Week 1: Introduction – ABAP RESTful Application Programming Model
Unit 1: The Big Picture

openSAP
open.sap.com

THE BEST RUN SAP

The Big Picture
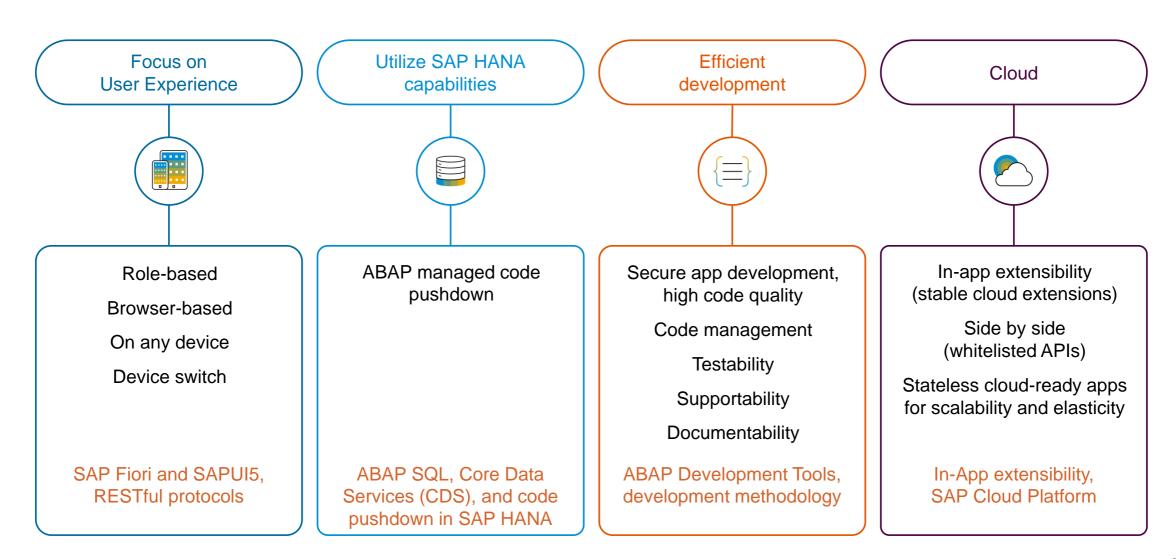**Topics**

01 The Big Picture

02 Architecture Overview

03 Involved Technologies

04 Introducing SAP Cloud Platform, ABAP Environment

05 Preparing Your ABAP Development Environment

Week 1　06 Creating Your First ABAP Cloud Console App

Modern ABAP application development

Focus on User Experience	Utilize SAP HANA capabilities	Efficient development	Cloud

Role-based

Browser-based

On any device

Device switch

SAP Fiori and SAPUI5, RESTful protocols

ABAP managed code pushdown

ABAP SQL, Core Data Services (CDS), and code pushdown in SAP HANA

Secure app development, high code quality

Code management

Testability

Supportability

Documentability

ABAP Development Tools, development methodology

In-app extensibility (stable cloud extensions)

Side by side (whitelisted APIs)

Stateless cloud-ready apps for scalability and elasticity

In-App extensibility, SAP Cloud Platform

The approach

PROVIDE A PROGRAMMING MODEL …

… AS A STRATEGICAL LONG-TERM SOLUTION FOR ABAP DEVELOPMENT

… FOR THE EFFICIENT DEVELOPMENT OF

SAP Fiori apps and Web APIs,
from scratch or **by integrating legacy code**

… OFFERING AN END-TO-END DEVELOPMENT EXPERIENCE WITH

standardized development flow

best practices & development guides

high development **efficiency**

focus on business logic, rather than technical aspects

native **testability**, **documentability**, and **supportability**

code pushdown to SAP HANA

comfortable support for **stateless** and **stateful** environments

… SUPPORTING THE PRODUCT QUALITIES
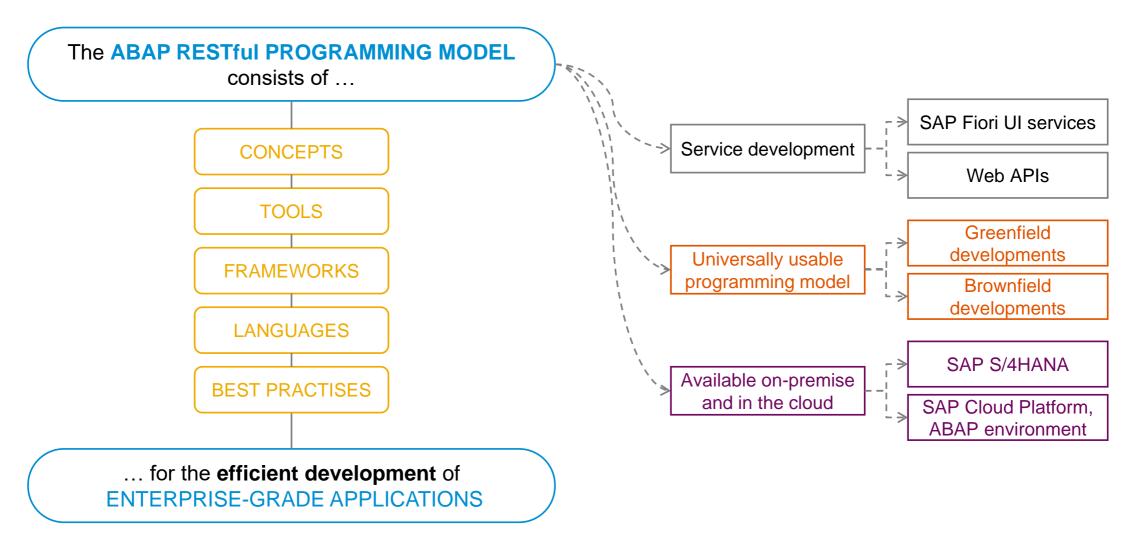
User Experience: SAP Fiori and SAP HANA

Cloud: scalability

Flexibility: break-outs for non-standardized implementations
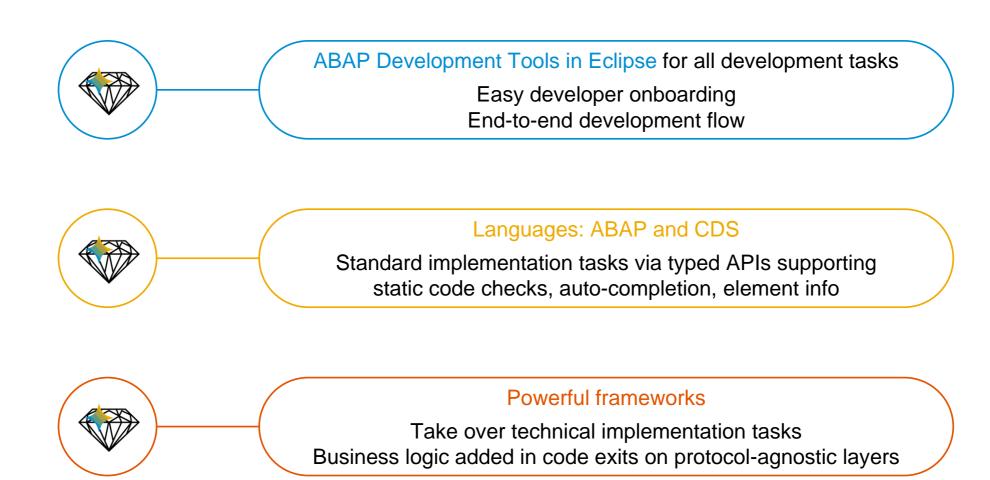
Out-of-the-box extensibility and verticalization

