Clean Core Extensibility: What is in it? – How to measure? – How clean can one be?

Public

Disclaimer

The information in this presentation is confidential and proprietary to SAP and may not be disclosed without the permission of SAP. Except for your obligation to protect confidential information, this presentation is not subject to your license agreement or any other service or subscription agreement with SAP. SAP has no obligation to pursue any course of business outlined in this presentation or any related document, or to develop or release any functionality mentioned therein.

This presentation, or any related document and SAP's strategy and possible future developments, products and or platforms directions and functionality are all subject to change and may be changed by SAP at any time for any reason without notice. The information in this presentation is not a commitment, promise or legal obligation to deliver any material, code or functionality. This presentation is provided without a warranty of any kind, either express or implied, including but not limited to, the implied warranties of merchantability, fitness for a particular purpose, or non-infringement. This presentation is for informational purposes and may not be incorporated into a contract. SAP assumes no responsibility for errors or omissions in this presentation, except if such damages were caused by SAP’s intentional or gross negligence.

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, which speak only as of their dates, and they should not be relied upon in making purchasing decisions.

Agenda: Clean Core Extensibility

What is in it?

- Brief Re-Cap on Clean Core
- Ideas and News on Clean Core (Extensibility)

How clean can one be?

- Governance is key for clean core

How to measure?

- SAP approach: Today, tomorrow
- Project Experiences

What is in it?
Brief Re-Cap & Ideas and News

Addressing clean core and the individual dimensions

A **clean core** describes a system or a landscape of systems that is as close to standard as possible while running cloud-compliant extensions and integrations.

It allows you to **adapt your system and system landscape** to changing business requirements in order to **adopt new capabilities**.

Strong governance is required for each technical dimension of clean core.

Decouple extensions from standard

Main aspects

- **Avoid** extensions when possible
- Set up a **strong governance** to create decoupled extensions in a way that they would work in the cloud (three-tier model)
- Separate extensions by leveraging released APIs – **custom extensions** do not break an upgrade and **upgrades** do not break an extension
- Leverage the full capabilities of extensibility **on the stack** as well as side by side **with SAP BTP**
- Create **technical debts** only as informed decision

Keep **competitiveness** while **reducing complexity**.

1 Business processes

Decouple extensions from standard.

2 Extensibility

Control data according to **latest standards**.

3 Data

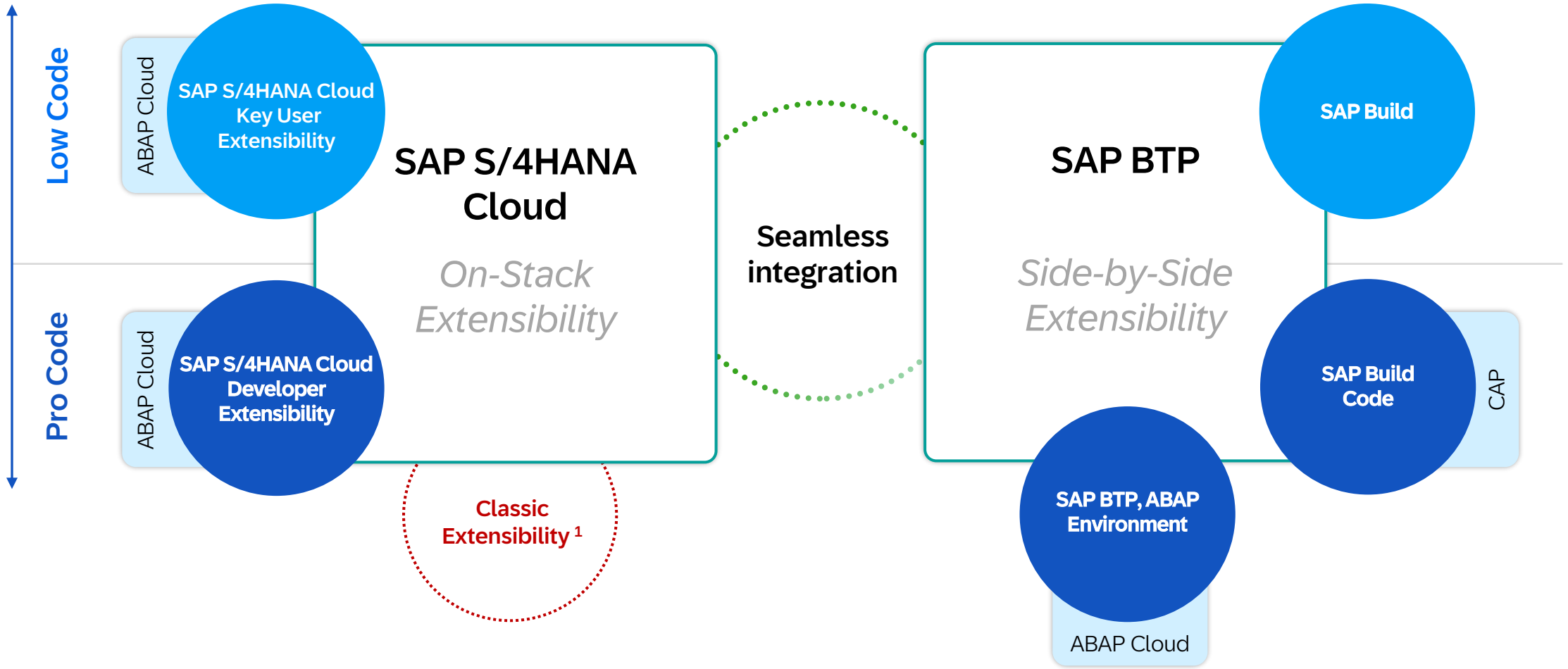
Keep the landscape **reliable and flexible**.

