products, and/or platforms, directions, and functionality are all subject to change and may be changed by SAP SE or its affiliated companies at any time for any reason without notice. The information in this document is not a commitment, promise, or legal obligation to deliver any material, code, or functionality. All forward-looking statements are subject to various risks and uncertainties that could cause actual results to differ materially from expectations. Readers are cautioned not to place undue reliance on these forward-looking statements, and they should not be relied upon in making purchasing decisions.

SAP and other SAP products and services mentioned herein as well as their respective logos are trademarks or registered trademarks of SAP SE (or an SAP affiliate company) in Germany and other countries. All other product and service names mentioned are the trademarks of their respective companies.

See www.sap.com/corporate-en/legal/copyright/index.epx for additional trademark information and notices.