SAP Inside Track BA

Modern ABAP Development Using BOPF

Agustin Pighin, SOVANTA AG
July 28, 2017

PUBLIC

SAP Run Simple

sovanta simplicity first

Agenda

Introduction

Product Qualities

State-Less Applications

Architecture

Hands-On example

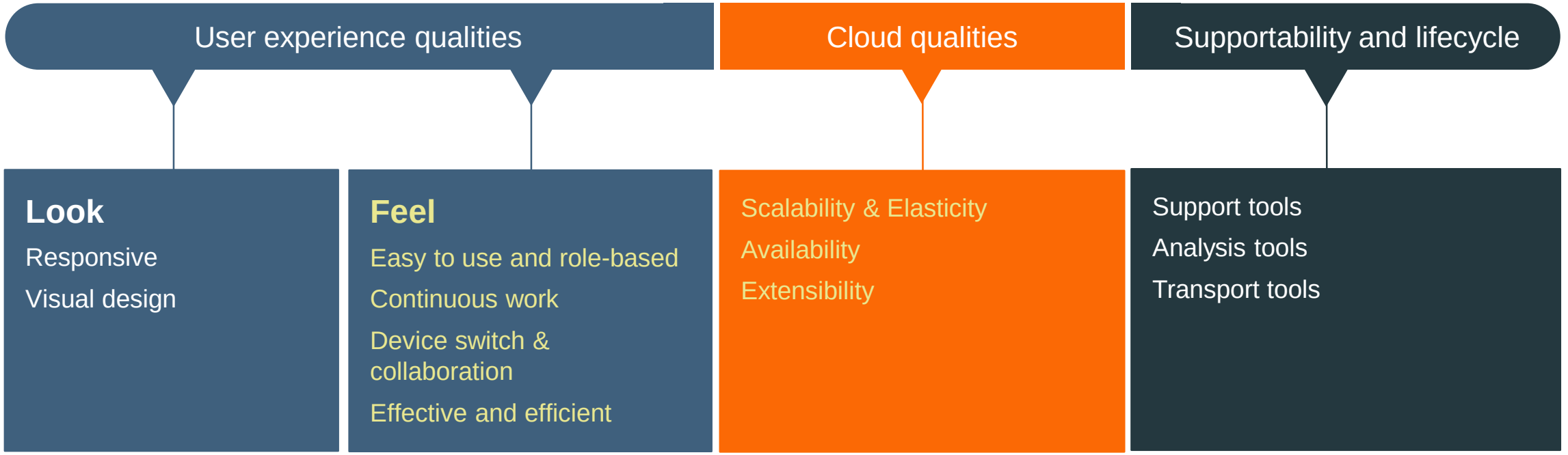
01

Product Qualities

An Overview

Product qualities

Expectations on modern applications



User experience qualities
Easy to use and role-based

Requirement I want an easy to use application providing only relevant functionality for the tasks of my role