4 Integration

Keep the operations **effective and efficient**.

5 Operations

Extensibility Portfolio for SAP S/4HANA Cloud



[1] SAP S/4HANA Cloud, private edition and on premise only; **not** "clean core"

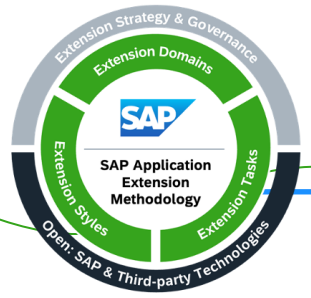
How clean can one be?

Governance Process (Extensibility Focus)

Functional Request / Gap Request

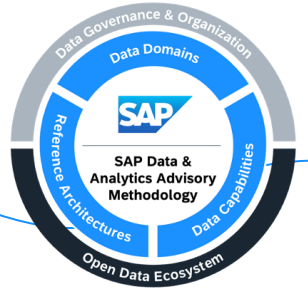
Solution Standardization Board

Extension Guidelines

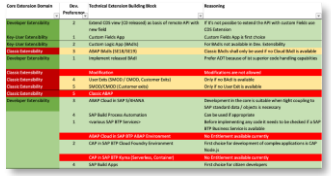


Integration Guidelines

Data Guidelines



Operations: Housekeeping, Quality, ...

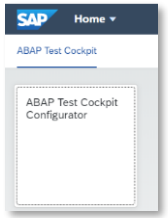


Extension Architecture

Exception Approval (Design)

Extension Implementation

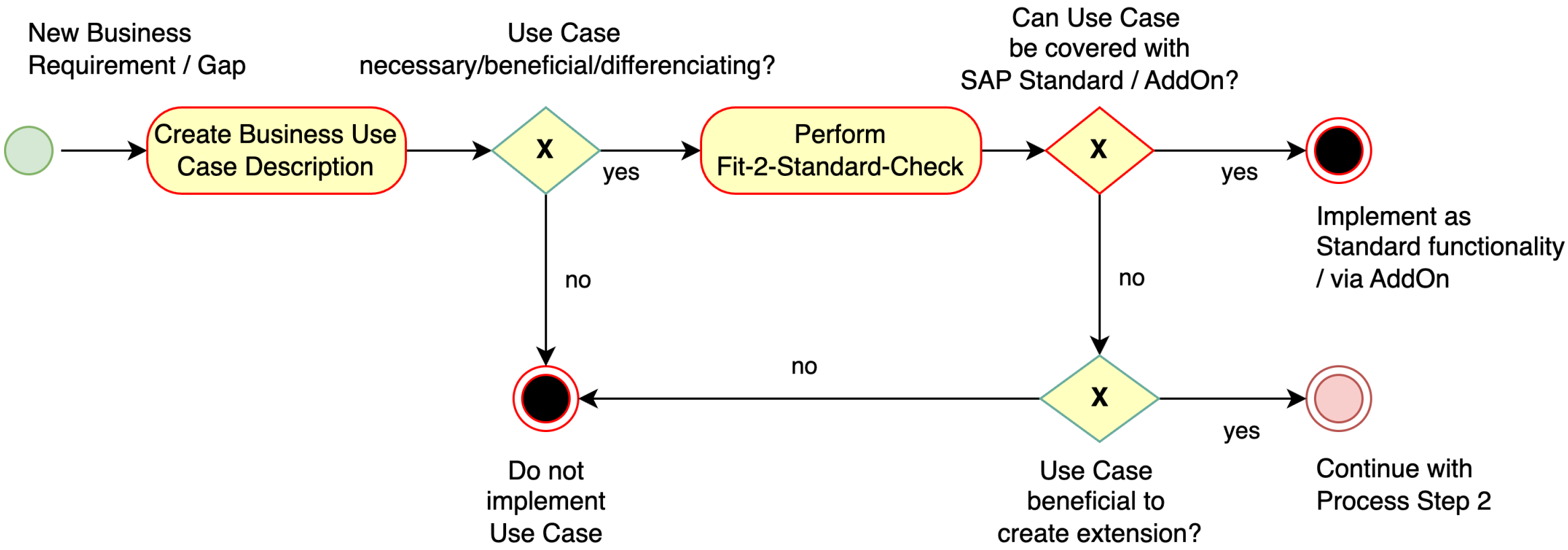
Exception Approval (Implementation)



