



SAP S/4HANA Cloud, Public edition ABAP拡張ブートキャンプ°

開発者拡張における中核技術 SAP ABAP RESTful Application Programming Model (RAP)

田中貴之/ 後藤健司
SAP BTP App. Dev & Integration, Customer Advisory
SAP Japan 株式会社

INTERNAL – SAP and Partners Only

*本資料に関する使用条件、免責条項、開示、または制限事項については、法律上の注意点 (www.sap.com/legal-notice) を参照してください。

本ワークショップのスケジュール

11月26日

開催場所: SAPジャパン本社 11F 山吹 (オンサイト)

10:00 – 17:00

10:00 – 11:45	プレゼンテーション “開発者拡張における中核技術 SAP ABAP RESTful Application Programing Model(RAP)”
11:45 – 13:00	昼食
13:00 – 14:00	演習 1 S/4HANA Cloud, public edition 拡張のシンプルなABAP クラスへの実装
14:00 – 14:15	休憩
14:15 – 16:45	演習2 S/4HANA Cloud, public edition 拡張 + RAPを用いた簡易 オンラインショップアプリの作成
16:45 – 17:00	クロージング

本日の内容

- ❑ “クラウドとオンプレにおけるSAP S/4HANAのためのクラウド拡張オプションの”超”概要” (2023年6月8日開催のパートナー様向けWebiner セッション)の振り返り
- ❑ 開発者拡張の中核技術: SAP RESTful Application Programing Model (RAP)概要
- ❑ RAP ファサード / RAP BO Interface を用いるための EML (Entity Manipulation Language)
- ❑ RAP における Behavior Definition および Behavior Implementationの実装
- ❑ ご参考情報

Agenda

- ❑ “クラウドとオンプレにおけるSAP S/4HANAのためのクラウド拡張オプションの“超”概要” (2023年6月8日開催のパートナー様向けWebiner セッション)の振り返り
- ❑ 開発者拡張の中核技術: SAP RESTful Application Programing Model (RAP)概要
- ❑ RAP ファサード / RAP BO Interface を用いるための EML (Entity Manipulation Language)
- ❑ RAP における Behavior Definition および Behavior Implementationの実装
- ❑ ご参考情報

新しいクラウド拡張オプションの基本的な考え方

SAP Cloud Solution における前提

パブリッククラウドでは、SAP ソフトウェアの更新が自動化され、すべてのテナントにおいて一斉に実行されます。

SAPがご提供する手順に基づいたアップグレードプロセスに準拠。

クラウド拡張オプション共通ルール

その 1

拡張では、リリース済 SAP API のみを使用可能。SAP では、これらの API の安定性を維持します。

その 2

SAP オブジェクトは、事前定義された拡張ポイントを介してのみ拡張可能。フリースタイル変更は許可されません。

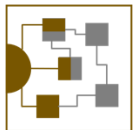
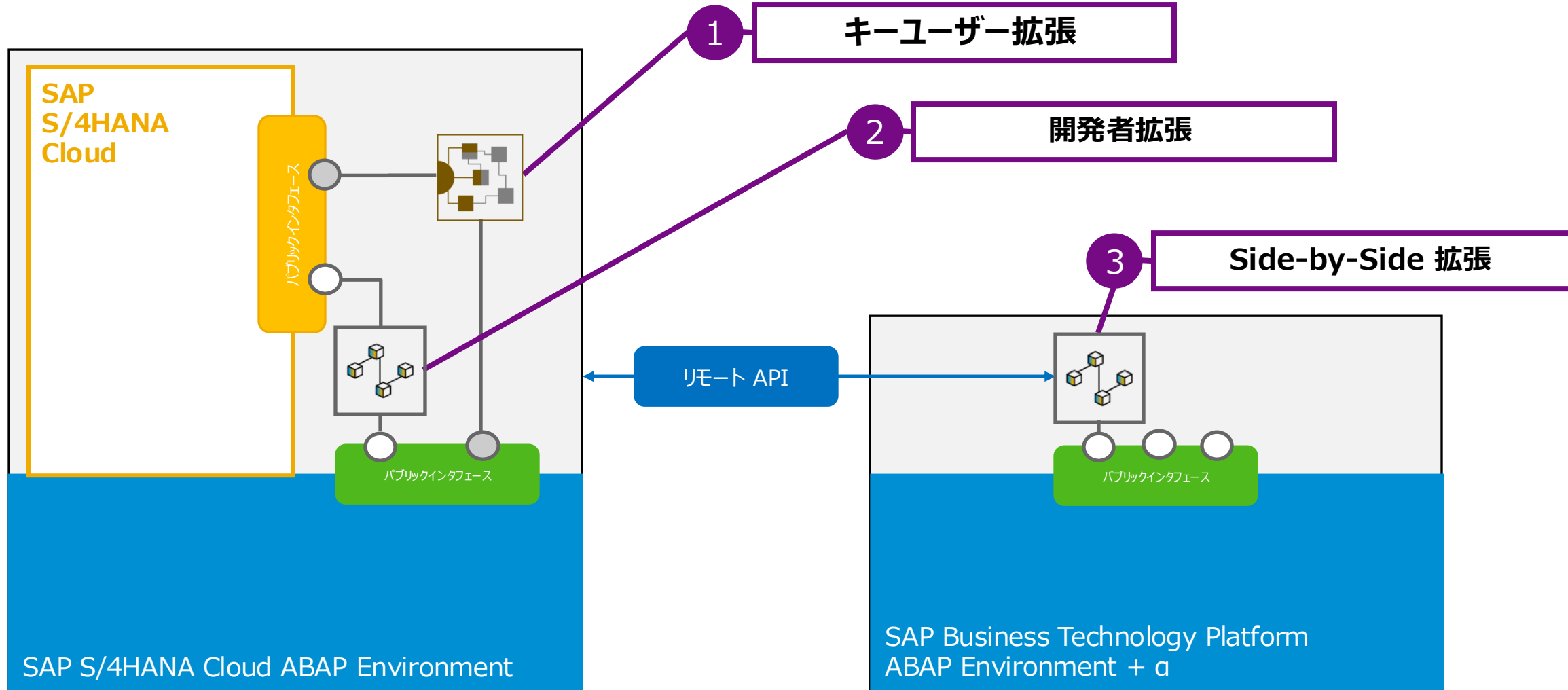
その 3

キーユーザおよび拡張開発者は、クラウド対応およびリリース済のテクノロジーのみを使用。開発者およびキーユーザのアクティビティは、アクセス権限によって規制されます。

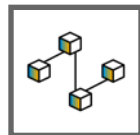
S/4HANA の各エディション毎の新しい拡張オプションの適応アプローチ (SAP推奨)

	新しいクラウド拡張オプションへの対応方法
SAP S/4HANA Cloud public Edition	必須 <input type="checkbox"/> 旧来型のクラシックなABAP拡張はサポートなし
SAP S/4HANA Cloud private Editionおよび オンプレミス グリーンフィールド導入	優れたクラウド拡張オプションをベースとして導入開始 <input type="checkbox"/> スムーズなSAPソフトウェアのアップグレード <input type="checkbox"/> 来るべきクラウド移行のための将来的に安全な拡張
SAP S/4HANA Cloud private Editionおよび オンプレミス クラシック拡張を変更する導入	新しいクラウド拡張オプションを取り入れつつ、適応が難しい部分のみ従来の拡張方法を利用した導入 <input type="checkbox"/> 開発者は新しいクラウド拡張オプション習得し、少しずつクラシック拡張をクラウドに対応した拡張に移行する