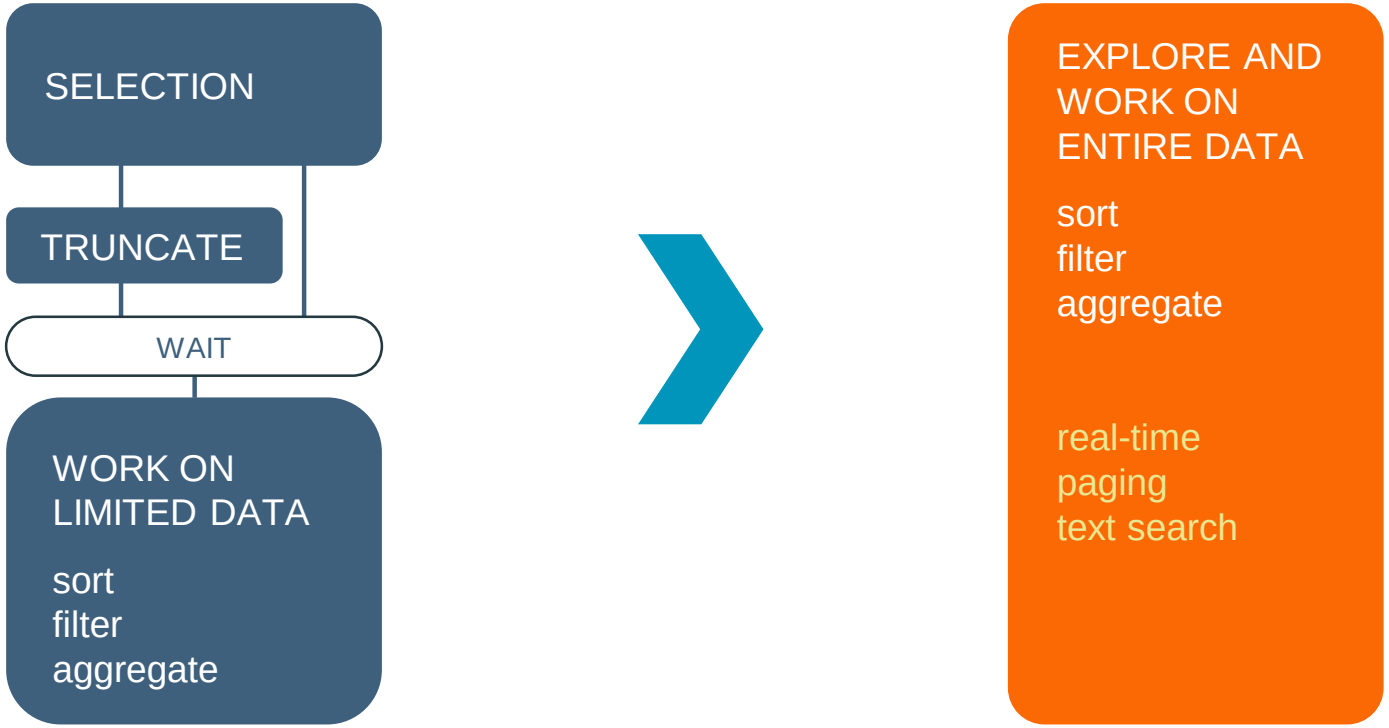
Impact on programming model

Efficient programming model to support multiple applications on same data and functionality
Manage redundancies in applications without inconsistencies

User experience qualities

Effective and efficient work on entirety of data

Requirement I want the freedom to explore and work with entire data – not only on restricted or truncated data



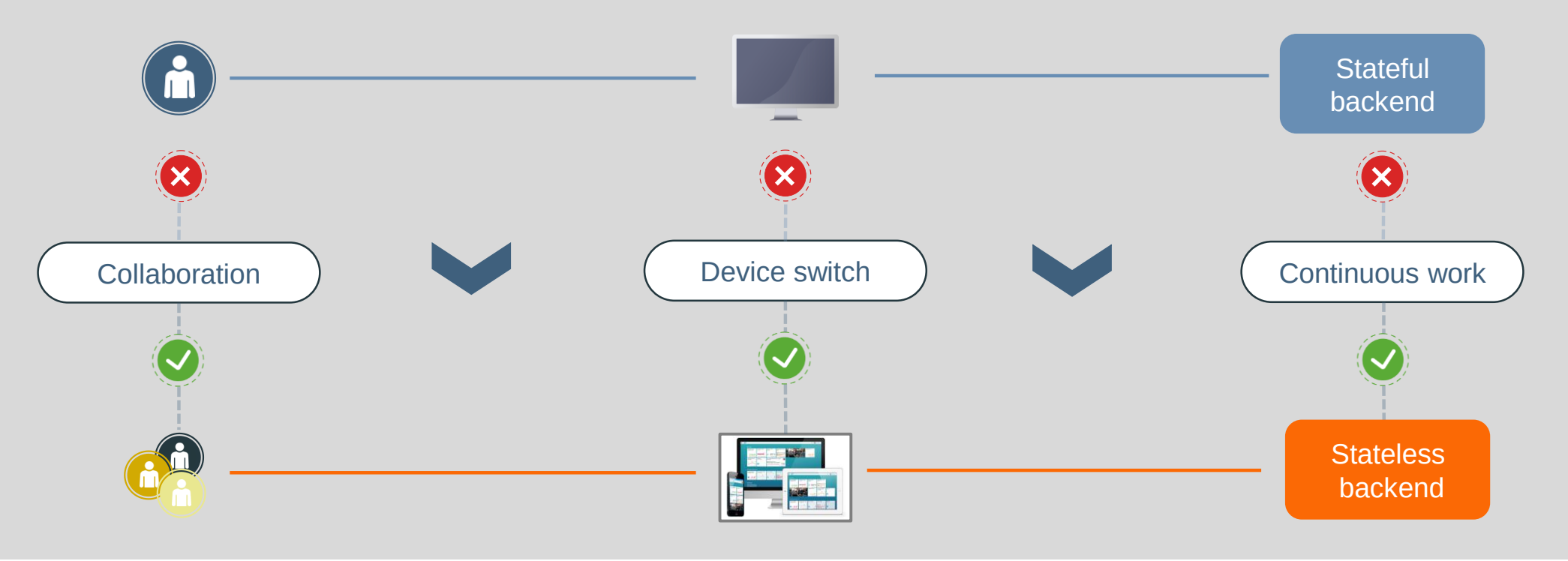
Impact on programming model

Easily enable navigation, filtering, text search and grouping of data for all applications
Provide intrinsic support for exploiting SAP HANA features and performance

User experience qualities

Continuous work, device switch and collaboration

Requirement I want to work from everywhere, continue on interrupted work and change my devices when needed