The Big Picture
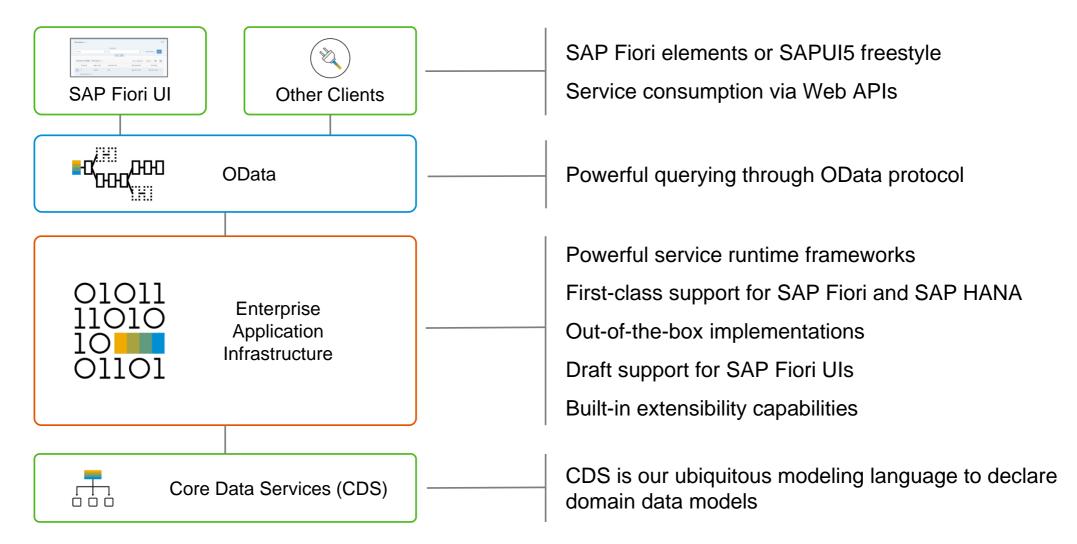# The ABAP RESTful Application Programming Model (RAP)

The **ABAP RESTful PROGRAMMING MODEL** consists of …

- CONCEPTS
- TOOLS
- FRAMEWORKS
- LANGUAGES
- BEST PRACTISES

… for the **efficient development** of ENTERPRISE-GRADE APPLICATIONS

Service development
- SAP Fiori UI services
- Web APIs

Universally usable programming model
- Greenfield developments
- Brownfield developments

Available on-premise and in the cloud
- SAP S/4HANA
- SAP Cloud Platform, ABAP environment

The key players

ABAP Development Tools in Eclipse for all development tasks

Easy developer onboarding
End-to-end development flow

Languages: ABAP and CDS

Standard implementation tasks via typed APIs supporting
static code checks, auto-completion, element info

Powerful frameworks

Take over technical implementation tasks
Business logic added in code exits on protocol-agnostic layers

RAP at a glance

SAP Fiori UI	**Other Clients**	SAP Fiori elements or SAPUI5 freestyle
		Service consumption via Web APIs
OData		Powerful querying through OData protocol
Enterprise Application Infrastructure		Powerful service runtime frameworks
		First-class support for SAP Fiori and SAP HANA
		Out-of-the-box implementations
		Draft support for SAP Fiori UIs
		Built-in extensibility capabilities
Core Data Services (CDS)		CDS is our ubiquitous modeling language to declare domain data models

Evolution towards the ABAP RESTful Application Programming Model (RAP)

ABAP Platform ≤ 7.5

CLASSIC
ABAP PROGRAMMING

Best practice freestyle
ABAP programming

ABAP Platform ≥ 7.5

ABAP PROGRAMMING MODEL
FOR SAP FIORI

SEGW / @OData
Referenced Data Source

Core Data
Services (CDS)

CDS-Based
BOPF

**SAP Cloud Platform, ABAP Environment
SAP S/4HANA ≥ 1909**

ABAP RESTful APPLICATION
PROGRAMMING MODEL

Business Service

Core Data
Services

Behavior Definition
& Implementation

Safe investments!

The Big Picture
Wrap-up

IN THIS UNIT, YOU LEARNED

- About requirements in modern app development
- What the mission of RAP is
- That RAP is the strategic long-term solution for ABAP development
- Which parts make RAP

NEXT UNIT

- Week 1 – Unit 2
  Architecture Overview

Further reading

ABAP RESTful APPLICATION PROGRAMMING MODEL INFORMATION PAGE

For more information, links to documentation, tutorials, and more, please visit the RAP at openSAP information page by following the link below

[RAP at openSAP information page (week 1)](#)

Thank you.

Contact information:

open@sap.com

Follow all of SAP

THE BEST RUN SAP

Week 1: Introduction – ABAP RESTful Application Programming Model
Unit 2: Architecture Overview
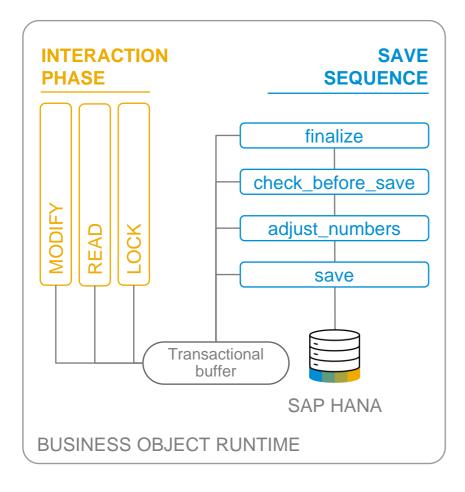
openSAP
open.sap.com

THE BEST RUN SAP

Topics

Week 1

BUSINESS OBJECTS

What is a business object?

Business object runtime implementation types

UNMANAGED

For brownfield developments with available application code for interaction phase, transactional buffer, and save sequence
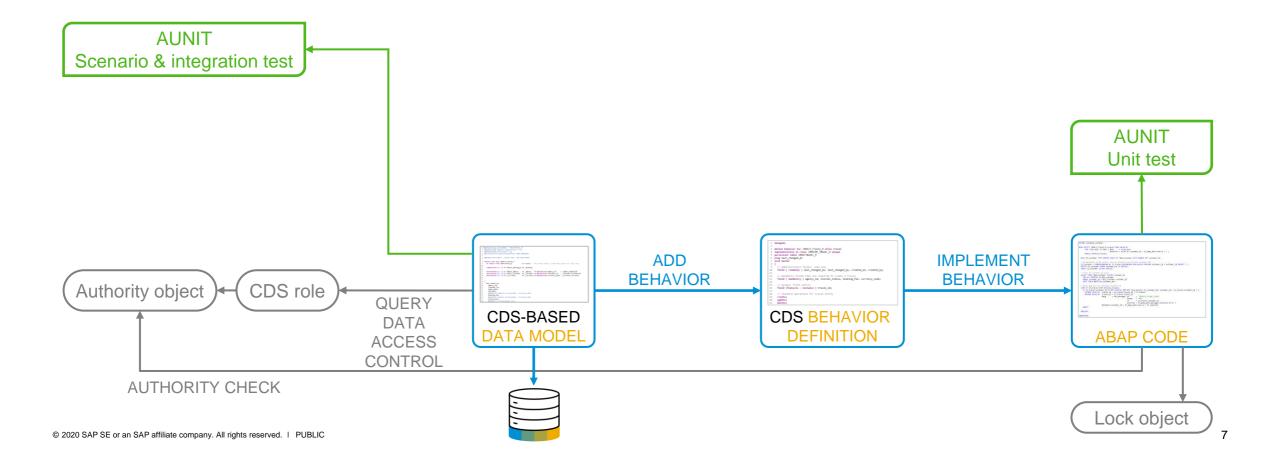
➔ Developers in charge of BO runtime: CRUD operations
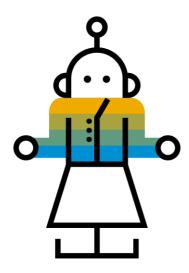
➔ Adapters needed to integrate the existing code

MANAGED

For greenfield developments with standard implementation
(Opt. unmanaged appl. components: DB tables, lock/PFCG object, update task FM)