新しいSAP S/4HANA Cloud 拡張オプションの全体像



キーユーザー拡張
Low コード/No コード拡張



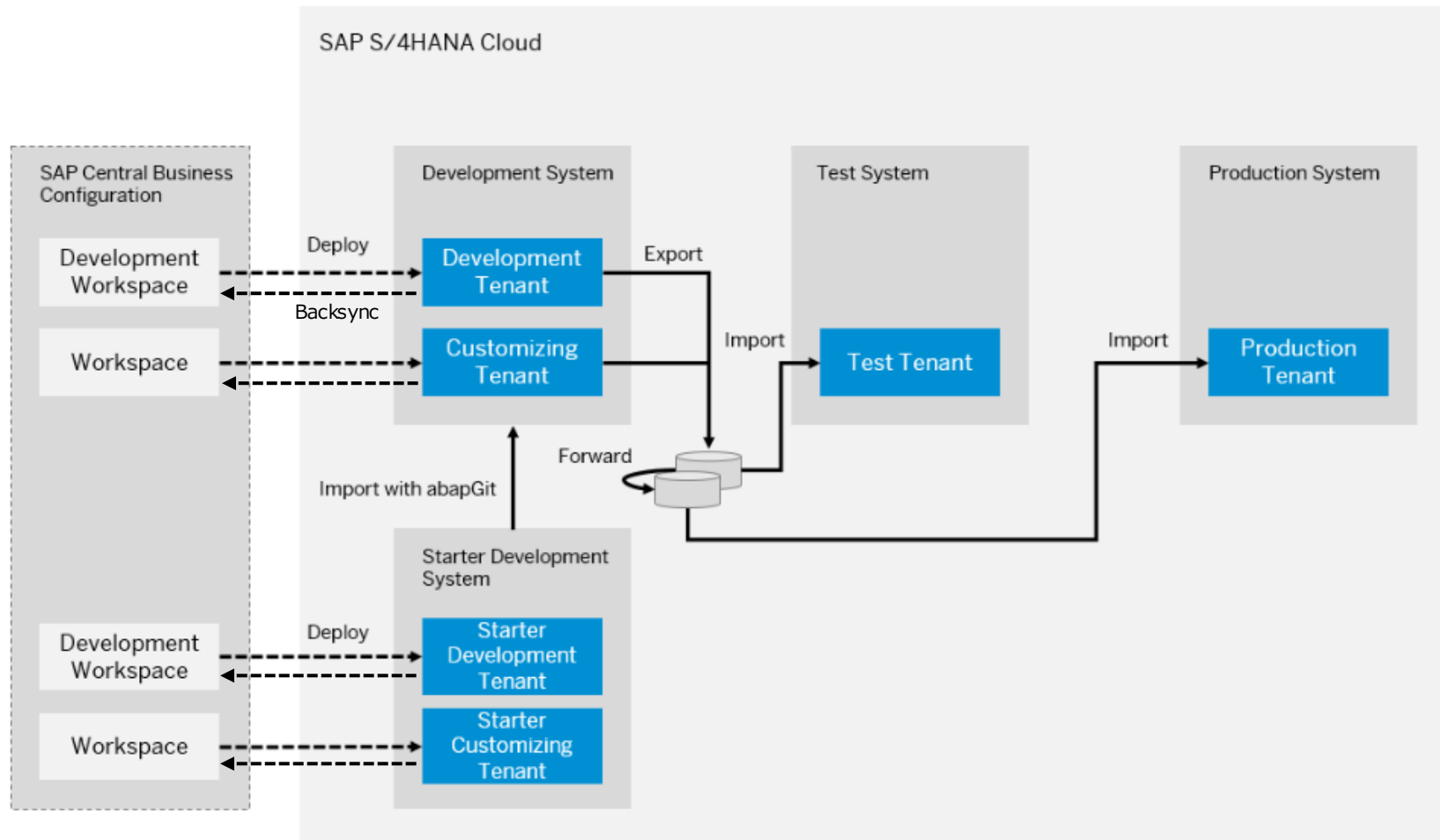
ABAPカスタムコード or パートナー拡張
クラウド拡張モデルに準拠

- 開発者拡張のためにリリースされたインターフェース
- キーユーザー拡張のためにリリースされたインターフェース

SAP S/4HANA Public Cloud における 拡張性オプションの概要

	キーユーザ拡張  ビジネスエキスパート, 導入 コンサルタント, シチズンデベロッパー, キーユーザ	開発者拡張  ABAP 開発者	Side-by-side拡張  ABAP 開発者+ α
シナリオ	SAP S/4HANA アプリケーションの ローコード/ノーコード適応と拡張	SAP S/4HANA データ、トランザクション、 またはアプリとの近接性または結合を必要と するカスタム ABAP 開発プロジェクト	疎結合されたアプリケーションとパートナーの SaaS ソリューション
ユース ケース	<ul style="list-style-type: none"> UI の調整、ユーザ定義項目の追加、 カスタムビジネスオブジェクト の追加など 	<ul style="list-style-type: none"> SAP S/4HANA データへの頻繁または複雑な SQL アクセスを伴うカスタムアプリ ケーション SAP アプリケーションと同じ作業論理単 位 (LUW) で実行されるカスタム拡張 Side-by-Side開発での SAP BTP アプリを 提供する、カスタマイズされたカスタムリ モート API またはサービス 	<ul style="list-style-type: none"> 個別のターゲットグループ向けのカスタムア プリケーション (S/4ユーザ以外が利用) ERP から分離して実行されるカスタムアプリ ケーションワークロード 機械学習や AI などのインテリジェントな SAP BTP サービスに近接する必要があるカス タムアプリケーション 複数の ERP システムおよびクラウドサービス と統合されるソリューション パートナーが提供する SaaS アプリケーション
利点	SAP S/4HANA Cloud で完全に管理 および統合 開発スキルがないか、非常に基本的な スキルのみ必要	SAP S/4HANA Cloud システム内での 拡張の開発 リモートアクセスまたはデータレプリケーシ ョンなし リリース済の SAP S/4HANA Cloud オブジ ェクトの使用および拡張	SAP S/4HANA Cloud の運用およびライフサイ クル管理に依存しない分離された拡張
	On-Stack拡張		Side-by-side拡張