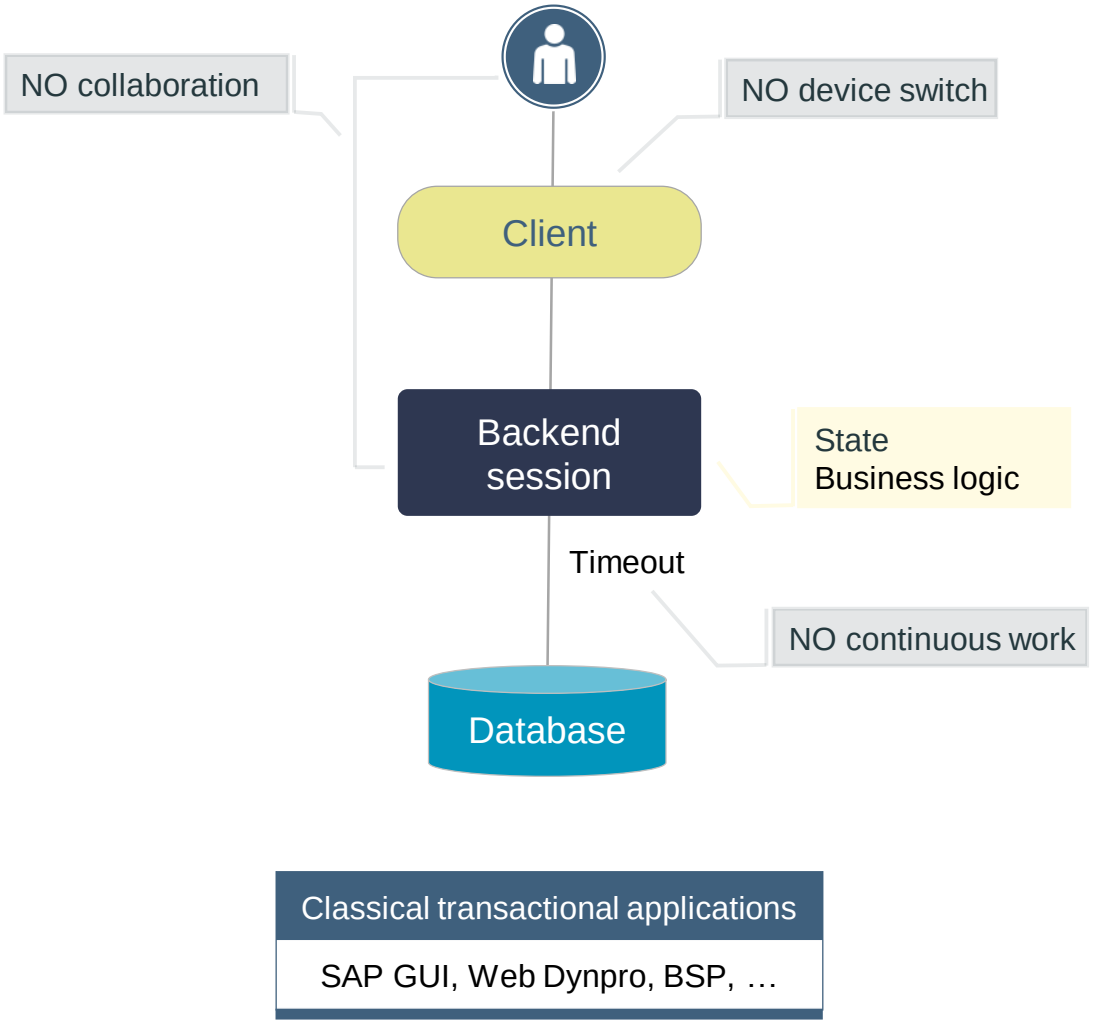
Impact on programming model Remove tight coupling of client and server to avoid server stickiness