➔ Standard CRUD operations work out-of-the-box
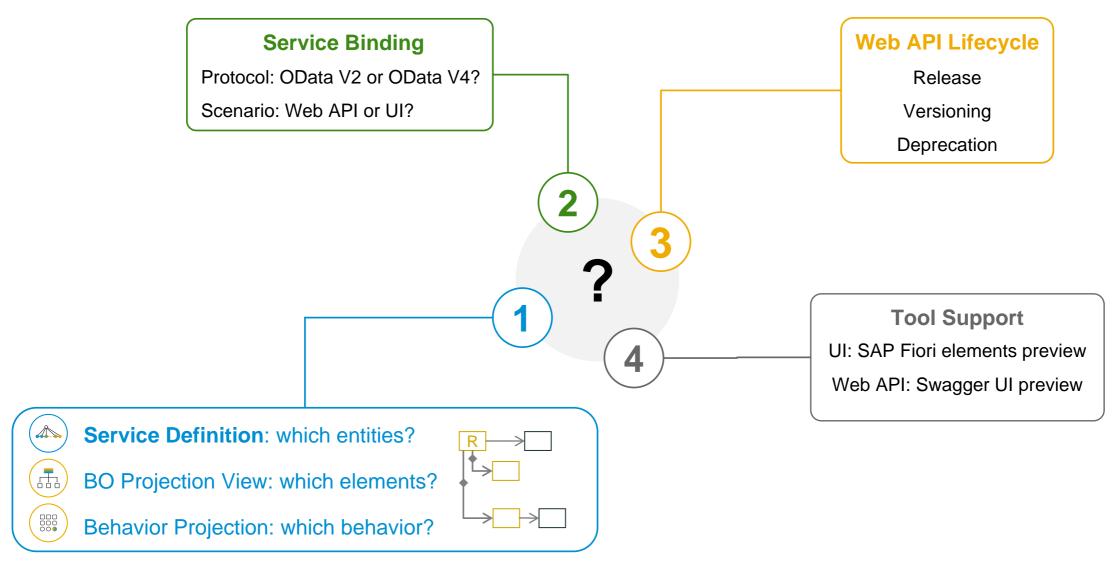
➔ Developers add BO-specific business logic
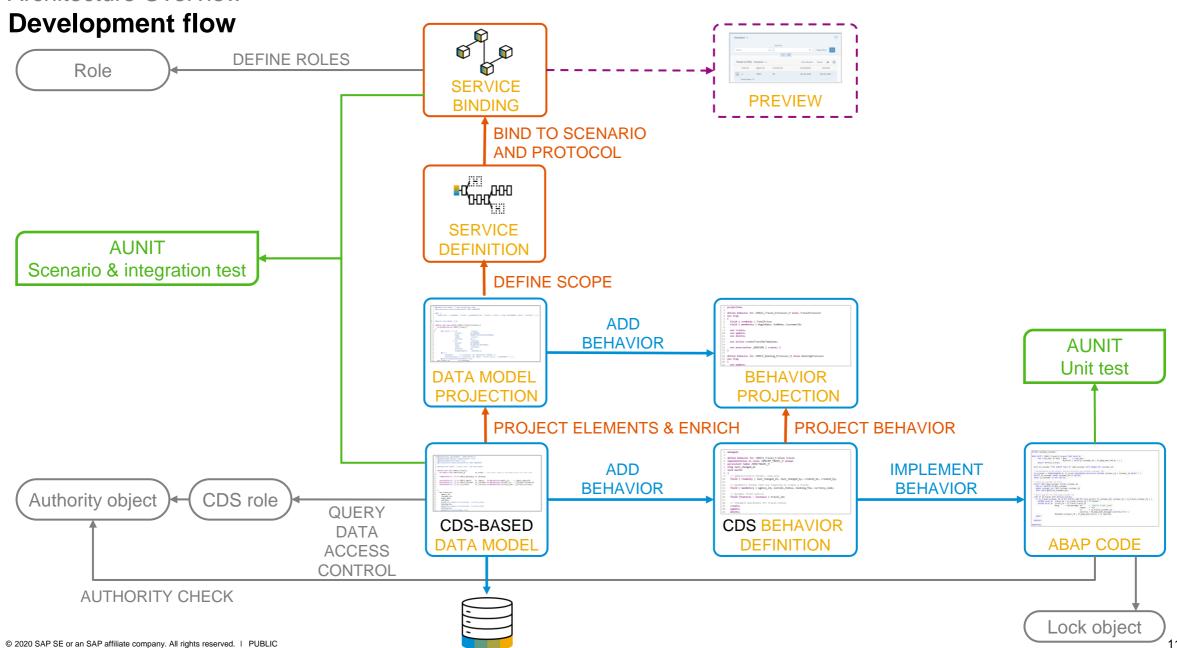
Development flow

BUSINESS SERVICES

What is a business service?

Service Binding

Protocol: OData V2 or OData V4?

Scenario: Web API or UI?

Web API Lifecycle

Release

Versioning

Deprecation

2

3

?

1

4

Tool Support

UI: SAP Fiori elements preview

Web API: Swagger UI preview

Service Definition: which entities?

BO Projection View: which elements?

Behavior Projection: which behavior?

R

SAP Fiori UI and Web API consumption

	SAP FIORI UI	Web API
UI annotation exposure	Required	Not required
Value help	Required	Not required
Dynamic feature control	Required	Not required
Draft	Required	Not required
Content IDs for complex $batch requests	Not required	Required
Release, versioning, and deprecation	Not required	Required

Development flow

Role

DEFINE ROLES

SERVICE BINDING

PREVIEW

BIND TO SCENARIO AND PROTOCOL

SERVICE DEFINITION

AUNIT
Scenario & integration test

DEFINE SCOPE

DATA MODEL PROJECTION

ADD BEHAVIOR

BEHAVIOR PROJECTION

AUNIT
Unit test

PROJECT ELEMENTS & ENRICH

PROJECT BEHAVIOR

CDS-BASED DATA MODEL

ADD BEHAVIOR

CDS BEHAVIOR DEFINITION

IMPLEMENT BEHAVIOR

ABAP CODE

Authority object

CDS role

QUERY DATA ACCESS CONTROL

AUTHORITY CHECK

Lock object

11

Wrap-up

IN THIS UNIT, YOU LEARNED

- About the architecture of the ABAP RESTful Application Programming Model
- What the vital parts of RAP are
- How business objects and business service fit in the picture
- About development types and development flow for RAP apps

NEXT UNIT

- Week 1 – Unit 3
  Involved Technologies
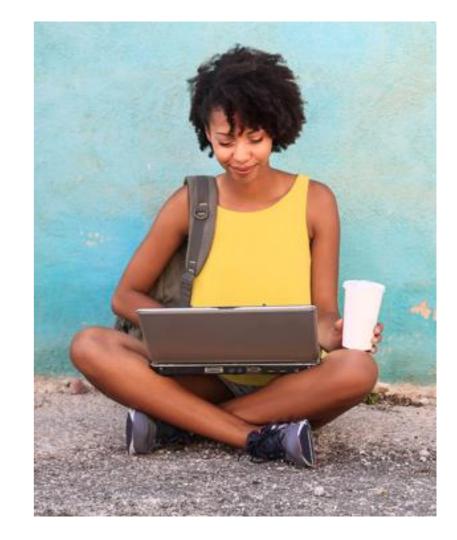
Further reading

ABAP RESTful APPLICATION PROGRAMMING MODEL INFORMATION PAGE

For more information, links to documentation, tutorials, and more, please visit the RAP at openSAP information page by following the link below

RAP at openSAP information page (week 1)

Thank you.