S/4HANA Cloud 3SLの場合の移送プロセスと機能システムおよびテナント



スターターシステム

- (On-Stack開発のための)開発テナント
- (設定およびキーユーザ拡張のための)カスタマイジングテナント

開発システム

- (On-Stack開発のための) 開発テナント
- (設定およびキーユーザ拡張のための)カスタマイジングテナント

テストシステム

- テストテナント

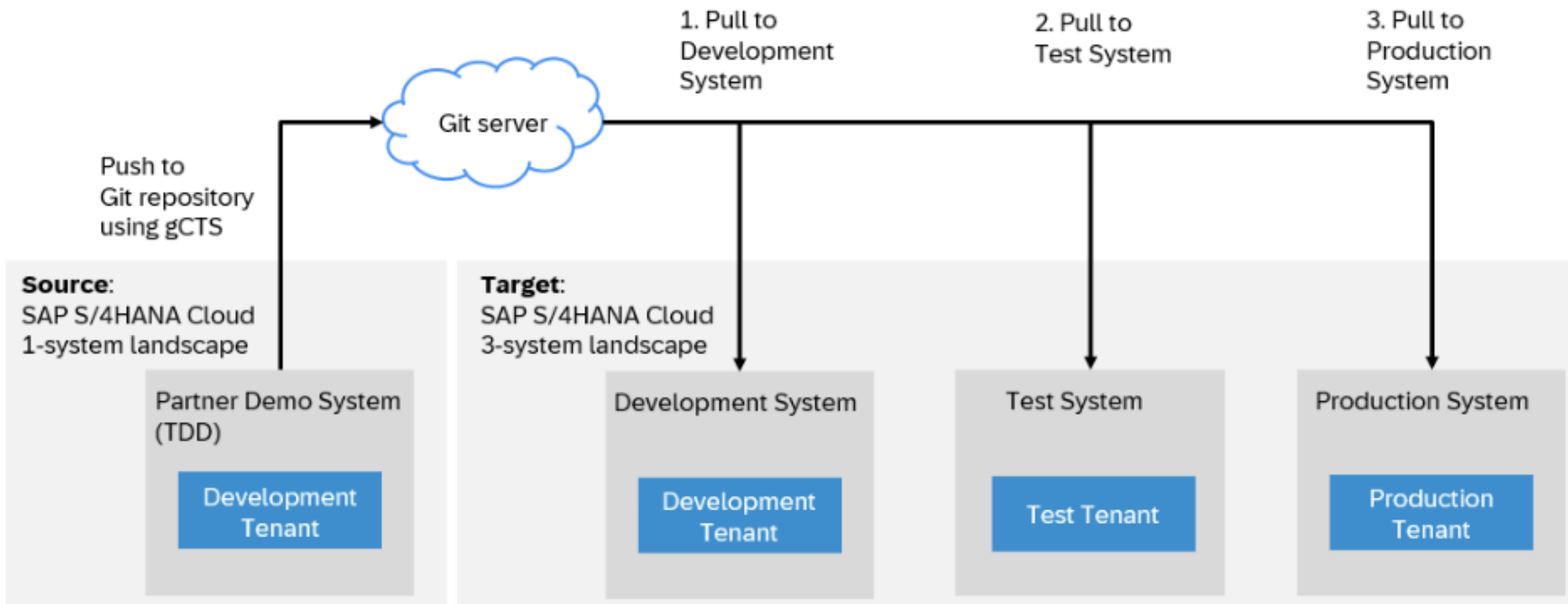
本稼動システム

- 本稼動テナント

マルチオフデリバリー機能

Git enabled Change and Transportant System (gCTS)を用いた
ランドスケープ登録されていないS/4HANA Cloud, public editionへの移送方法




利用ケース: S/4HANA Cloud のパートナーソリューションの顧客環境への移送プロセス



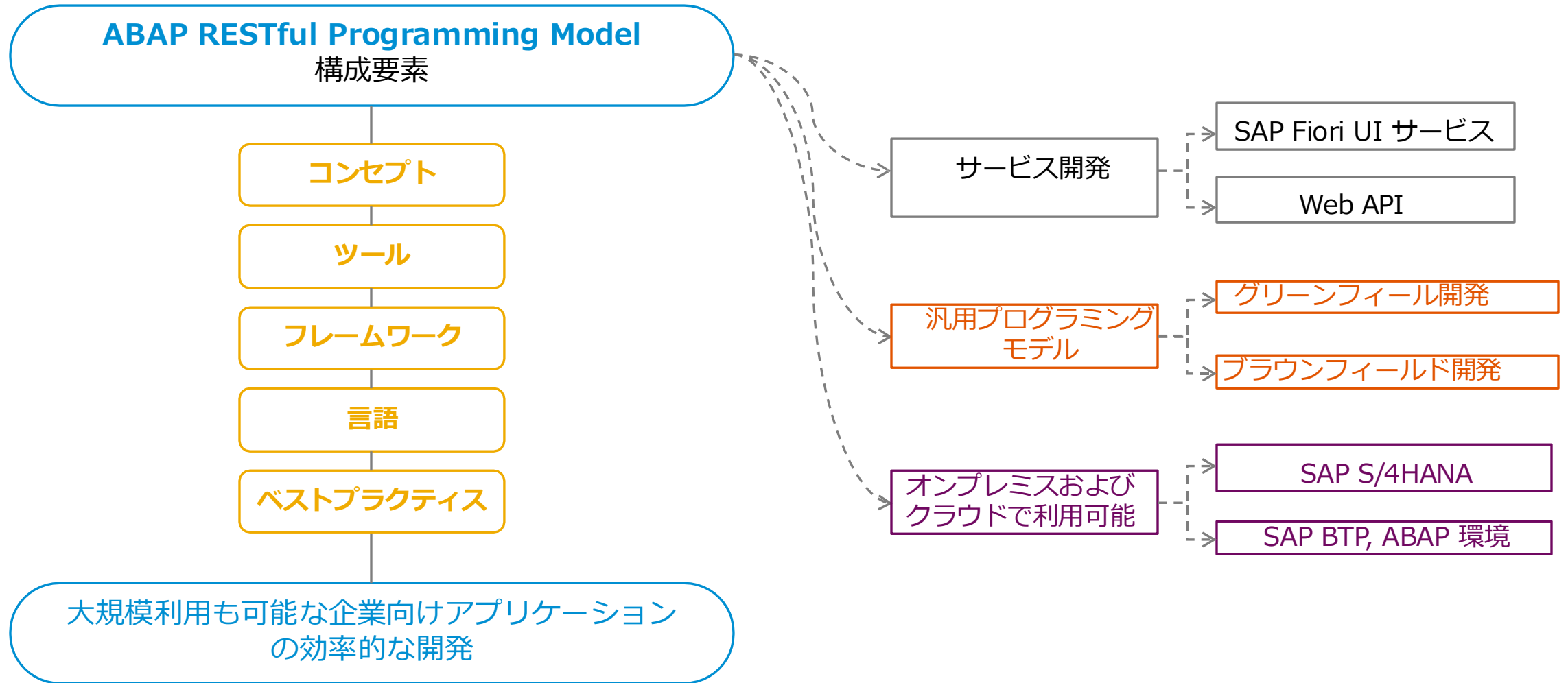
Agenda

- ❑ “クラウドとオンプレにおけるSAP S/4HANAのためのクラウド拡張オプションの”超”概要”
(2023年6月8日開催のパートナー様向けWebiner セッション)の振り返り
- ❑ **開発者拡張の中核技術: SAP RESTful Application Programing Model (RAP)概要**
- ❑ RAP ファサード / RAP BO Interface を用いるための EML (Entity Manipulation Language)
- ❑ RAP における Behavior Definition および Behavior Implementationの実装
- ❑ ご参考情報

SAP S/4HANA Public Cloud における 拡張性オプションの概要

	キーユーザ拡張  ビジネスエキスパート, 導入 コンサルタント, シチズンデベロッパー, キーユーザー	開発者拡張  ABAP 開発者	Side-by-side拡張  ABAP 開発者+ α
シナリオ	SAP S/4HANA アプリケーションの ローコード/ノーコード適応と拡張	SAP S/4HANA データ、トランザクション、 またはアプリとの近接性または結合を必要と するカスタム ABAP 開発プロジェクト	疎結合されたアプリケーションとパートナーの SaaS ソリューション
ユース ケース	<ul style="list-style-type: none"> UI の調整、ユーザ定義項目の追加、 カスタムビジネスオブジェクト の追加など 	<ul style="list-style-type: none"> SAP S/4HANA データへの頻繁または複雑な SQL アクセスを伴うカスタムアプリ ケーション SAP アプリケーションと同じ作業論理単 位 (LUW) で実行されるカスタム拡張 Side-by-Side開発での SAP BTP アプリを 提供する、カスタマイズされたカスタムリ モート API またはサービス 	<ul style="list-style-type: none"> 個別のターゲットグループ向けのカスタムア プリケーション (S/4ユーザ以外が利用) ERP から分離して実行されるカスタムアプリ ケーションワークロード 機械学習や AI などのインテリジェントな SAP BTP サービスに近接する必要があるカス タムアプリケーション 複数の ERP システムおよびクラウドサービス と統合されるソリューション パートナーが提供する SaaS アプリケーション
利点	SAP S/4HANA Cloud で完全に管理 および統合 開発スキルがないか、非常に基本的な スキルのみ必要	SAP S/4HANA Cloud システム内での 拡張の開発 リモートアクセスまたはデータレプリケーシ ョンなし リリース済の SAP S/4HANA Cloud オブジ ェクトの使用および拡張	SAP S/4HANA Cloud の運用およびライフサイ クル管理に依存しない分離された拡張
	On-stack拡張		Side-by-side拡張

ABAP RESTful Application Programming Model (RAP) とは?