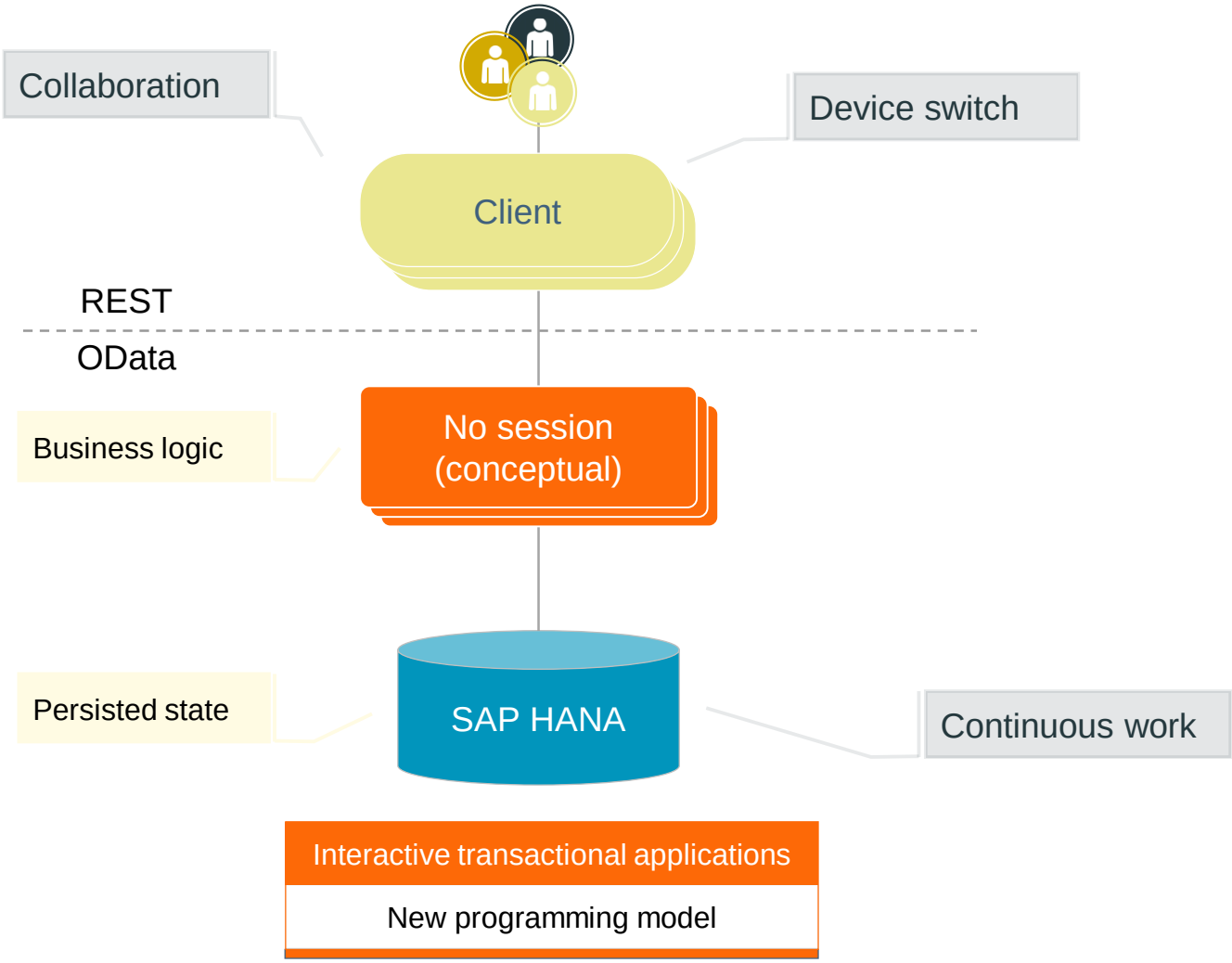
02

StateLess Apps

Impact of a RESTful architecture

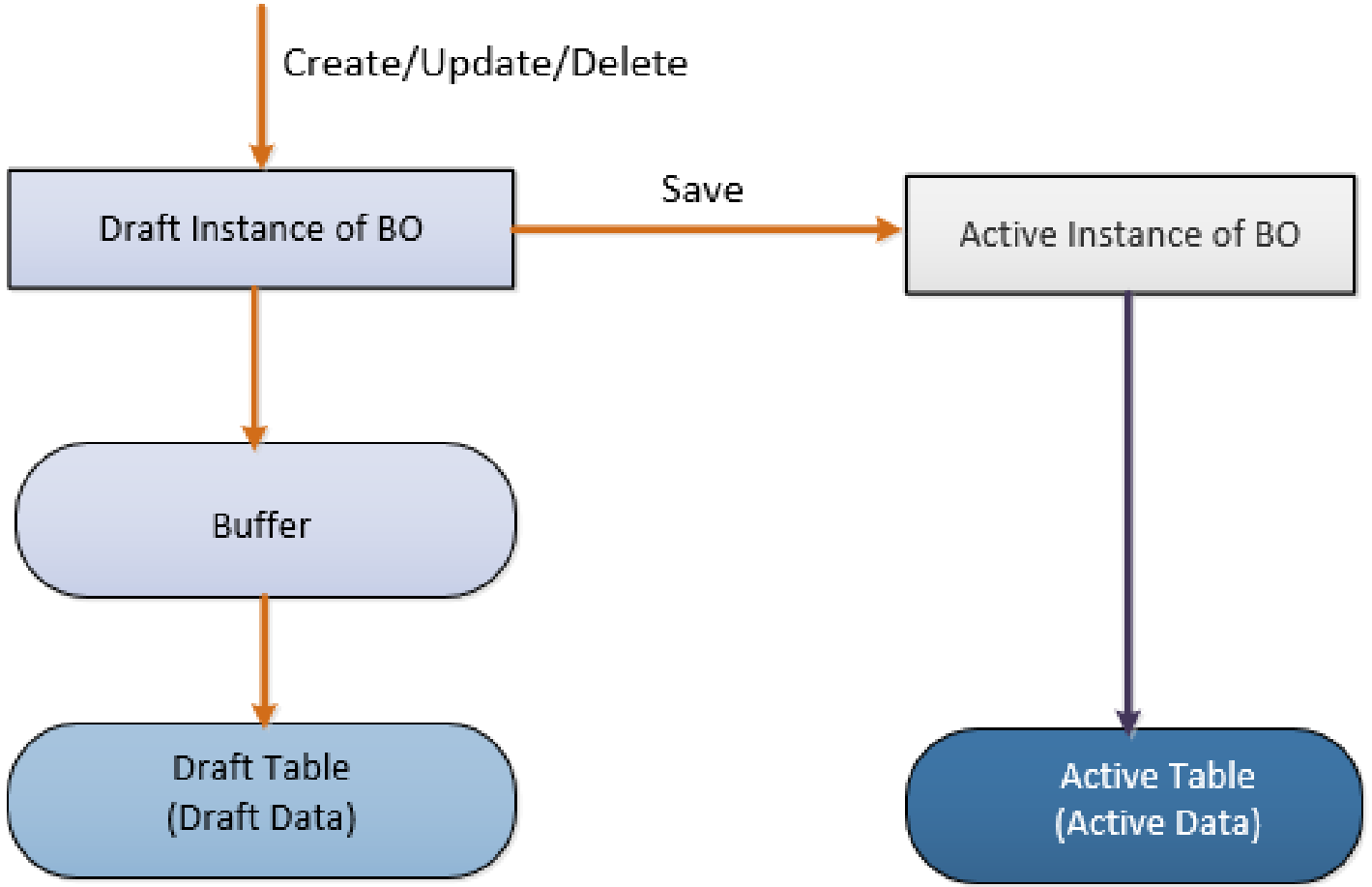
Impact on session, state and business logic