Contact information:

open@sap.com

THE BEST RUN SAP

Follow all of SAP

www.sap.com/contactsap

THE BEST RUN SAP

Week 1: Introduction – ABAP RESTful Application Programming Model
Unit 3: Involved Technologies

openSAP
open.sap.com

THE BEST RUN SAP

Involved Technologies
Topics

01	The Big Picture
02	Architecture Overview
03	Involved Technologies
04	Introducing SAP Cloud Platform, ABAP Environment
05	Preparing Your ABAP Development Environment
06	Creating Your First ABAP Cloud Console App

Week 1

ABAP development for SAP HANA

KEY TECHNOLOGY ASPECTS OF SAP HANA

IN-MEMORY COMPUTING
Allowing OLTP and OLAP in real time

ROW AND COLUMN-BASED DATA STORE
Very fast aggregation and search

HIGH DATA COMPRESSION
Make use of real-life / sparse fill of tables

MULTICORE ARCHITECTURE SUPPORT
Benefit from massive parallelization

KEY IMPACTS ON ABAP APP DEVELOPMENT

CODE PUSHDOWN
Delegation of data computing to the DB

NO AGGREGATES
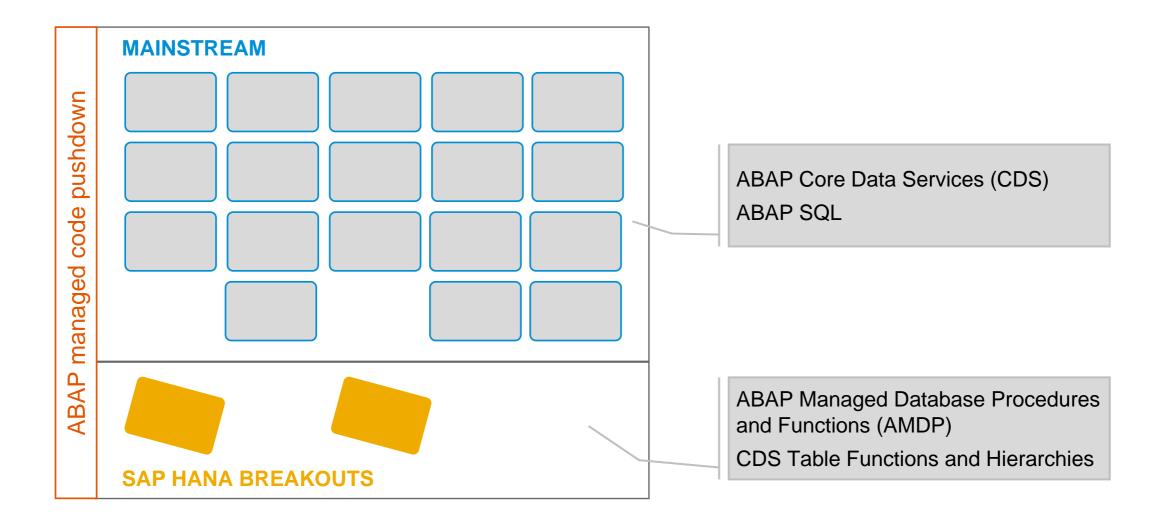On-the-fly data models without duplicates

FEWER INDICES
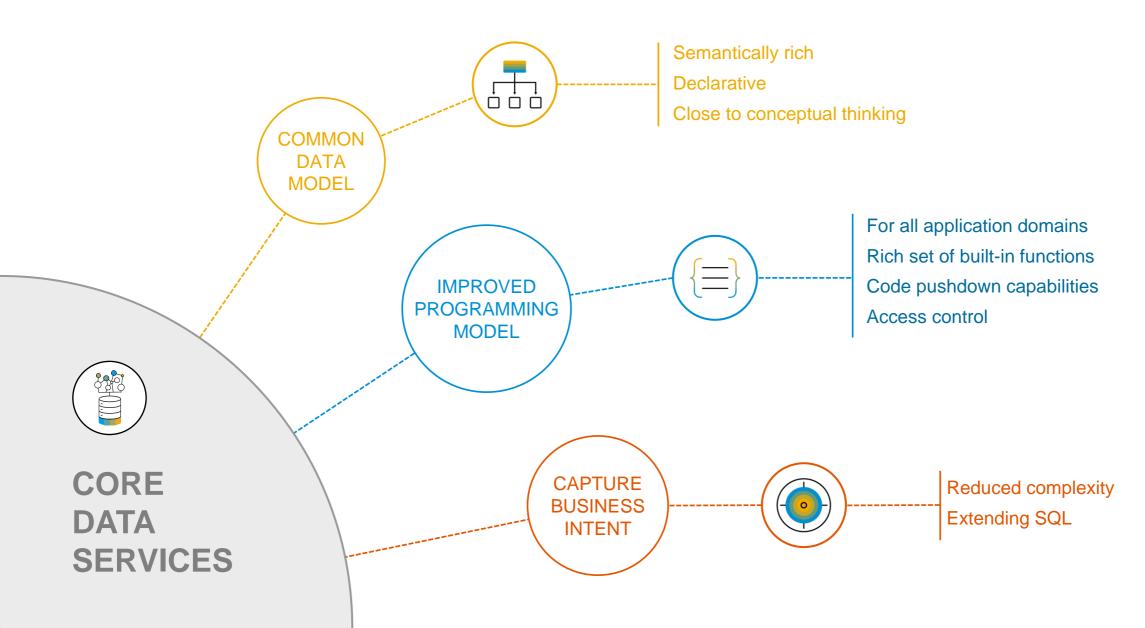Flexible and fast retrieval of the dataset

FEWER CODE LINES
Less complexity in data models and code

ABAP development for SAP HANA: Mainstream and code breakouts

MAINSTREAM

ABAP managed code pushdown

SAP HANA BREAKOUTS

ABAP Core Data Services (CDS)

ABAP SQL

ABAP Managed Database Procedures and Functions (AMDP)

CDS Table Functions and Hierarchies

4

ABAP Core Data Services (CDS) – Next-generation data modeling

COMMON DATA MODEL

- Semantically rich
- Declarative
- Close to conceptual thinking

IMPROVED PROGRAMMING MODEL

- For all application domains
- Rich set of built-in functions
- Code pushdown capabilities
- Access control

CORE DATA SERVICES

CAPTURE BUSINESS INTENT

- Reduced complexity
- Extending SQL

ABAP Core Data Services (CDS) – Combine the best of both worlds

VIEW BUILDING

SQL views with restricted feature set for SAP HANA capabilities

LIFECYCLE MANAGEMENT

ABAP lifecycle management e.g. transports

ABAP type system

VIEW BUILDING

Specialized views (analytical views, attribute views, calculated views, …)

LIFECYCLE MANAGEMENT

Non-ABAP lifecycle management, e.g. transports

Non-ABAP type system

ABAP AS

SAP HANA

ABAP CDS

VIEW BUILDING

Advanced data modeling

Code pushdown with various built-in SQL functions and expressions

Code breakouts to utilize native SAP HANA features and capabilities (CDS table functions)

LIFECYCLE MANAGEMENT

Proven and consistent **ABAP** lifecycle management, e.g. transports

ABAP type system

Consumption in ABAP SQL and ABAP frameworks

ABAP Core Data Services (CDS) – Overview of supported CDS modelling entity types

CATEGORY	CDS ENTITY TYPE	ABAP CDS STATEMENT

Standard view building	CDS View Entity
	CDS Projection View
	CDS DDIC-based View
		`DEFINE VIEW ENTITY`
		`DEFINE VIEW ENTITY AS PROJECTION`
		`DEFINE VIEW`

Transactional RAP BO behaviour definition	CDS Behavior Definition	`DEFINE BEHAVIOR`
Modification-free extension	CDS Metadata Extension CDS Views (Entity) Extension	`ANNOTATE VIEW` `EXTEND VIEW (ENTITY)`
SAP HANA breakout	CDS Table Function CDS Hierarchy	`DEFINE TABLE FUNCTION` `DEFINE HIERARCHY`
External implementation	CDS Custom Entity CDS Abstract Entity	`DEFINE CUSTOM ENTITY` `DEFINE ABSTRACT ENTITY`
Access control	CDS Role	`DEFINE ROLE`