Exception Approval (Transport)

Deployment

How to handle a Business Requirement?

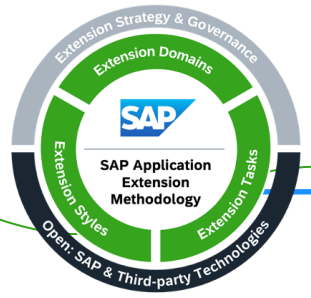


Governance Process (Extensibility Focus)

Functional Request / Gap Request

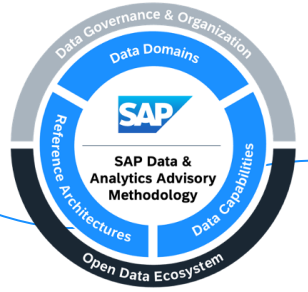
Solution Standardization Board

Extension Guidelines

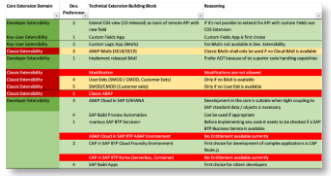


Integration Guidelines

Data Guidelines



Operations: Housekeeping, Quality, …



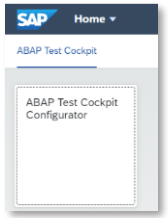
Extension Architecture

2

Exception Approval (Design)

Extension Implementation

Exception Approval (Implementation)

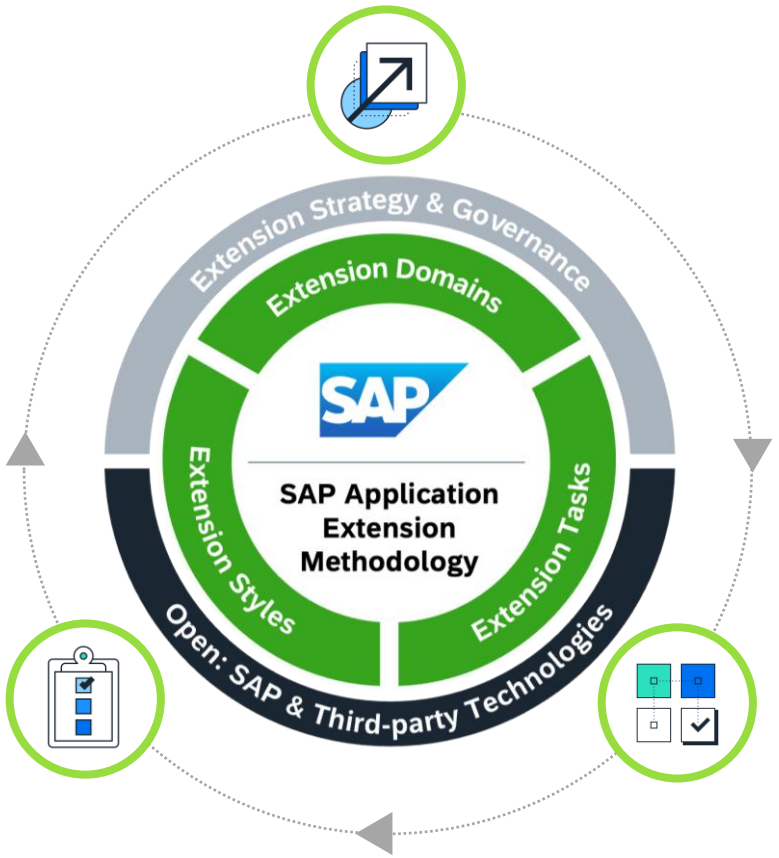


Exception Approval (Transport)