Draft support

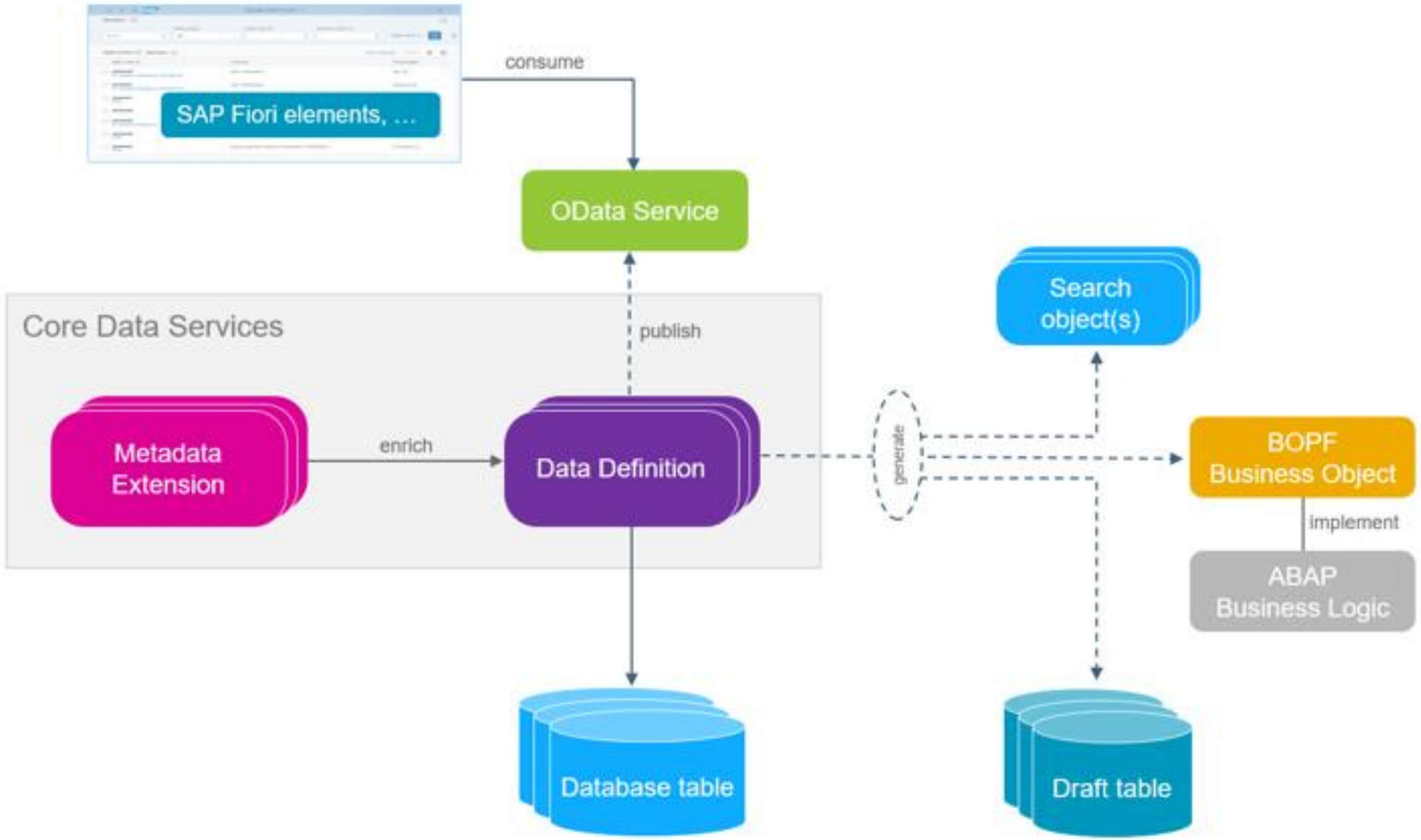
Active vs draft instances



03

Architecture

Involved technologies



Advanced data model definition and access
ABAP Core Data Services (CDS)