CDS in the ABAP RESTful Application Programming Model

SERVICE CONSUMPTION

SAP FIORI UI
Consume OData UI services

WEB API
Consume OData Web APIs

BUSINESS SERVICES PROVISIONING

SERVICE BINDING – Bind to protocol version and scenario

SERVICE DEFINITION – Define scope to be exposed

BUSINESS OBJECT PROJECTION
- CDS: Projection views
- BDEF: Behavior projection
- ABAP: Behavior implementation

DATA MODELING & BEHAVIOR

BUSINESS OBJECTS
- CDS: Data modeling
- BDEF: Behavior definition
- ABAP: Behavior implementation

QUERIES
- CDS: Data modeling

Declarative and expression-oriented ABAP language

MODERN
ABAP

Simple and concise ABAP code through new language features like inline declarations, constructor expressions

Extensively expression-oriented syntax

Advanced table operations like CORRESPONDING( ) operator, grouping and filtering

Entity Manipulation Language (EML) to control the transactional business object behavior in the RAP context

JSON support in sXML library

Inline code documentation with ABAP Doc

ABAP Unit Testing with test doubles and test seams

Modern ABAP & EML in the ABAP RESTful Application Programming Model

SERVICE CONSUMPTION

SAP FIORI UI	WEB API
Consume OData UI services	Consume OData Web APIs

BUSINESS SERVICES PROVISIONING

SERVICE BINDING – Bind to protocol version and scenario

SERVICE DEFINITION – Define scope to be exposed

BUSINESS OBJECT PROJECTION

- CDS: Projection views
- BDEF: Behavior projection
- ABAP: Behavior implementation

DATA MODELING & BEHAVIOR

BUSINESS OBJECTS

- CDS: Data modeling
- BDEF: Behavior definition
- ABAP: Behavior implementation

QUERIES

- CDS: Data modeling

The Open Data Protocol (OData) as service protocol for simplified access to SAP data

ODATA MAKES DATA FROM BACKEND SYSTEMS AVAILABLE
to engaging user interfaces and Web APIs in a standardized way

ODATA IS USED IN

SAP Fiori & SAP Fiori elements

SAPUI5

SAP S/4HANA

SAP Cloud Platform

SAP Mobile Platform 3.0

SAP Gateway

…

ODATA IS EXTENSIBLE

Allows addition of features required when building easy-to-use applications, both mobile and browser-based

ODATA IS BASED ON INDUSTRY STANDARDS

HTTP, XML, Atom, JSON

REST-based architecture

Driven by Microsoft, SAP, and IBM

OASIS standard (OData V.4 and OData JSON Format 4.0)

OData protocol in the ABAP RESTful Application Programming Model

SERVICE CONSUMPTION

SAP FIORI UI
Consume OData UI services

WEB API
Consume OData Web APIs

SERVICE BINDING – Bind to protocol version and scenario

SERVICE DEFINITION – Define scope to be exposed

BUSINESS SERVICES PROVISIONING

BUSINESS OBJECT PROJECTION

CDS: Projection views

BDEF: Behavior projection

ABAP: Behavior implementation

DATA MODELING & BEHAVIOR

BUSINESS OBJECTS

CDS: Data modeling

BDEF: Behavior definition

ABAP: Behavior implementation

QUERIES
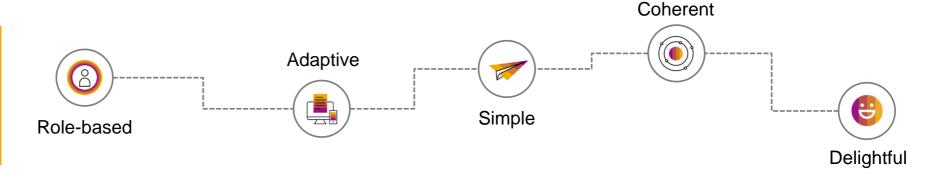
CDS: Data modeling

SAP Fiori user experience (UX) for the Intelligent Suite

Coherent

Adaptive

DESIGN PRINCIPLES

Role-based

Simple

Delightful

UI TECHNOLOGIES

SAPUI5   iOS iOS   Android Android   Other technologies

APP DEVELOPMENT
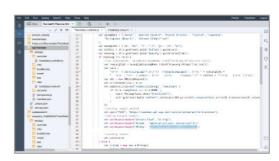
SAP Fiori tools
Productivity tools to make developers more efficient

SAP Fiori elements
Templates for low-code development with SAPUI5

Freestyle SAPUI5 or native mobile apps

Custom apps developed for customers and partners

SAP Fiori in the ABAP RESTful Application Programming Model

SERVICE CONSUMPTION

SAP FIORI UI
Consume OData UI services

WEB API
Consume OData Web APIs

BUSINESS SERVICES PROVISIONING

SERVICE BINDING – Bind to protocol version and scenario

SERVICE DEFINITION – Define scope to be exposed

BUSINESS OBJECT PROJECTION
- CDS: Projection views
- BDEF: Behavior projection
- ABAP: Behavior implementation

DATA MODELING & BEHAVIOR

BUSINESS OBJECTS
- CDS: Data modeling
- BDEF: Behavior definition
- ABAP: Behavior implementation

QUERIES
- CDS: Data modeling

Involved Technologies
Modern ABAP development in Eclipse

HIGH DEVELOPER PRODUCTIVITY WITH
ABAP DEVELOPMENT TOOLS (ADT)

MODERN DEVELOPMENT TOOLSET

Fully Eclipse-based
Syntax check, Code completion
Syntax highlighting, Pretty printing
Navigation, Search, Quick Fixes
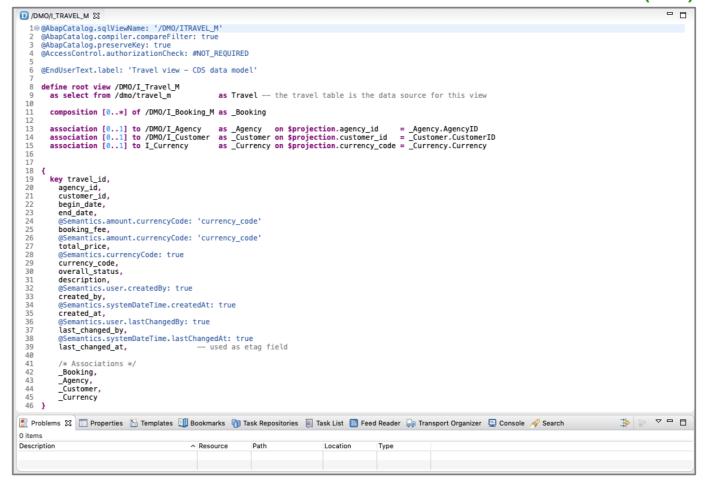
QUALITY ASSURANCE

Static code checks (ATC, CVA) with
remote and local scenarios
Unit testing incl. isolation frameworks
Test seams and injections

SUPPORTABILITY

Debugging, profiling
Static and dynamic logging
Runtime monitoring and analysis

```
/DMO/I_TRAVEL_M

 1  @AbapCatalog.sqlViewName: '/DMO/ITRAVEL_M'
 2  @AbapCatalog.compiler.compareFilter: true
 3  @AbapCatalog.preserveKey: true
 4  @AccessControl.authorizationCheck: #NOT_REQUIRED
 5
 6  @EndUserText.label: 'Travel view – CDS data model'
 7
 8  define root view /DMO/I_Travel_M
 9    as select from /dmo/travel_m          as Travel -- the travel table is the data source for this view
10
11    composition [0..*] of /DMO/I_Booking_M as _Booking
12
13    association [0..1] to /DMO/I_Agency    as _Agency   on $projection.agency_id    = _Agency.AgencyID
14    association [0..1] to /DMO/I_Customer  as _Customer on $projection.customer_id  = _Customer.CustomerID
15    association [0..1] to I_Currency       as _Currency on $projection.currency_code = _Currency.Currency
16
17
18  {
19    key travel_id,
20      agency_id,
21      customer_id,
22      begin_date,
23      end_date,
24      @Semantics.amount.currencyCode: 'currency_code'
25      booking_fee,
26      @Semantics.amount.currencyCode: 'currency_code'
27      total_price,
28      @Semantics.currencyCode: true
29      currency_code,
30      overall_status,
31      description,
32      @Semantics.user.createdBy: true
33      created_by,
34      @Semantics.systemDateTime.createdAt: true
35      created_at,
36      @Semantics.user.lastChangedBy: true
37      last_changed_by,
38      @Semantics.systemDateTime.lastChangedAt: true
39      last_changed_at,              -- used as etag field
40
41      /* Associations */
42      _Booking,
43      _Agency,
44      _Customer,
45      _Currency
46  }
```