Deployment

SAP Application Extension Methodology – Goal & Outcome

Phase 1:
Assess Extension
Use Case



Phase 3:
Define Extension
Target Solution

Phase 2:
Assess Extension
Technology

Goal:

SAP Application Extension Methodology provides a structured, technology-agnostic approach for customers and partners to define and execute an enterprise extension strategy.

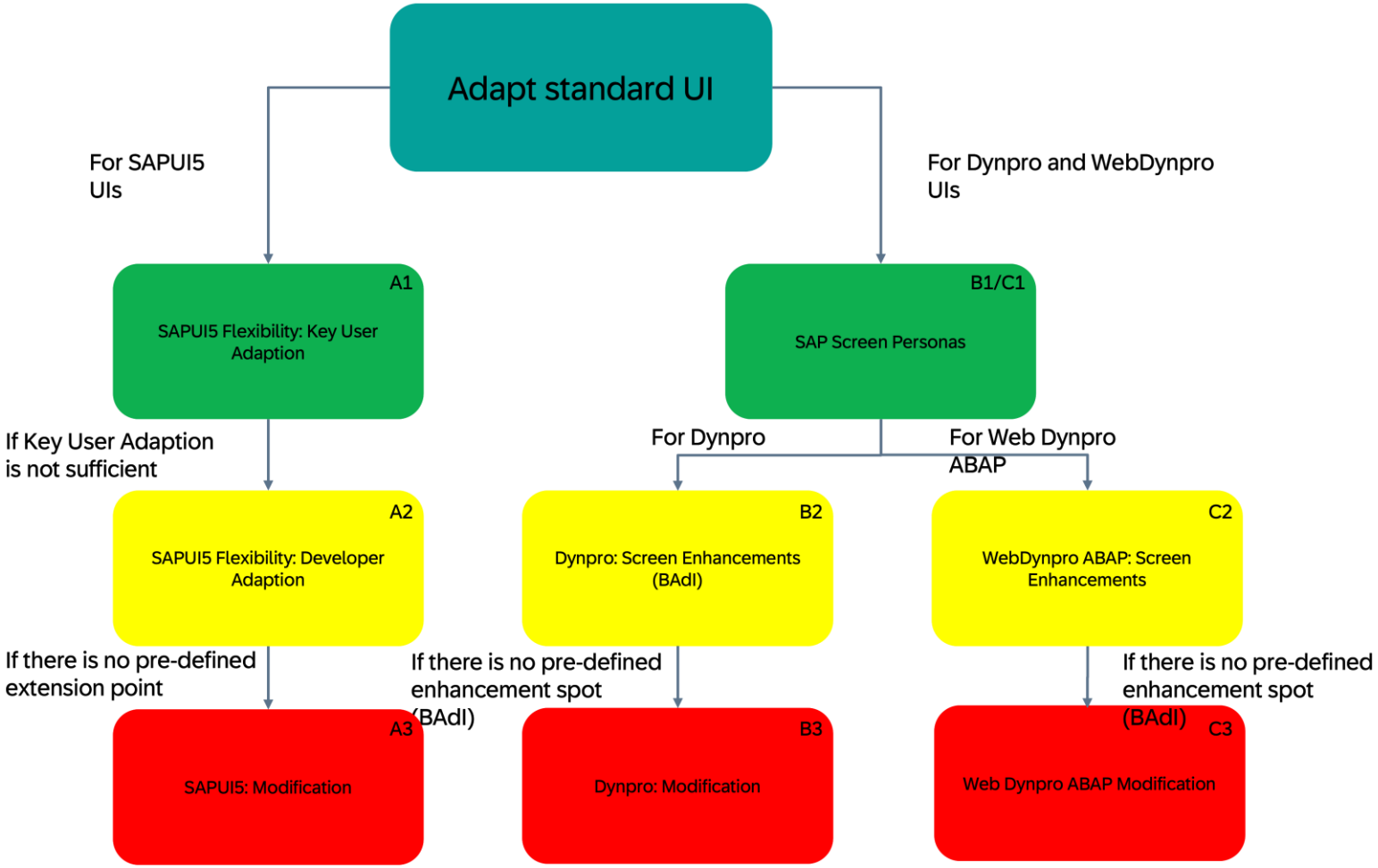
Outcome:

Define a clean core strategy to ensure upgrade stability

Enable enterprise architects to assess extension use cases and define an extension target solution.

Visualization of the Decision Guidance per Extension Task (customer specific)

Example



Enforce Governance on SAP S/4HANA

Exception Approval (Implementation / Transport)

	On-stack	Software Components	Authorization S_ABPLNGVS	ABAP Test Cockpit (ATC) Transport Release Checks
	SAP S/4HANA			
Clean Core	**TIER 1** • Key User Extensibility • Developer Extensibility (ABAP Cloud)	ZABAPCLOUD	ABAP for Key Users ABAP for Cloud Development	ABAP Cloud Readiness
Not Clean Core	**TIER 2** Cloud API enablement	HOME	Standard ABAP	ABAP Cloud Readiness
Not Clean Core	**TIER 3** Classical ABAP development	HOME	Standard ABAP	ATC 3-Tier Check (Note 3565942)

Housekeeping

Standardization

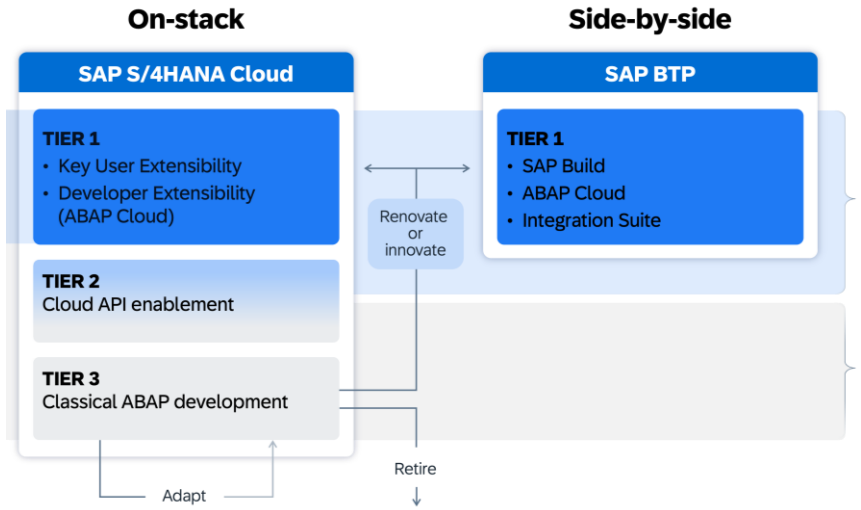
- Fit-2-Standard Workshop for Single Processes / Applications
- SAP Signavio Process Explorer – SAP Reference Architecture

Renovate & Innovate & Adapt

- SAP Application Extension Methodology
- Whitepaper: ABAP Cloud - Technical Use Cases and Recommended Technologies
- ABAP Test Cockpit
  Variant: ABAP_CLOUD_DEVELOPMENT_3TIER

Retire

- Usage Data Evaluation with Custom Code Migration App



Governance and Operational Practices Extensibility Maturity - Result

Extensibility Score	0	<comment>

ILLUSTRATIVE

Key Action(s)	<actions>

	Practice	Importance	Score
Governance Setup (0,00)	Does your governance process (SSB) ensure that extensions are only created if needed / valuable?	★★★★★	0
	Do your architecture guidelines consider the following aspects around the SAP S/4HANA Extensibility Model?	★★★★★	0
	Are your development guidelines following the clean core principles?	★★★★★	0
	Do your developers have the right skills to cover the new extensibility options?	★★★★★	0
	Is there a strategy to identify extensions that are relevant for a "clean-up" to clean up technical debts?	★★★★☆	0
	Do you ensure that only used code is in scope of the conversion and code adaptation for SAP S/4HANA Readiness?	★★★★☆	0
	Are you measuring KPIs around Clean Core and defining goals around them?	★★★★★	0