Capture business intent

Reduced complexity

Extending SQL

Common data model

Semantically rich

Declarative

Close to conceptual thinking

Improved programming model

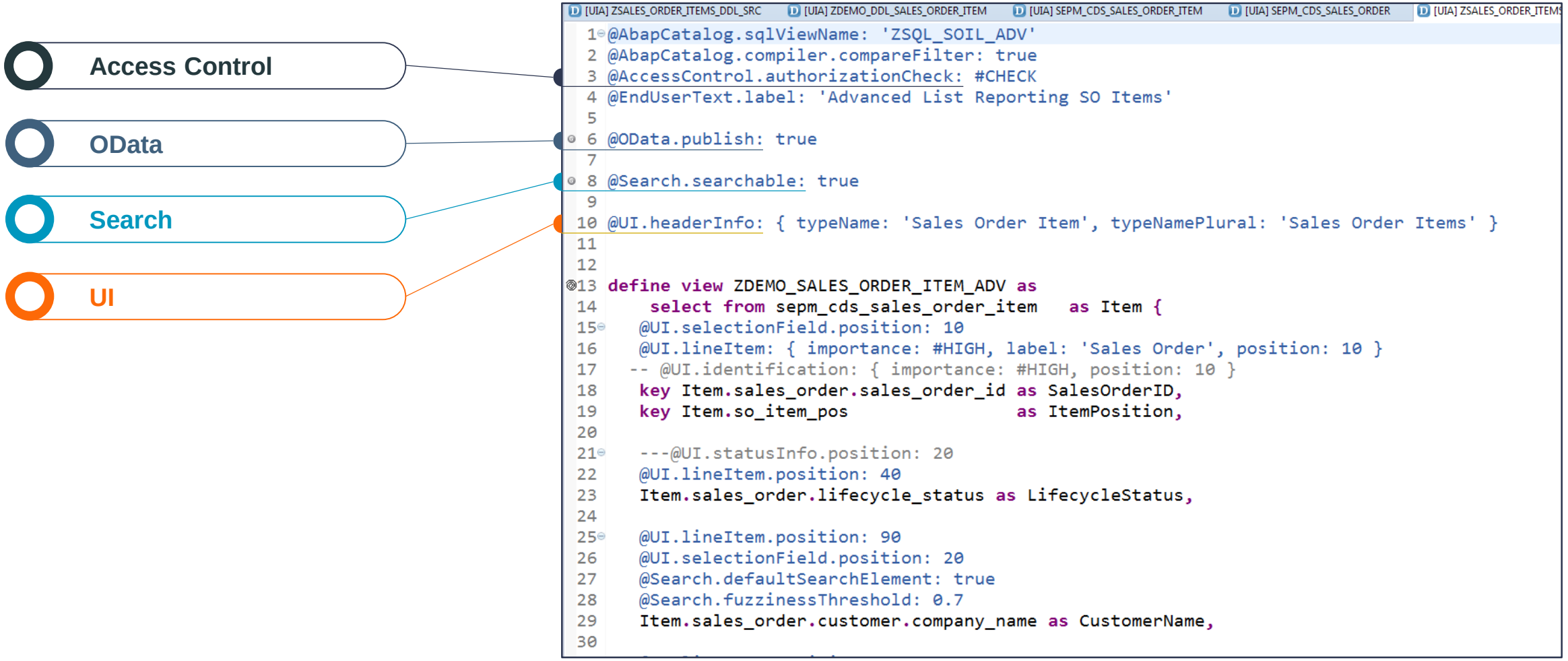
For all application domains

Rich set of build-in functions

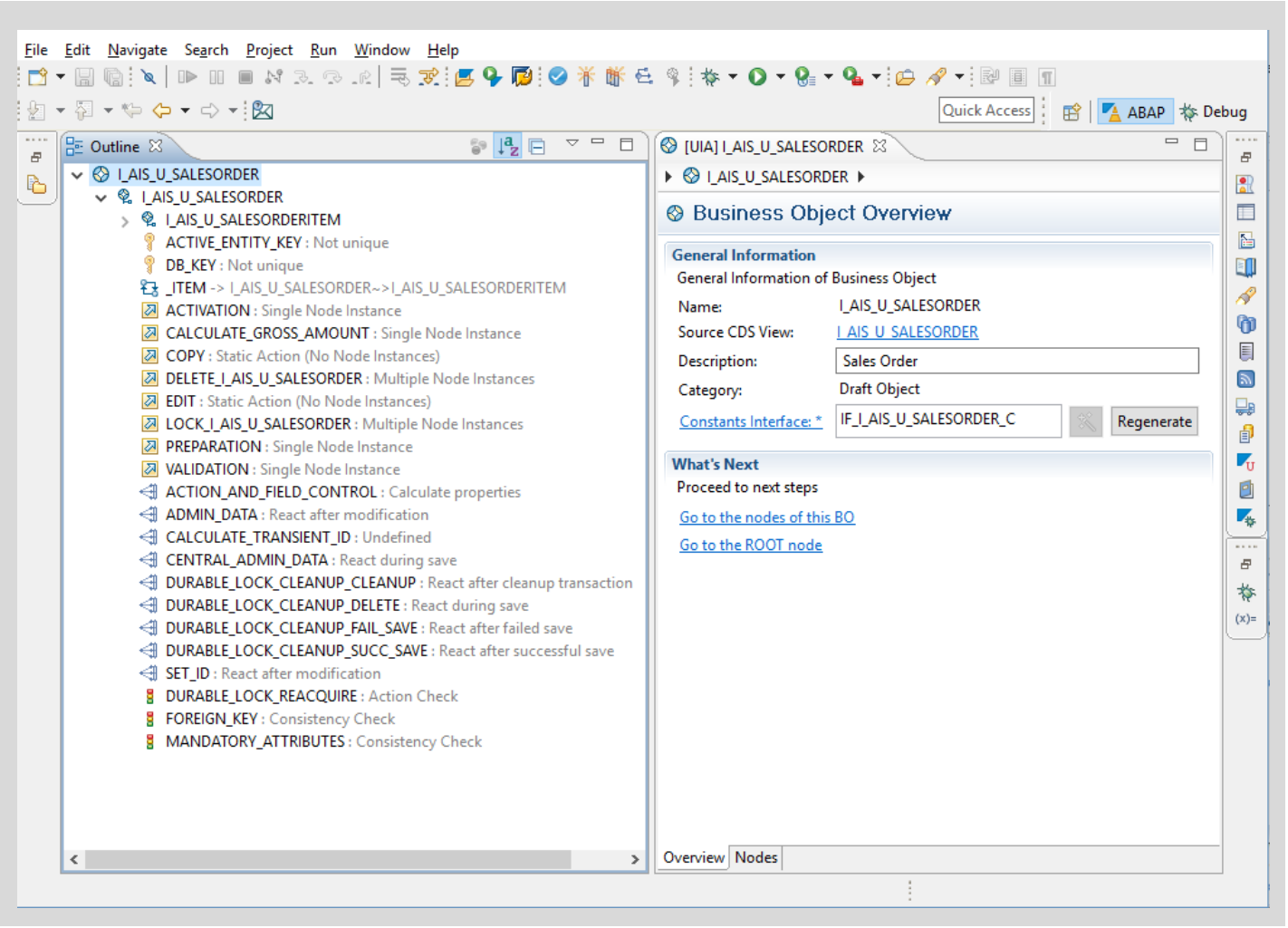
Code pushdown capabilities

Contributions to SAP S/4HANA

SAP S/4HANA applications – Example CDS view with annotations



Role of BOPF in programming model



Enables model-based and standardized development approach

Manages transactional data of draft and active instances within one business object