RAP のサポート環境

SAP S/4HANA Cloud, ABAP Environment
(code name: Embedded Steampunk)

SAP S/4HANA, ABAP Enviroment (1909~*)

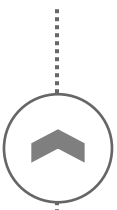
SAP BTP, ABAP Enviroment
(code name: Steampunk)

RAP – 全体像

サービス利用



ビジネスサービス
プロビジョニング



データモデリング
& 動作



SAP FIORI UI
OData UI サービスの利用



WEB API
OData Web API の利用



Service Binding – プロトコルバージョンとシナリオへのバインド



Service Definition – 公開する範囲の定義

Business Object Projection



CDS: プロジェクションビュー



BDEF: 動作プロジェクション



ABAP: 動作実装

Business Object



CDS: データモデリング



BDEF: 動作定義



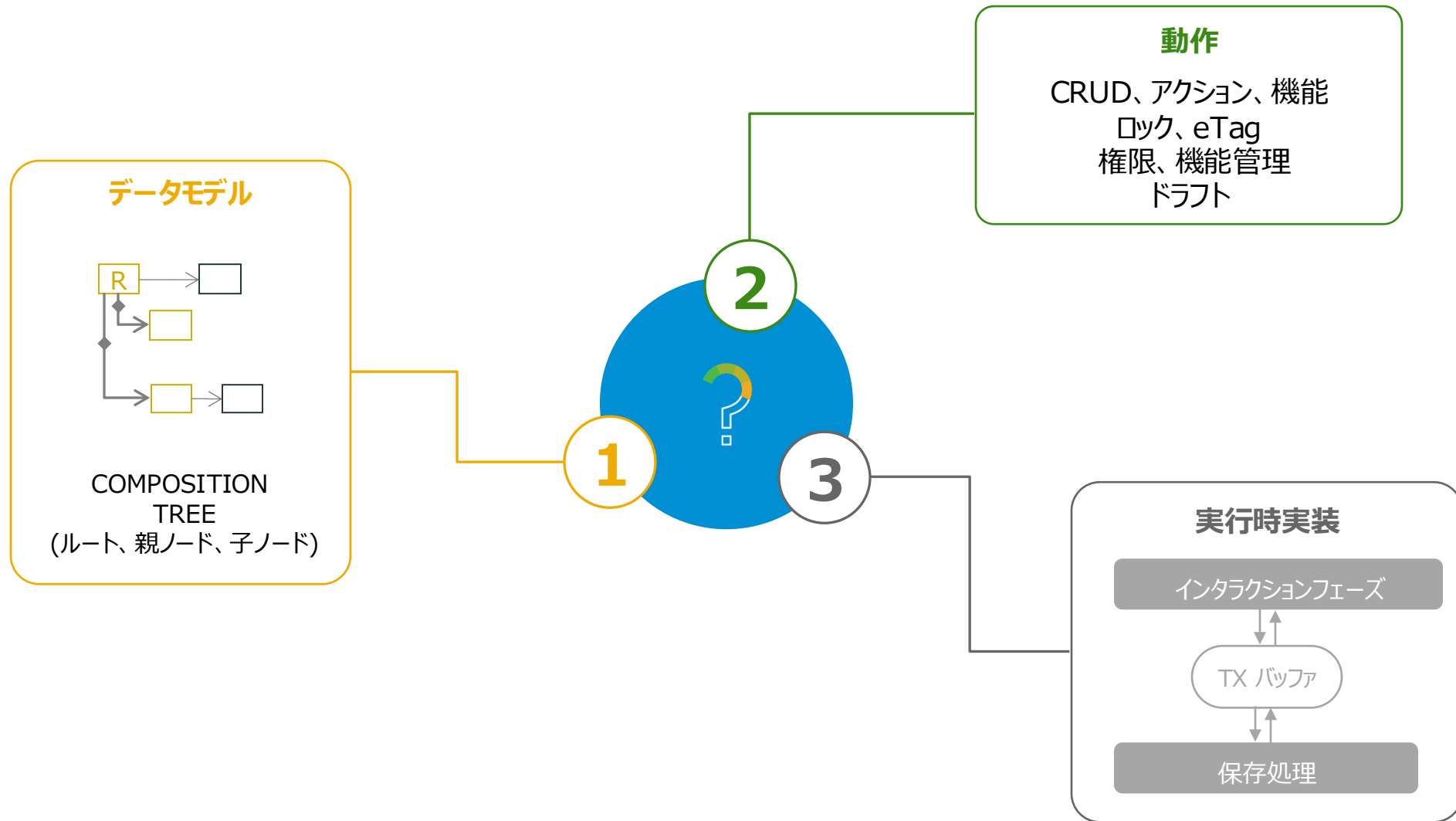
ABAP: 動作実装

Query

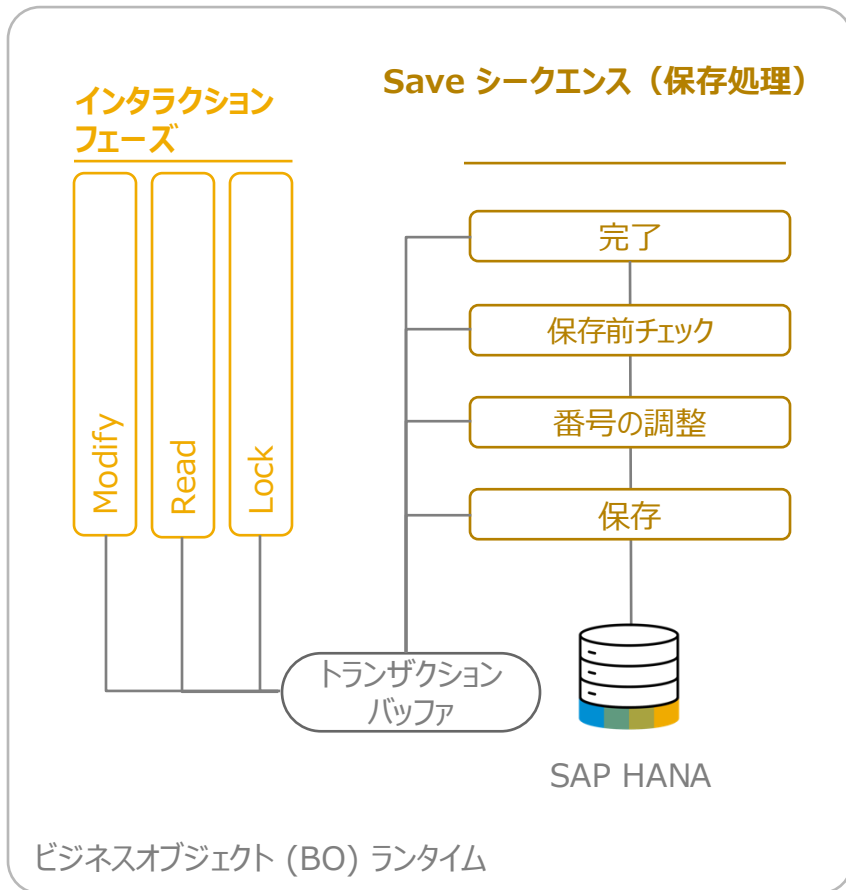


CDS: データモデリング

ビジネスオブジェクトとは？



ビジネスオブジェクト - 実行時実装タイプ



UNMANAGED

アプリケーションコーディングを完全に使用できるブラウンフィールド開発 :
インタラクティブフェーズ + トランザクションバッファ + 保存順序

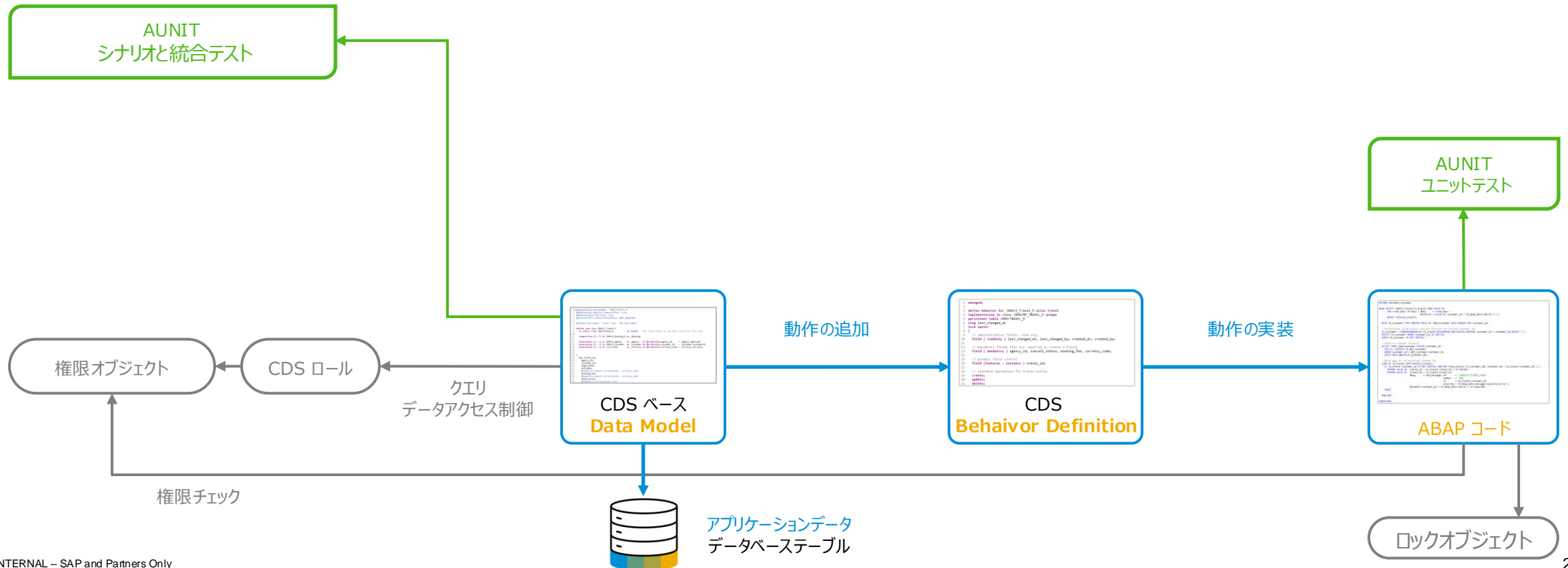
- BO ランタイムを担当する開発者: CRUD 操作
- 既存のコードの統合に必要なアダプタ

MANAGED

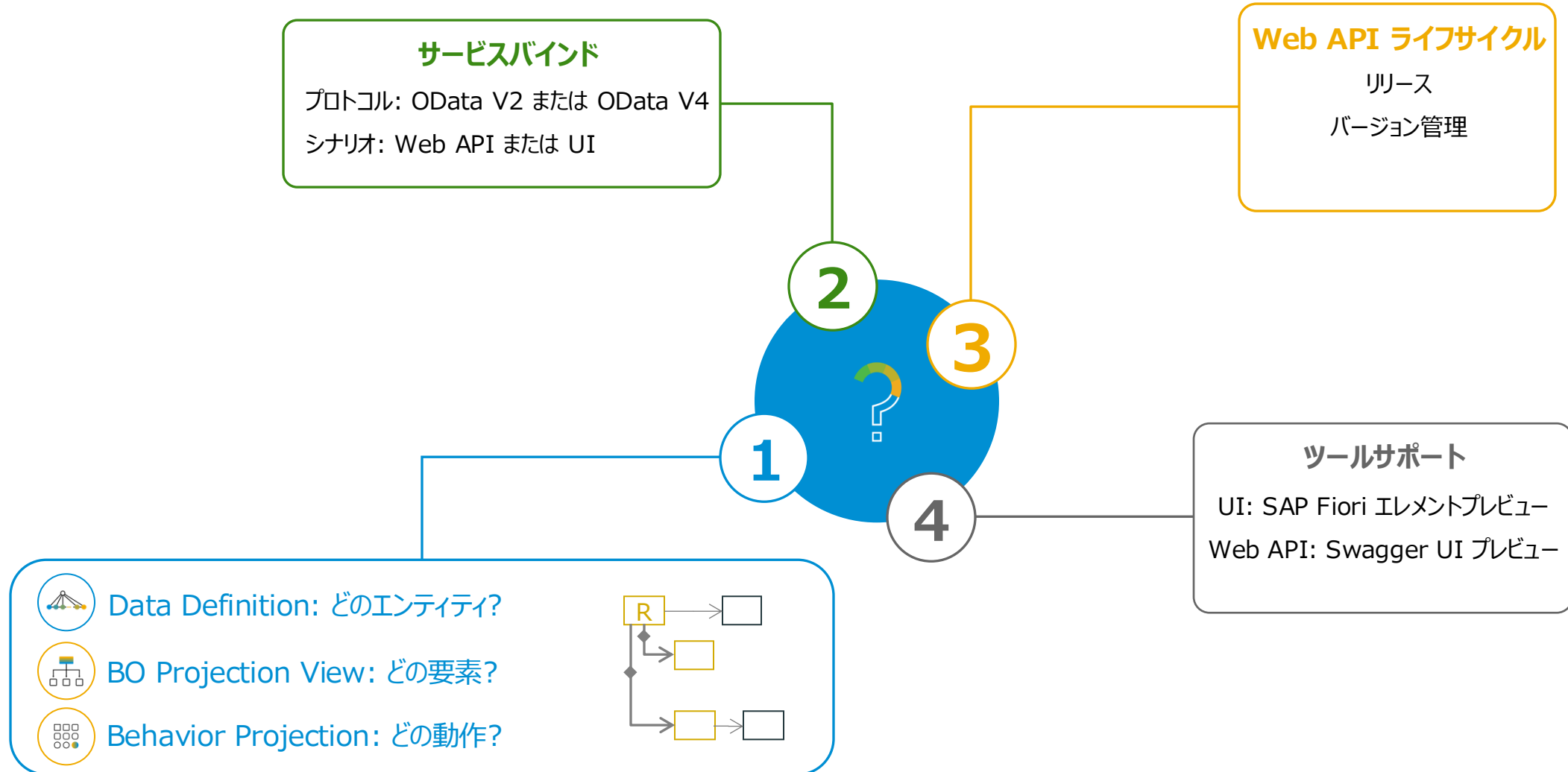
標準実装によるグリーンフィールド開発 (任意管理されていないアプリケーションコンポーネント: DB テーブル、ロック/PFCG オブジェクト、更新タスク FM)