	Practice	Importance	Score
System Setup (0,00)	Have you setup automated code checks in your system?	★★★★★	0
	Are Exemptions Setup properly in your System? (ATC)	★★★★★	0
	Have you already setup in your development system a separate software component for Tier 1 development?	★★★☆☆	0
	Is usage data collection active in your productive system (with SCMON / SUSG)?	★★★☆☆	0
	Is your SAP BTP Account properly configured to support the Clean Core Extensibility approach?	★★☆☆☆	0

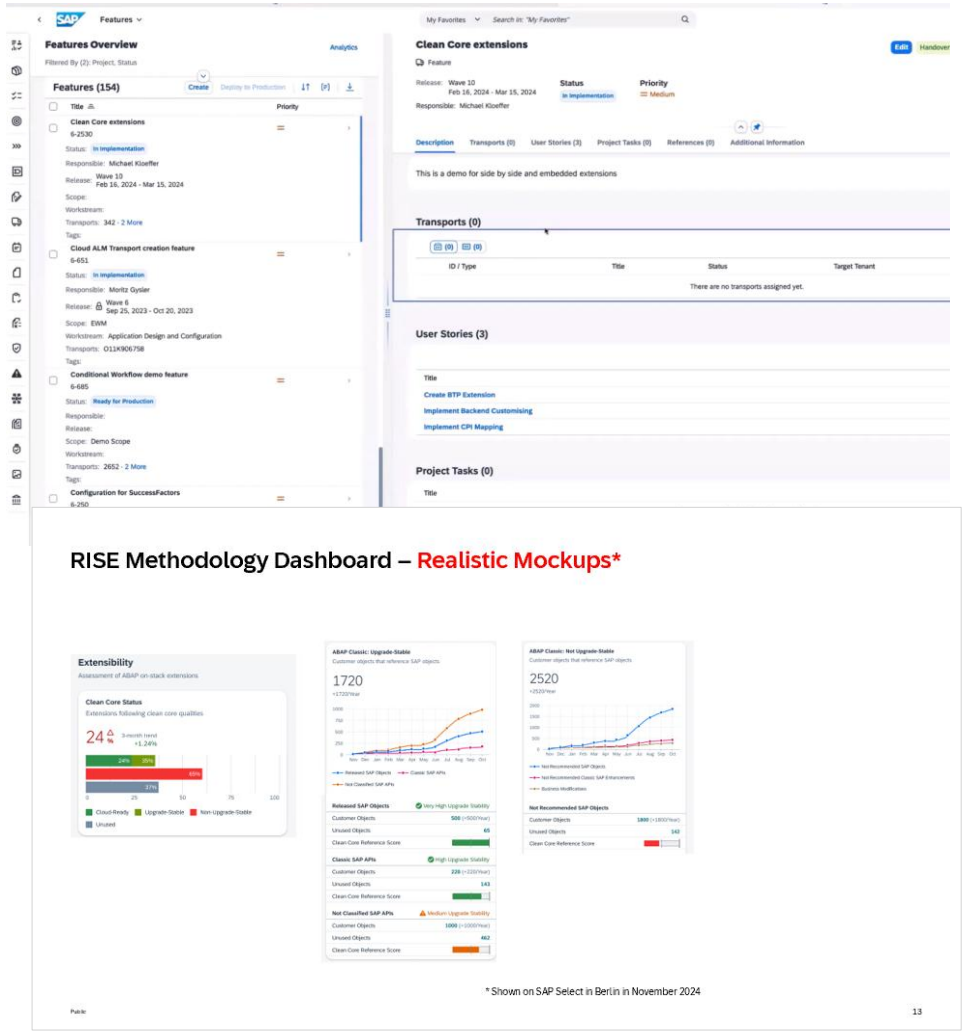
How to measure?