CRUDE

Locking

Authorization

Orchestrates business logic in well defined exits like

actions

determinations

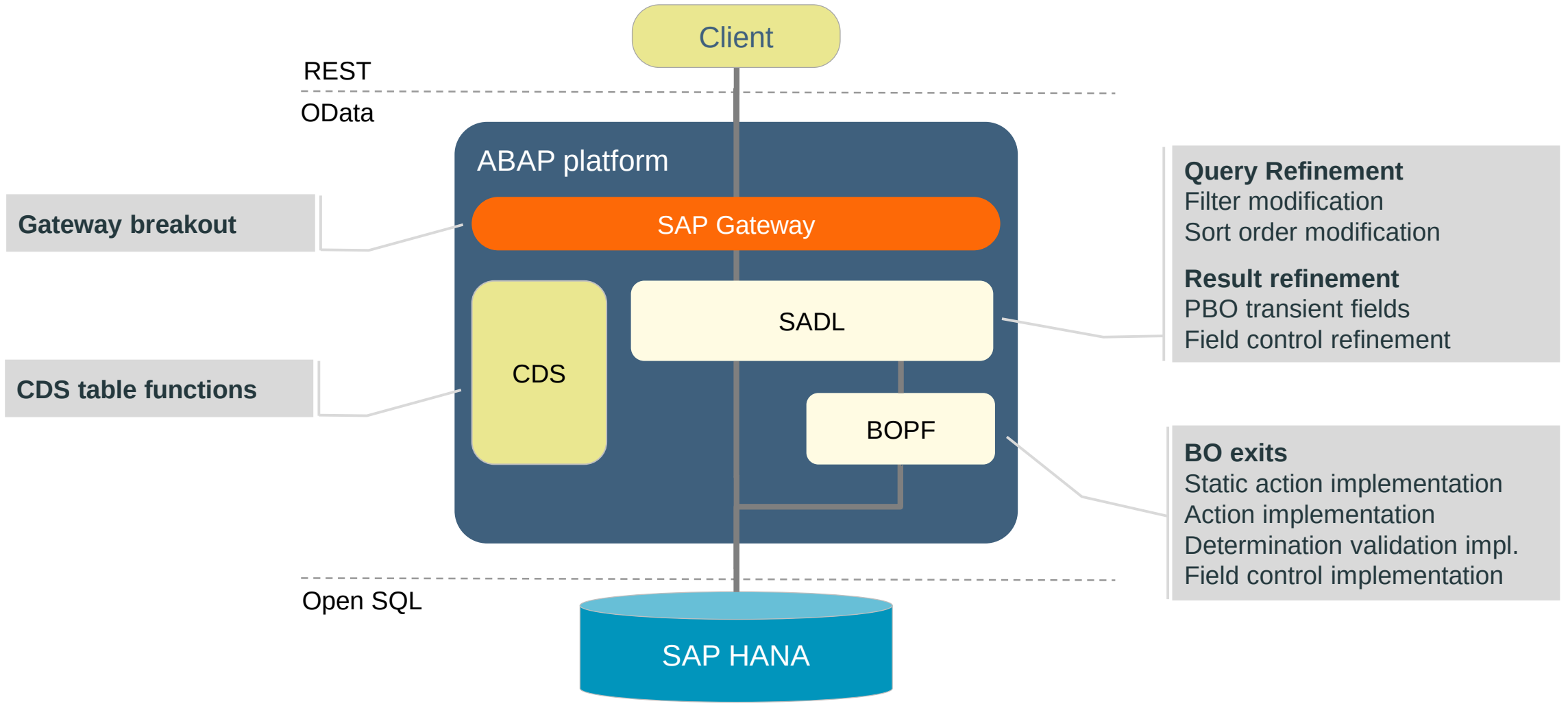
validations

draft handling

ABAP programming model for SAP Fiori and SAP S/4HANA

Runtime stack – Exits

PLANNED INNOVATION

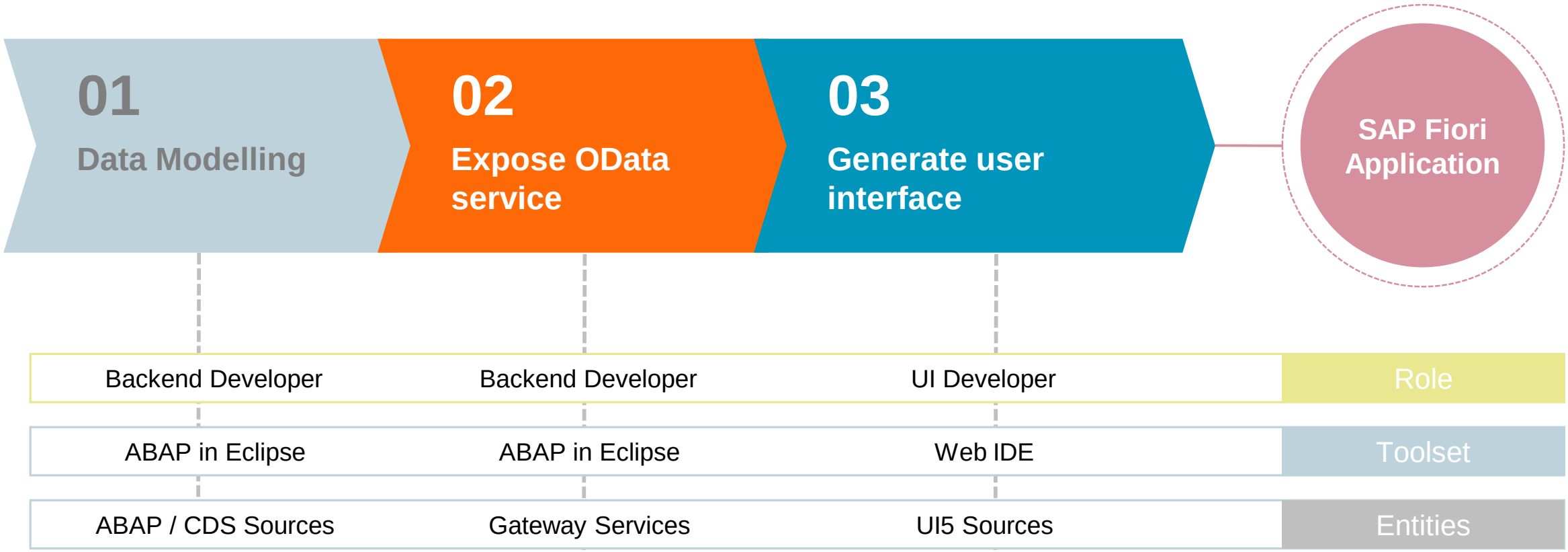


04

Build an App from Scratch

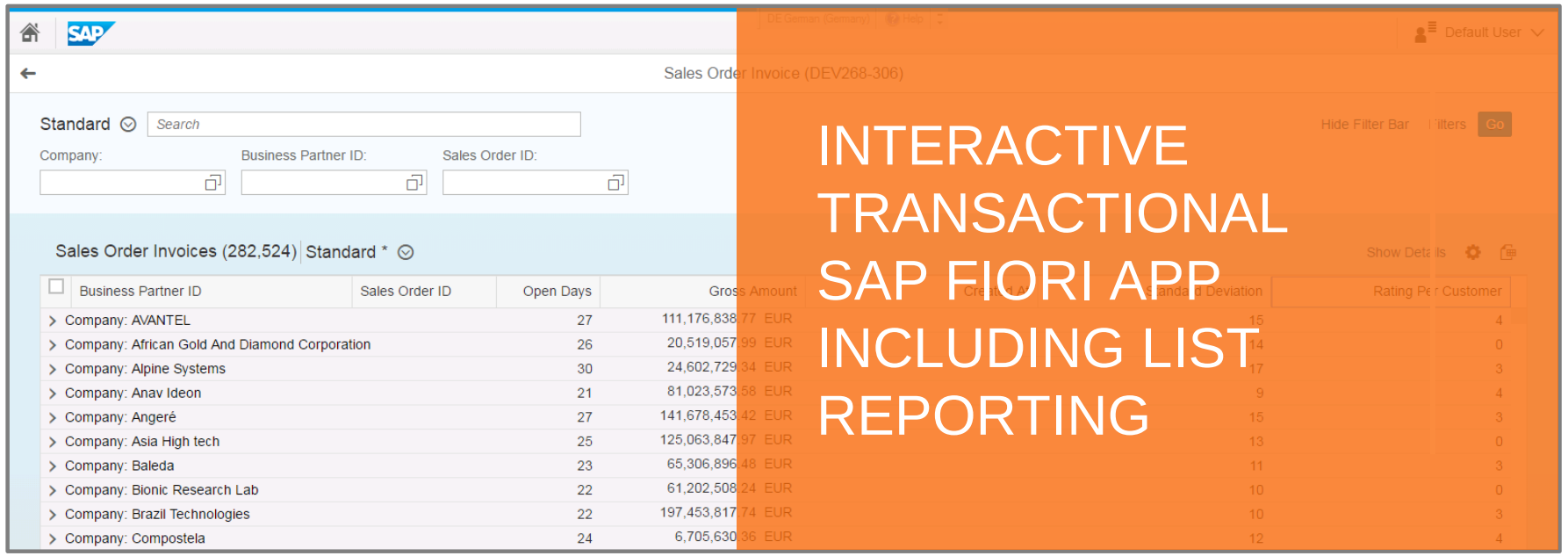
Programming model for SAP Fiori

Major steps and development tools



BUILDING AN APP FROM SCRATCH

Get the idea



We want to create a sales order overview with **analytics, search** and **transactional capabilities**.

05

Final

Summary
Key message

New programming model

New programming model available but still evolving

Documentation available

➨ Search for ‘ABAP Programming model Fiori’

Availability of app types

Read-only	➨	ABAP 7.50
’Batch input’	➨	Planned for ABAP 7.51
Interactive / Draft	➨	Planned for ABAP 7.51 SP01

Resources

SAP – Abap Programming Model for SAP Fiori

BOPF API Reference (Advance)