- 即座に利用可能な標準 CRUD 操作
- 開発者による BO 固有ビジネスロジックの追加

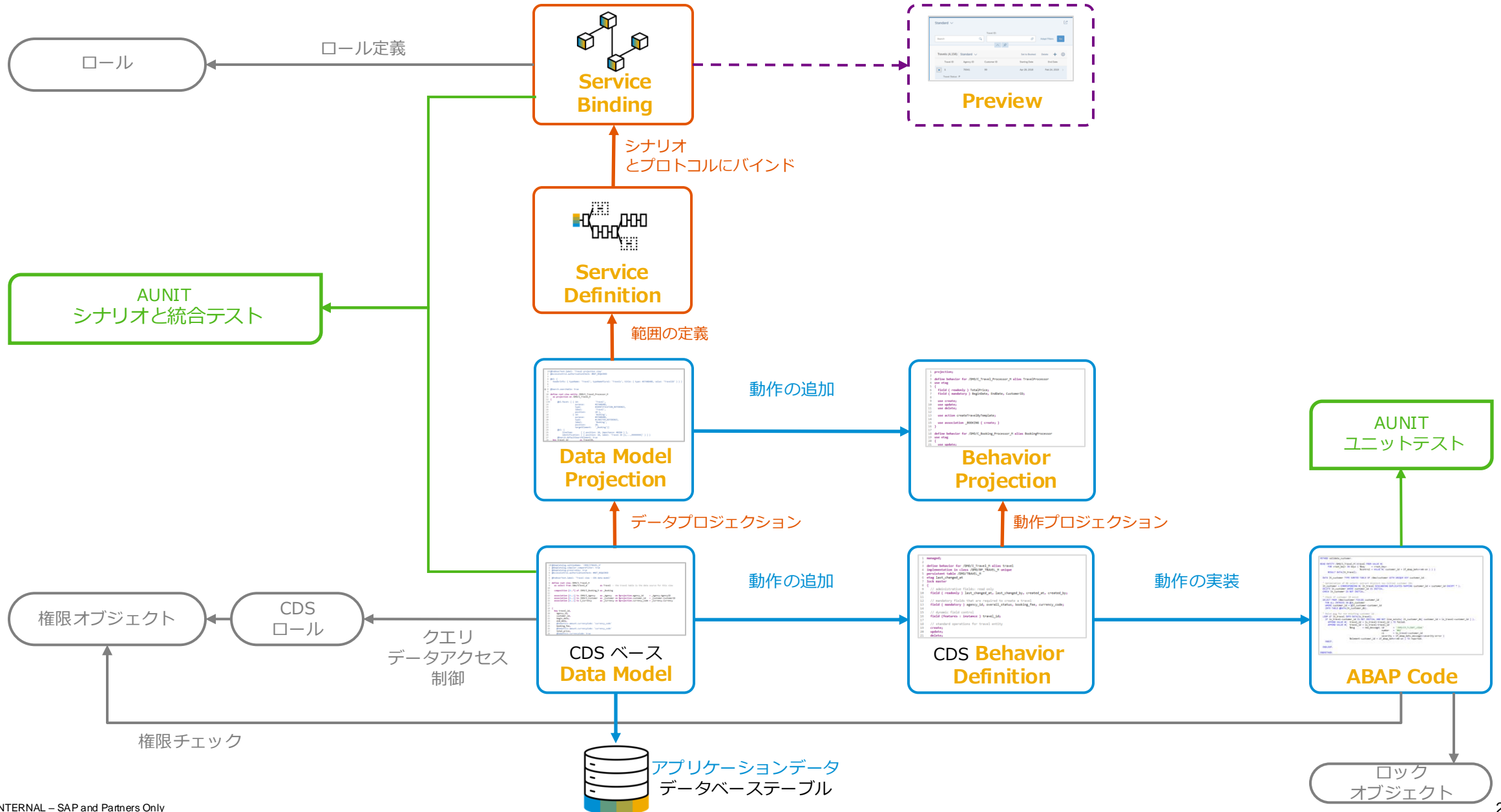
開発フロー



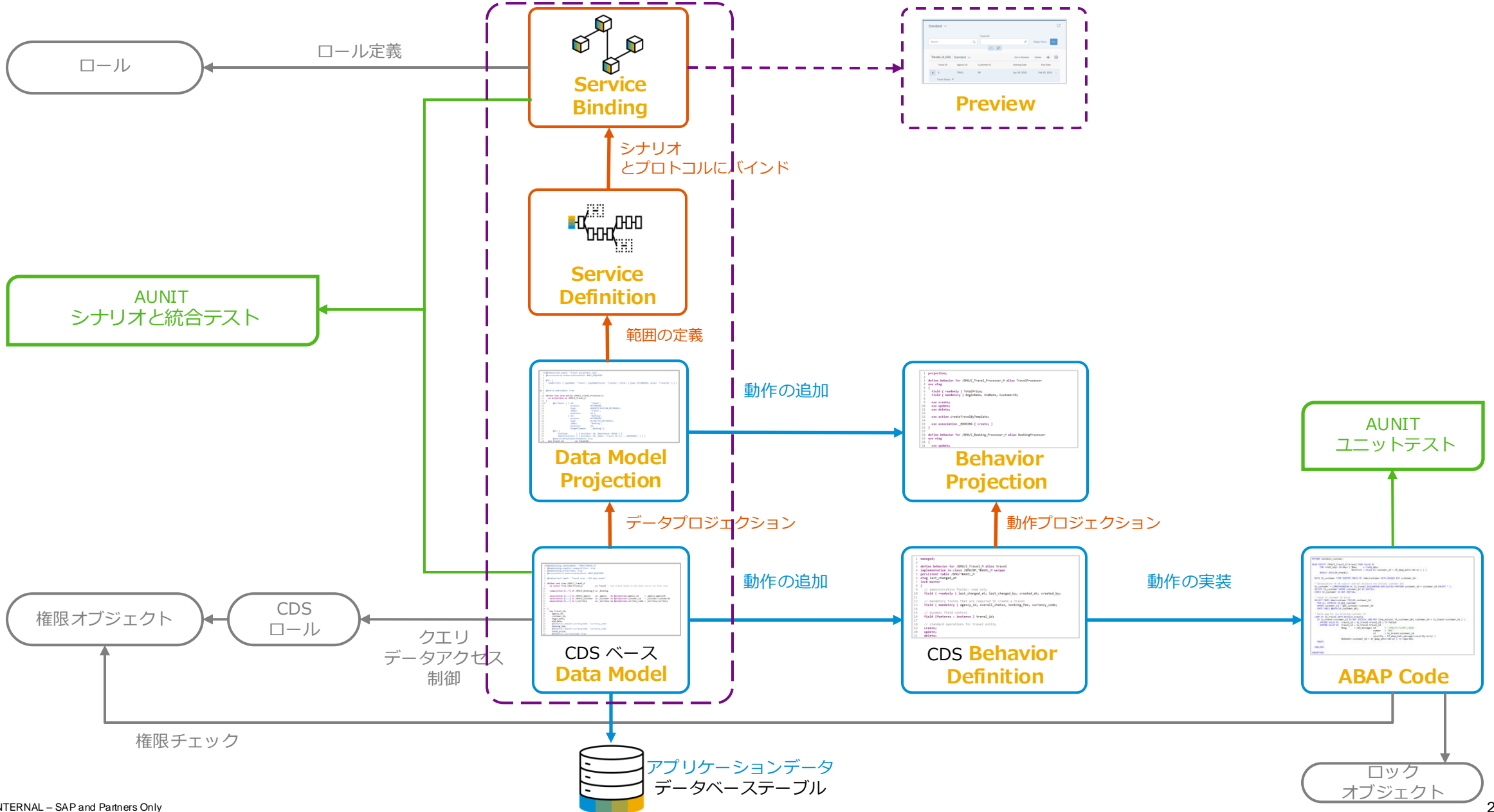
ビジネスサービスとは？



開発フロー



コード解説: 最単純なCDSアプリケーション(READ オペレーションのみの実装)



Data Model –サンプル

```
1 @EndUserText.label: 'Travel data XXX'
2 @AccessControl.authorizationCheck: #CHECK
3 define root view entity ZI_TRAVEL_M_510
4   as select from ztravel 510 as Travel
5
6
7   /* Associations */
8   association [0..1] to /DMO/I_Agency as Agency on $projection.agency_id = Agency.AgencyID
9   association [0..1] to /DMO/I_Customer as _Customer on $projection.customer_id = _Customer.CustomerID
10  association [0..1] to I_Currency as _Currency on $projection.currency_code = _Currency.Currency
11
12 {
13   key mykey,
14     travel_id,
15     agency_id,
16     customer_id,
17     begin_date,
18     end_date,
19     @Semantics.amount.currencyCode: 'currency_code'
20     booking_fee,
21     @Semantics.amount.currencyCode: 'currency_code'
22     total_price,
23     currency_code,
24     overall_status,
25     description,
26
27   /*-- Admin data --*/
28   @Semantics.user.createdBy: true
29     created_by,
30     @Semantics.systemDateTime.createdAt: true
31     created_at,
32     @Semantics.user.lastChangedBy: true
33     last_changed_by,
34     @Semantics.systemDateTime.lastChangedAt: true
35     last_changed_at,
36
37   /* Public associations */
38     _Agency,
39     _Customer,
40     _Currency
41 }
42
43
```