Measuring Clean Core – the SAP way

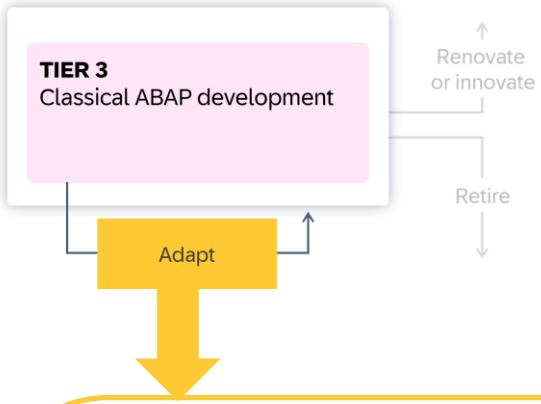
Manage your gaps and features in SAP Cloud Application Lifecycle Management (CALM) – e.g. with “clean” tags

Learn about improvements (now and upcoming), especially on:

- RISE with SAP Methodology Dashboards in CALM
- ABAP Test Cockpit (ATC)



Enhanced guidance for custom development in tier 3



- Recommended classic APIs for tier 3 (e.g. CL_GUI_ALV_GRID) provided as JSON file on GitHub
- New ATC check variant detecting the classic ABAP custom code which shall be adapted
  - Report violation of standard security and quality rules
  - Report on modifications and implicit and explicit enhancements
  - Report use of critical statements, for example, SYSTEM-CALLs
  - Enforce usage of SAP released and classic APIs instead of internal APIs
  - Enforce Clean ABAP guidelines

Classic APIs on GitHub

CL_SALV_TABLE Classic API

Class (CLAS)

Application Component: BC-SRV-ALV
Software Component: SAP_BASIS
Main Object Name: CL_SALV_TABLE
Main Object Type: CLAS
Object Type: CLAS

BAPI_PR_CREATE Classic API

Function Module (FUNC)

Application Component: MM-PUR-REQ
Software Component: S4CORE
Main Object Name: 2105
Main Object Type: FUGR
Object Type: FUNC

Labels

Label
remote-enabled
transactional-consistent

New ATC Check based on usage guidelines for SAP APIs

New ATC check

Usage of APIs <remote-enabled>

- Checks usages of SAP standard objects in custom code like interfaces, classes, function modules, CDS views, behavior definitions, DDIC database tables and DDIC database views, programs or their subroutines.

- Does not check usages SAP DDIC object types like data elements, domains, table types and structures

- Check behavior
  - ➢ **Priority 3 (Info)**
    - ✓ Usage of classic APIs
  - ➢ **Priority 2 (Warning)**
    - ✓ Usage of internal API (not classified API)
    - ✓ SQL read access to SAP database table
    - ✓ SUBMIT statement on programs
  - ➢ **Priority 1 (Error)**
    - ✓ Usage of “no API” (with successor if available)
    - ✓ SQL write access to SAP database table
    - ✓ PERFORM statements on external subroutines

Example: Check a simple ABAP report with ALV using

ABAP_CLOUD_READINESS checks:

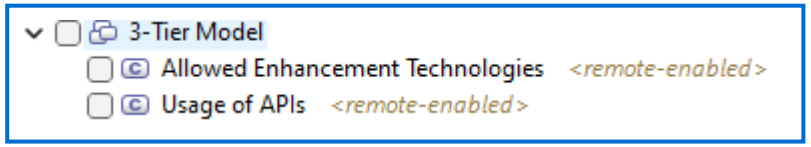
Findings: 14 Errors	
ABAP Language Version (Syntax) (5 Errors)	
Syntax error in restricted language scope (Open SQL)	SELECT
Syntax error in restricted language scope (Open SQL)	UPDATE
Syntax error in restricted language scope (dynpro)	PARAMETERS
Syntax error in restricted language scope (report)	REPORT
Syntax error in restricted language scope (report)	START-OF-SELECTION
Allowed Object Types in Cloud Development (1 Errors)	
Objects of type PROG are not allowed in ABAP Cloud Development	
Usage of Released APIs (8 Errors)	
Usage of API that will not be released.	BUT000
Usage of API that will not be released.	BUT000
Usage of API that will not be released.	BUT000
Usage of API that will not be released.	BUT000
Usage of not released ABAP Platform APIs.	BU_PARTNER
Usage of not released ABAP Platform APIs.	CL_SALV_TABLE
Usage of not released ABAP Platform APIs.	CL_SALV_TABLE
Usage of not released ABAP Platform APIs.	CX_SALV_MSG

New ATC check Usage of APIs

Findings: 1 Errors, 2 Warnings, 2 Infos	
Errors (1 Errors)	
Updating DDIC database tables or DDIC table views is not allowed (successor available)	BUT000
Warnings (2 Warnings)	
Reading from DDIC database tables or DDIC table views is not recommended (successor available)	BUT000
Usage of internal API	CX_SALV_MSG
Infos (2 Infos)	
Usage of classic API	CL_SALV_TABLE
Usage of classic API	CL_SALV_TABLE