Problems | Properties | Templates | Bookmarks | Task Repositories | Task List | Feed Reader | Transport Organizer | Console | Search

0 items

Description	^ Resource	Path	Location	Type

ATC: ABAP Test Cockpit
CVA: Code Vulnerability Analyzer

Wrap-up

IN THIS UNIT, YOU LEARNED

- About the vital technologies that make the
  ABAP RESTful Application Programming Model
  and how these technologies work together

NEXT UNIT

- Week 1 Unit 4
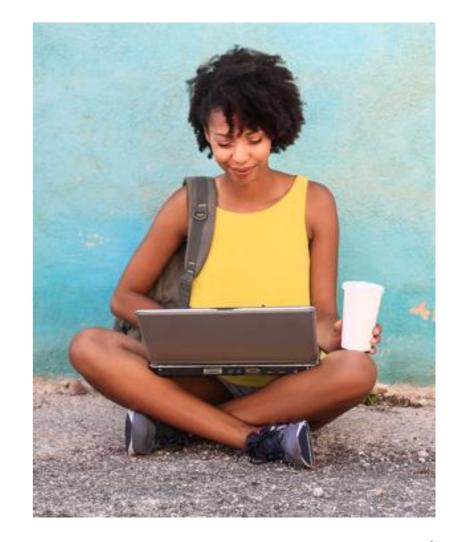 Introducing SAP Cloud Platform, ABAP Environment

Further reading

ABAP RESTful APPLICATION PROGRAMMING MODEL INFORMATION PAGE

For more information, links to documentation, tutorials, and more, please visit the RAP at openSAP information page by following the link below

RAP at openSAP information page (week 1)

Thank you.

Contact information:

open@sap.com

Follow all of SAP

THE BEST RUN SAP

Week 1: Introduction – ABAP RESTful Application Programming Model

Unit 4: Introducing SAP Cloud Platform, ABAP Environment

Topics

01 The Big Picture

02 Architecture Overview

03 Involved Technologies

04 Introducing SAP Cloud Platform, ABAP Environment

05 Preparing Your ABAP Development Environment

Week 1 **06** Creating Your First ABAP Cloud Console App
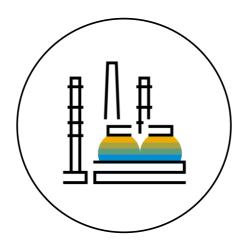
Motivation

Customer Base

Huge **customer and partner base** running ABAP-based solutions with custom code

Enterprise Readiness
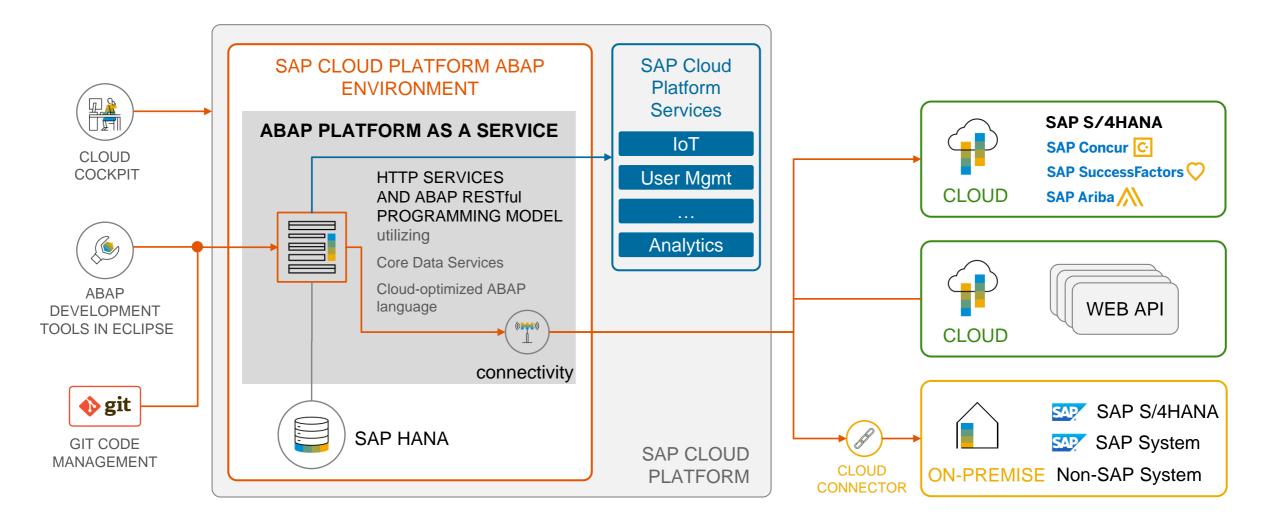
ABAP as proven environment for **enterprise-ready** business applications with competitive cost of development
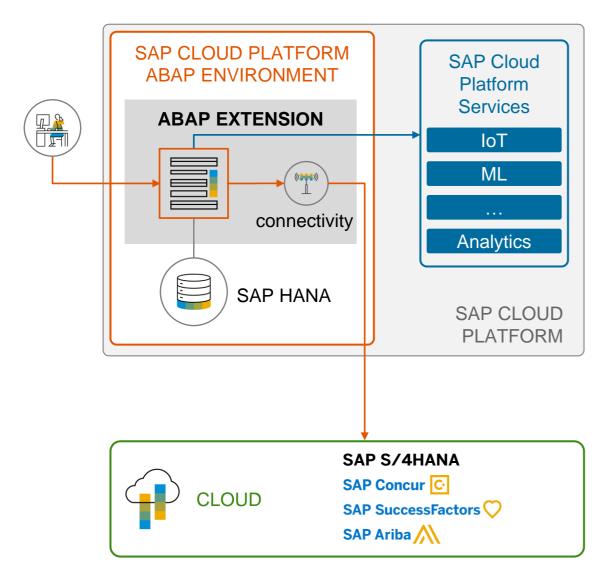
Transition to Cloud

Growing market adoption of **SAP S/4HANA Cloud** with tailored extensibility for customers and partners

The vital parts
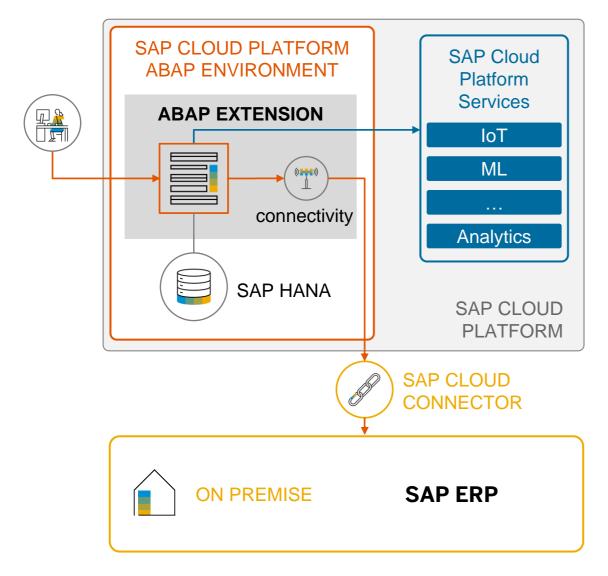
Extension scenario 1: Cloud ERP