Rootビジネスオブジェクト"Travel"の定義
エンティティ"ztravel_510"からデータ選択

エンティティ"ztravel_510"と他のエンティティの
連携を定義
[0..1] は、それぞれの連携にゼロまたは 1 つのエン
ティティを関連付けることができることを示します。

標準テーブルの"currency_code"を参照する
アノテーション

データ作成したユーザーを保存するアノテーション

データ作成した日付を保存するアノテーション

Data Model Projection – サンプル

```

8  @Search.searchable: true
9  define root view entity ZC_TRAVEL_M_510
10  as projection on ZI_TRAVEL_M_510
11  {
12    @UI.facet: [ { id: 'Travel',
13                  purpose: #STANDARD,
14                  type: #IDENTIFICATION_REFERENCE,
15                  label: 'Travel',
16                  position: 10 } ]
17
18    @UI.hidden: true
19    key mykey as TravelUUID,
20
21
22    @UI: {
23      lineItem: [ { position: 10, importance: #HIGH } ],
24      identification: [ { position: 10, label: 'Travel ID [1,...,99999999]' } ] }
25    @Search.defaultSearchElement: true
26    travel_id as TravelID,
27
28    @UI: {
29      lineItem: [ { position: 20, importance: #HIGH } ],
30      identification: [ { position: 20 } ],
31      selectionField: [ { position: 20 } ] }
32    @Consumption.valueHelpDefinition: [{ entity: {name: '/DMO/I_Agency', element: 'AgencyID' } }]
33
34    @ObjectModel.text.element: ['AgencyName'] ----meaning?
35    @Search.defaultSearchElement: true
36    agency_id as AgencyID,
37    _Agency.Name as AgencyName,
38
39    @UI: {
40      lineItem: [ { position: 30, importance: #HIGH } ],
41      identification: [ { position: 30 } ],
42      selectionField: [ { position: 30 } ] }
43    @Consumption.valueHelpDefinition: [{ entity: {name: '/DMO/I_Customer', element: 'CustomerID' } }]
44    @ObjectModel.text.element: ['CustomerName']
45    @Search.defaultSearchElement: true
46    customer_id as CustomerID,
47
48    @UI.hidden: true
49    _Customer.LastName as CustomerName,
50
51
52    @UI: {
53      lineItem: [ { position: 40, importance: #MEDIUM } ],
54      identification: [ { position: 40 } ] }
55    begin_date as BeginDate,
56
57    @UI: {
58      lineItem: [ { position: 41, importance: #MEDIUM } ],
59      identification: [ { position: 41 } ] }
60    end_date as EndDate,
61
62    @UI: {
63      lineItem: [ { position: 50, importance: #MEDIUM } ],
64      identification: [ { position: 50, label: 'Total Price' } ] }
65    @Semantics.amount.currencyCode: 'CurrencyCode'
66    total_price as TotalPrice,
67
68    @Consumption.valueHelpDefinition: [{ entity: {name: 'I_Currency', element: 'Currency' } }]
69    currency_code as CurrencyCode,
70
71    @UI: {
72      lineItem: [ { position: 60, importance: #HIGH },
73                  { type: #FOR_ACTION, dataAction: 'acceptTravel', label: 'Accept Travel' } ],
74      identification: [ { position: 60, label: 'Status [0(Open)|A(Accepted)|X(Canceled)]' } ] }
75    overall_status as TravelStatus,
76

```

エンティティ検索を可能にするアノテーション

エンティティ全体の属性を定義するアノテーション

項目の属性を定義するアノテーション

項目のフリースタイル検索を可能にするアノテーション

既存バリューヘルプ設定するアノテーション

項目の表示テキストを指定するアノテーション

通貨コードを設定するアノテーション


Service definition / Service binding

1. Service Definition

```
1 @EndUserText.label: 'Service definition for travel '  
2 define service ZUI_C TRAVEL_M_510 {  
3   expose ZC_TRAVEL_M_510 as TravelProcessor;  
4   expose /DM0/I_Customer as Passenger;  
5   expose /DM0/I_Agency as TravelAgency;  
6   expose /DM0/I_Airport as Airport;  
7   expose I_Currency as Currency;  
8   expose I_Country as Country;  
9 }
```

Data Projection "ZC_TRAVEL_M_510"
を"TravelProcessor"として公開

2. Service Binding

 Service Binding: ZUI_C_TRAVEL_M_510

General Information
This section describes general information about this service binding

Binding Type: OData V2 - UI

Service Versions
Define service versions associated with the service binding
type filter text

Version	API State	Service Definition	
1.0.0	Not Released	ZUI_C_TRAVEL_M_510	<div>Add... Remove</div>

TravelProcessor"をOdata V2として
設定

外部インターフェースの作成

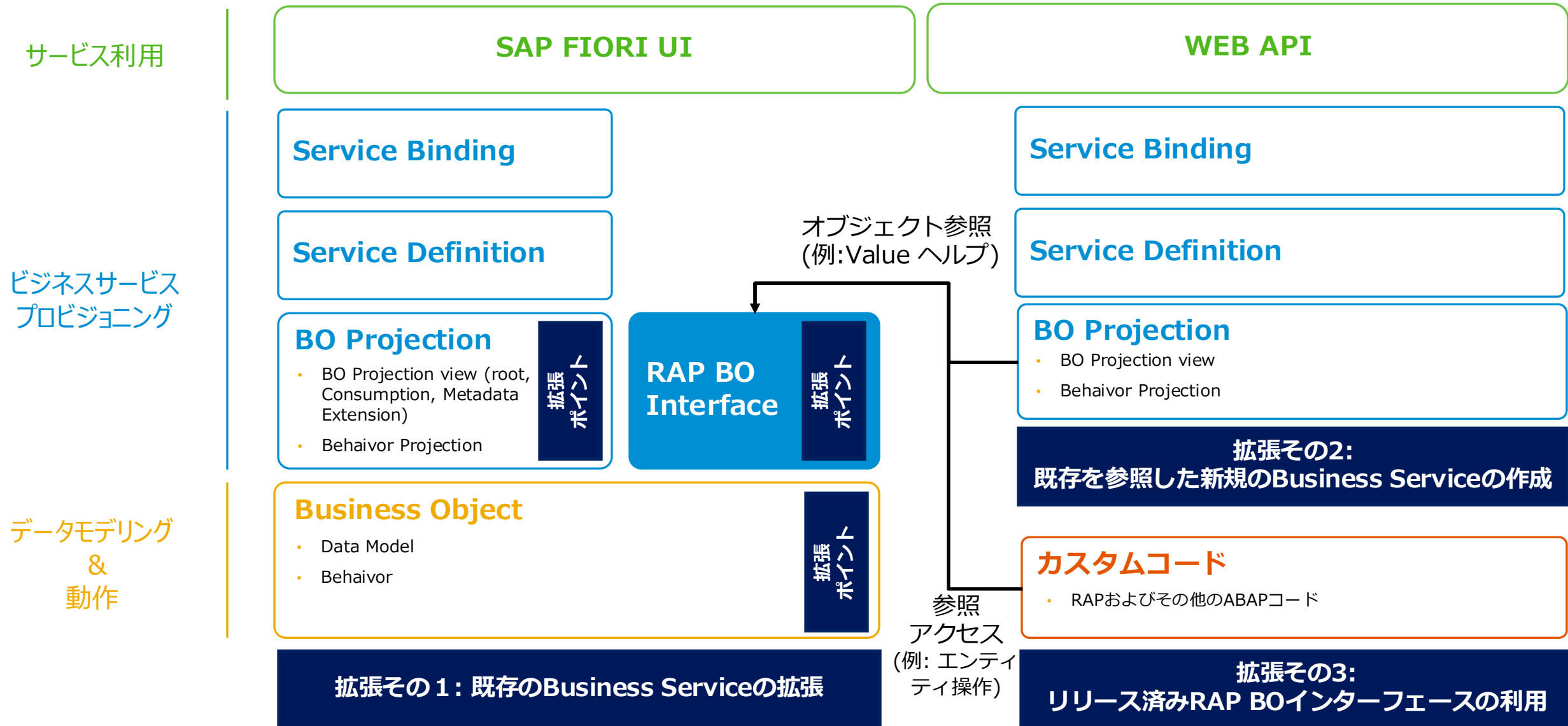
開発者拡張で**RAPベースで作成した外部インターフェース**は、外部APIとして2つのシナリオ(**SAP Fiori UI連携**および**Web API**)で利用できます。さらにそれぞれのシナリオ毎にS/4HANA 標準の外部APIと同様の機能を実装が可能です。(共存は可能)