New ATC Checks – Availability Timeline

The new ATC Checks



- Already available with SAP Note 3565942 for **SAP S/4HANA Cloud Private Edition 2023** and **SAP S/4HANA on-premise 2023**
- Will become available for ATC on SAP BTP **15th March with HFC for SAP BTP ABAP Environment 2502**

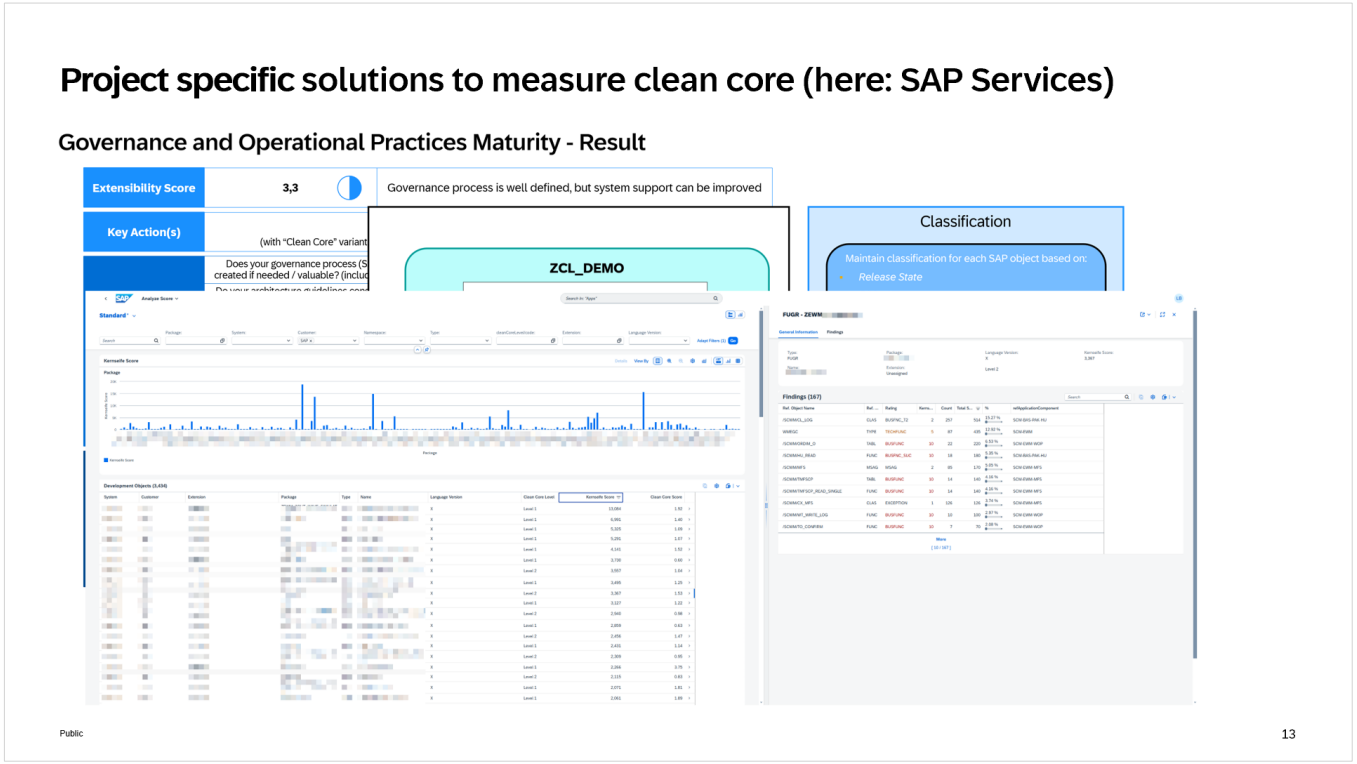
Measuring Clean Core – the {projectname} way

Governance, Governance, Governance!

Identify reasonable KPIs (insightful, actionable) for your project – if needed: Build Dashboards on your own based on your project need (e.g. greenfield vs. brownfield has different needs)

Manage your requirements in “your” tool (project example: Azure DevOps) with a connection to “clean core” and approvals

Leverage ATC Checks and measure ATC results regularly



Thank you.

Contact information:

Lukas Bretschneider
lukas.bretschneider@sap.com