SAP CLOUD PLATFORM ABAP ENVIRONMENT

ABAP EXTENSION

connectivity

SAP HANA

SAP Cloud Platform Services

IoT

ML

…

Analytics

SAP CLOUD PLATFORM

CLOUD

SAP S/4HANA

SAP Concur

SAP SuccessFactors

SAP Ariba

Use SAP Cloud Platform ABAP Environment to extend SAP S/4HANA Cloud or other SAP cloud solutions

- SAP Cloud solutions like SAP S/4HANA Cloud provide in-app extensibility to extend SAP apps and processes, but there is no support for classic custom ABAP development on top of SAP S/4HANA Cloud

- SAP Cloud Platform is the foundation to develop and run custom cloud extensions and the ABAP environment shall be used for ABAP based cloud extensions
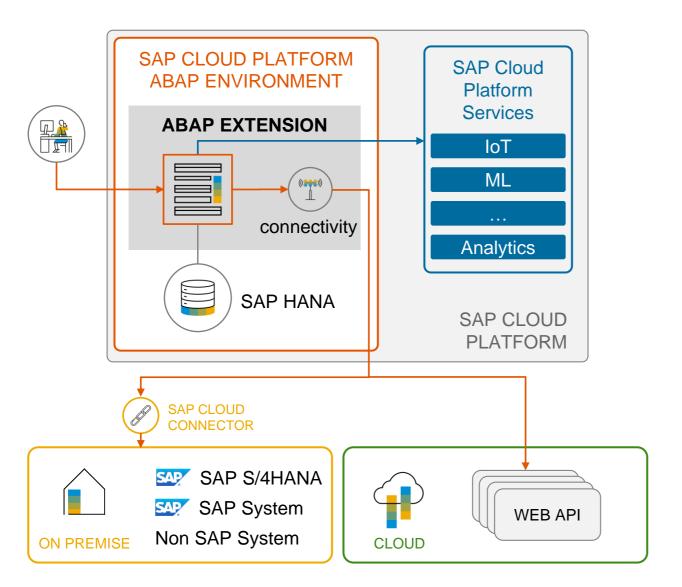
Extension scenario 2: Innovation Platform

Develop and run innovative ABAP apps
on a PaaS in the Cloud

- Benefit from the newest ABAP Platform and
  SAP HANA database technologies independent
  from your existing on-premise system landscape

- Build your Fiori apps with the future proof
  ABAP RESTful Application Programming Model

- Utilize SAP Cloud Platform services like IoT,
  machine learning etc. in your cloud extension

- Delegate operation of the ABAP PaaS and new
  technologies to SAP

Extension scenario 3: Hub-like usage

Decouple ABAP implementations from on-premise core business systems

- **EXTERNAL USER GROUP**
  Make your cloud app available to a broader audience that does not have access to your core business systems (e.g. consumer apps)

- **INTEGRATION HUB**
  Integrate in your cloud extension multiple cloud/on-premise systems with SAP/non-SAP cloud services

- **DATA INTEGRATION**
  Collect data from multiple sources in your cloud extension for further processing and analysis

- **DECOUPLED EXTENSION**
  Cloud extensions use only well defined (remote) APIs of the Business system. This reduces the risk and effort for business system upgrades.

Wrap-up

IN THIS UNIT, YOU LEARNED

- About SAP Cloud Platform, ABAP environment
- What the vital parts are
- What the advantages are
- What the main scenarios are

NEXT UNIT

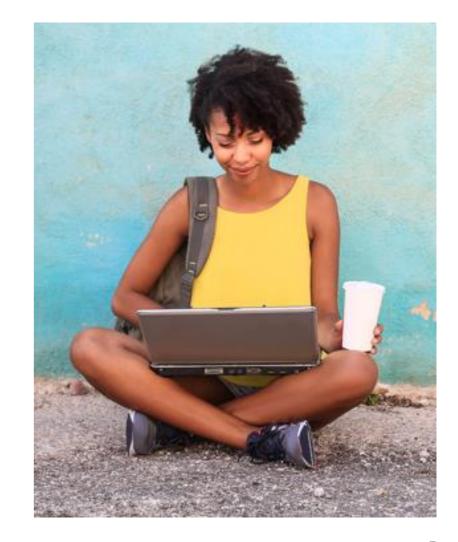- Week 1 Unit 5
  Preparing Your ABAP Development Environment

Further reading

ABAP RESTful APPLICATION PROGRAMMING MODEL INFORMATION PAGE

For more information, links to documentation, tutorials, and more, please visit the RAP at openSAP information page by following the link below

RAP at openSAP information page (week 1)

Thank you.

Contact information:

open@sap.com

Follow all of SAP

THE BEST RUN SAP

Week 1: Introduction – ABAP RESTful Application Programming Model

Unit 5: Preparing Your ABAP Development Environment

openSAP
open.sap.com

THE BEST RUN SAP

Preparing Your ABAP Development Environment
Topics

01	The Big Picture
02	Architecture Overview
03	Involved Technologies
04	Introducing SAP Cloud Platform, ABAP Environment
05	Preparing Your ABAP Development Environment
06	Creating Your First ABAP Cloud Console App

Week 1
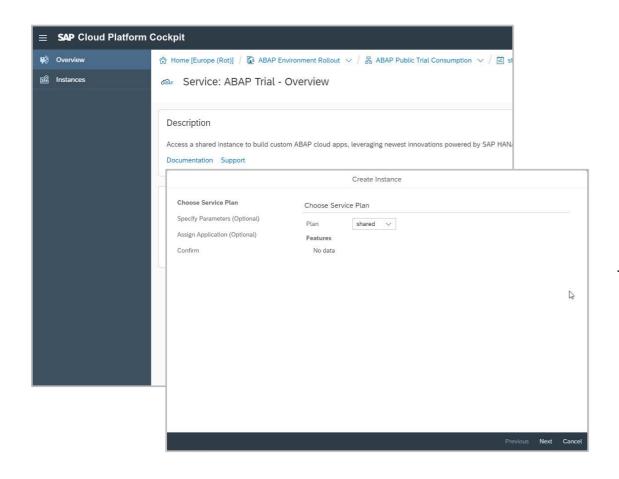
Getting an SAP Cloud Platform, ABAP environment trial account

Access to trial system

Free of charge

Via SAP Cloud Platform trial access
https://cockpit.hanatrial.ondemand.com/

Trial account expires after 30 days, but can be extended to a maximum of 90 days

System lifetime is 4 months

Rules of the game

Shared Trial Offering
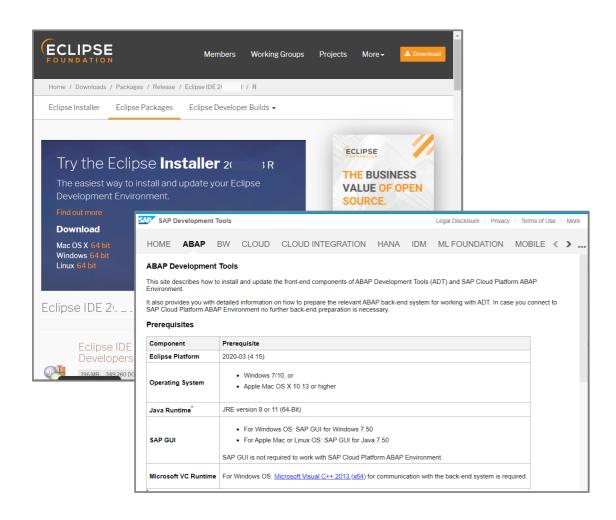
For educational purposes only
No service-level agreement with regard to platform availability

Trial users share the same backend system

No content separation between different users

Installing and connecting ADT to the ABAP environment trial system
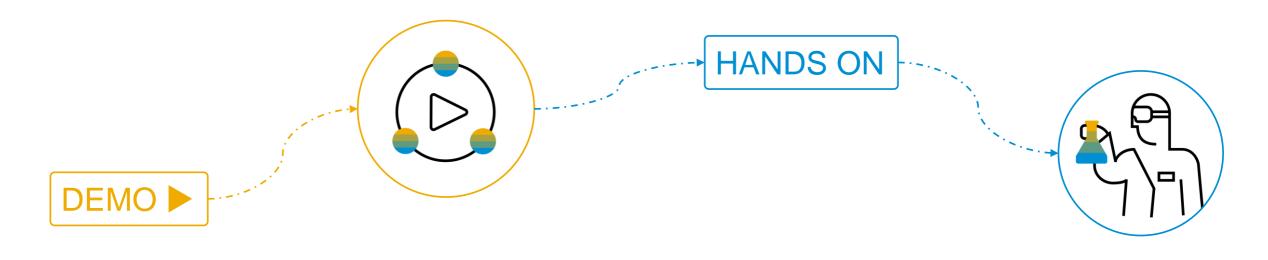
Access to ABAP tools

Instructions about the local installation of Eclipse and ADT available on the central update site for all Eclipse-based SAP development tools

https://tools.hana.ondemand.com/#abap

Demo

PREPARING YOUR ABAP DEVELOPMENT ENVIRONMENT

1. Onboarding to the SAP Cloud Platform, ABAP environment trial

2. Local installation of Eclipse platform and ABAP Development Tools

3. Connection to the trial system

Wrap-up

IN THIS UNIT, YOU LEARNED

- About setting up your development environment
- How to get access to the SAP Cloud Platform trial account
- How to install ABAP Development Tools
- How to connect to the trial system

NEXT UNIT

- Week 1 – Unit 6
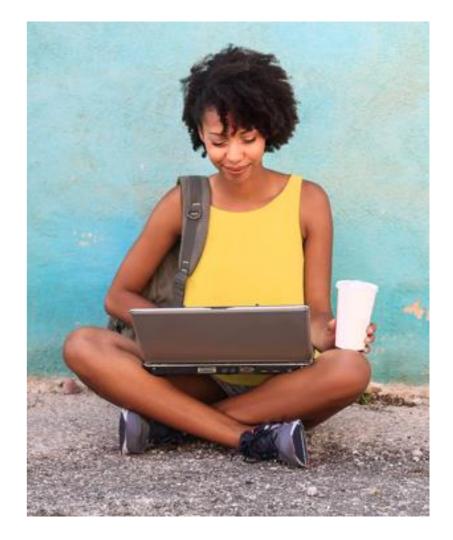 Creating Your First ABAP Cloud Console App

Further reading

ABAP RESTful APPLICATION PROGRAMMING MODEL INFORMATION PAGE

For more information, links to documentation, tutorials, and more, please visit the RAP at openSAP information page by following the link below

RAP at openSAP information page (week 1)

Thank you.

Contact information:

open@sap.com

Follow all of SAP

THE BEST RUN SAP

Week 1: Introduction – ABAP RESTful Application Programming Model
Unit 6: Creating Your First ABAP Cloud Console App

openSAP
open.sap.com

THE BEST RUN SAP

Topics

01 The Big Picture

02 Architecture Overview

03 Involved Technologies

04 Introducing SAP Cloud Platform, ABAP Environment

05 Preparing Your ABAP Development Environment

Week 1

06 Creating Your First ABAP Cloud Console App
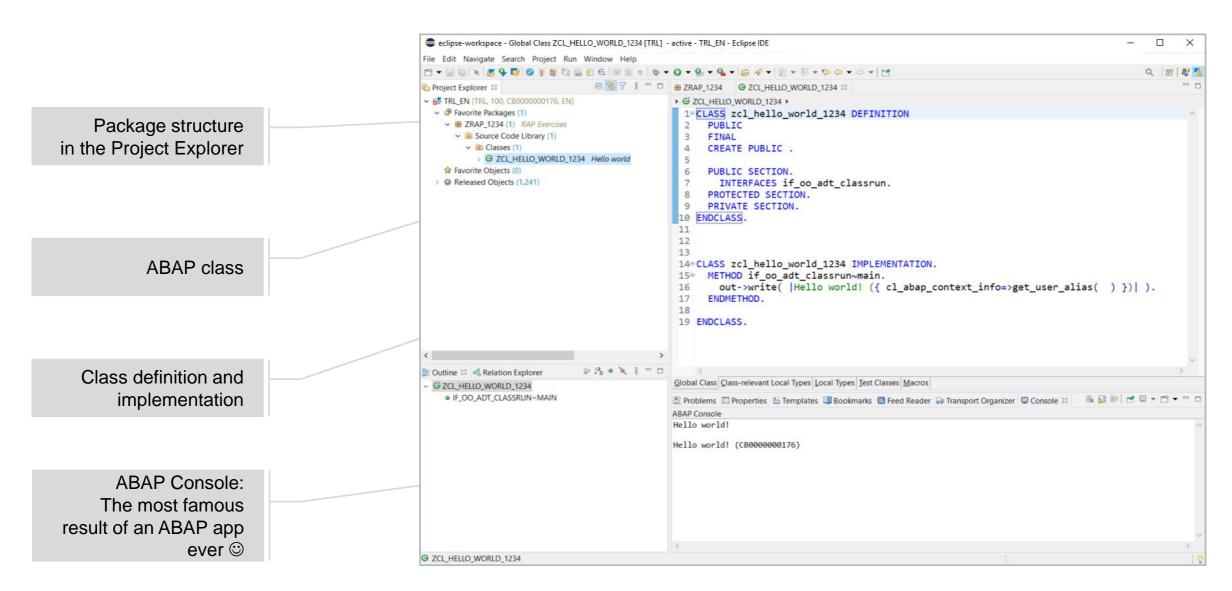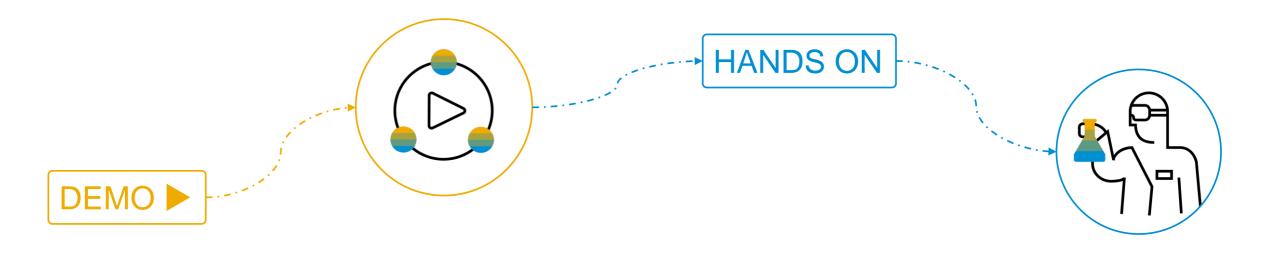
Creating Your First ABAP Cloud Console App
Getting started

Package structure in the Project Explorer

ABAP class

Class definition and implementation

ABAP Console: The most famous result of an ABAP app ever ☺

Getting started

DEMO ▶

HANDS ON

CREATE YOUR FIRST 'HELLO WORLD!' CONSOLE APP

---

1. Create your course ABAP package

2. Create a HelloWorld ABAP class

3. Activate and run your first console app

Wrap-up

IN THIS UNIT, YOU LEARNED

- How to create an ABAP 'Hello World!' console app
- How to use ABAP Development Tools
- How to create ABAP packages and classes
- How to activate and run a program in ABAP

NEXT UNIT

- Week 2
  Developing a Read-Only List Report App

Further reading

ABAP RESTful APPLICATION PROGRAMMING MODEL INFORMATION PAGE

For more information, links to documentation, tutorials, and more, please visit the RAP at openSAP information page by following the link below

RAP at openSAP information page (week 1)

Thank you.

Contact information:

open@sap.com

Follow all of SAP

THE BEST RUN **SAP**