#	CDS機能	説明	シナリオ1: SAP Fiori UI	シナリオ2: Web API
1	UIアノテーション	Fiori Elements 利用時の各項目の書式、ラベルなどのメタデータ情報 (SAP Fiori Elements ツール で利用)	必須	オプション
2	Value Help	Fiori Elements 利用時の参照するValue Help設定のメタデータ情報 (SAP Fiori Elements ツール で利用)	必須	オプション
3	Dynamic Feature Control	Fiori UI上で特定インスタンスを選択した際に 関連するアクションを特定するための制御	必須	オプション
4	ドラフトテーブル機能	Fiori UIでのデータ保存前のドラフトデータ保持 機能	必須	オプション
5	複合的な\$batchリクエストのための コンテキストID	バルクデータ登録時のコンテキスト特定のための ID	オプション	必須
6	リリース、バージョン管理および非推奨APIの管理	バージョン管理および推奨/非推奨管理	オプション	必須

Agenda

- ❑ “クラウドとオンプレにおけるSAP S/4HANAのためのクラウド拡張オプションの”超”概要” (2023年6月8日開催のパートナー様向けWebiner セッション)の振り返り
- ❑ 開発者拡張の中核技術: SAP RESTful Application Programing Model (RAP)概要
- ❑ **RAP ファサード / RAP BO Interface を用いるための EML (Entity Manipulation Language)**
- ❑ RAP における Behavior Definition および Behavior Implementationの実装
- ❑ ご参考情報

RAP における拡張



SAP S/4HANA Cloud のBusiness API, 拡張ポイントおよびイベント

リモートアクセス	(主に)ローカルアクセス	ローカル拡張ポイント
Side-by-Side 拡張 でのS/4HANA Cloud のビジネスオブジェクトの データにアクセス	On-Stack 拡張 でのS/4HANA Cloud のビジネスオブジェクトの データにアクセス	On-Stack 拡張 でのS/4HANA Cloud のビジネスオブジェクトの ロジックを拡張
<ul style="list-style-type: none">• Odata サービス• SOAP サービス• イベント	<ul style="list-style-type: none">• CDS View***• RAP BO インターフェース (RAP ファサード)• クラス• イベント(計画****)	<ul style="list-style-type: none">• Business Add-Ins (BAdIs)• RAP BO エクステンション

* リリース済みのS/4HANA Cloud のAPIについては製品ドキュメントおよび[SAP Business Accelerator Hub \(旧: SAP API Business Hub\)](#)をご確認ください。

** もしも上記でAPIが未リリースであることがわかった場合は、Customer Influence Channel for SAP S/4HANA Cloud (<https://influence.sap.com/sap/ino/#/campaign/2759>)からリクエストすることが可能です。

*** ODBC接続による外部ツールからにアクセスが可能 ([SAP Help](#))

**** 2023年Q2現在

エンティティ操作言語 (EML) の理解

EML の概要

エンティティ 管理言語

ABAP 言語の拡張
SQL のような構文

RAP コンテキストでのトランザクション BO 動作の管理

RAP BO への API を用いた直接アクセス

標準 EML API
RAP BO へのタイプセーフなREADおよびModifyメソッドを用いたアクセス

一般 EML API
RAP BO の他のフレームワークへの統合

データベース LUW によってサポートされるデータ整合性
変更を保持するには COMMIT 操作が必要

エンティティ操作言語 (EML) の理解

READ オペレーション

RAP BOへの
READアクセス

例

```
READ ENTITIES OF ZI_RAP_Travel_1234
ENTITY travel
  ALL FIELDS WITH VALUE #( ( TravelUUID = '<someUUID>' ) )

RESULT DATA(travels)
FAILED DATA(failed)
REPORTED DATA(reported).
```

エラー処理
(失敗 & レポート済)

エンティティ操作言語 (EML) の理解

MODIFY CREATE オペレーション

例

```
MODIFY ENTITIES OF ZI_RAP_Travel_1234
ENTITY travel
CREATE
SET FIELDS WITH VALUE
  #( ( %cid      = 'MyContentID_1'
      AgencyID   = '70012'
      CustomerID = '14'
      BeginDate  = cl_abap_context_info=>get_system_date( )
      EndDate    = cl_abap_context_info=>get_system_date( ) + 10
      Description = 'I like RAP@openSAP' ) )

MAPPED DATA(mapped)
FAILED DATA(failed)
REPORTED DATA(reported).

COMMIT ENTITIES
RESPONSE OF ZI_RAP_Travel_1234
FAILED DATA(failed_commit)
REPORTED DATA(reported_commit).
```

RAP BO への
MODIFY CREATE
(新規インスタンス作
成)アクセス

エラー処理
(失敗 & レポート済)

適切な LUW 処理
のための
COMMIT 文

エンティティ操作言語 (EML) の理解

MODIFY UPDATE オペレーション

RAP BOへの
MODIFY UPDATE
(更新)アクセス

例

```
MODIFY ENTITIES OF ZI_RAP_Travel_1234
ENTITY travel
UPDATE
  SET FIELDS WITH VALUE
  #( ( TravelUUID = '<someUUID>'
      Description = 'I like RAP@openSAP' ) )

FAILED DATA(failed)
REPORTED DATA(reported).

COMMIT ENTITIES
RESPONSE OF ZI_RAP_Travel_1234
FAILED      DATA(failed_commit)
REPORTED    DATA(reported_commit).
```

エラー処理
(失敗 & レポート済)

適切な LUW 処理
のための
COMMIT 文

エンティティ操作言語 (EML) の理解

MODIFY DELETE オペレーション

RAP BOへの
UPDATE DELETE
アクセス

例

```
MODIFY ENTITIES OF ZI_RAP_Travel_1234
ENTITY travel
DELETE FROM
VALUE
#( ( TravelUUID = '<someUUID>' ) )

FAILED DATA(failed)
REPORTED DATA(reported).

COMMIT ENTITIES
RESPONSE OF ZI_RAP_Travel_1234
FAILED DATA(failed_commit)
REPORTED DATA(reported_commit).
```

エラー処理
(失敗 & レポート済)

適切な LUW 処理
のための
COMMIT 文

エンティティ操作言語 (EML) の理解

MODIFY EXECUTE ACTION オペレーション

RAP BO に対する
特定のAction (標準オ
ペレーション以外の処
理)の実行

例

```
MODIFY ENTITIES OF ZI_RAP_Travel_1234
ENTITY travel
EXECUTE acceptTravel
FROM VALUE
#( ( TravelUUID = '<someUUID>' ) )

RESULT DATA(result)
MAPPED DATA(mapped)
FAILED DATA(failed)
REPORTED DATA(reporting).

COMMIT ENTITIES
RESPONSE OF ZI_RAP_Travel_1234
FAILED DATA(failed_commit)
REPORTED DATA(reporting_commit).
```

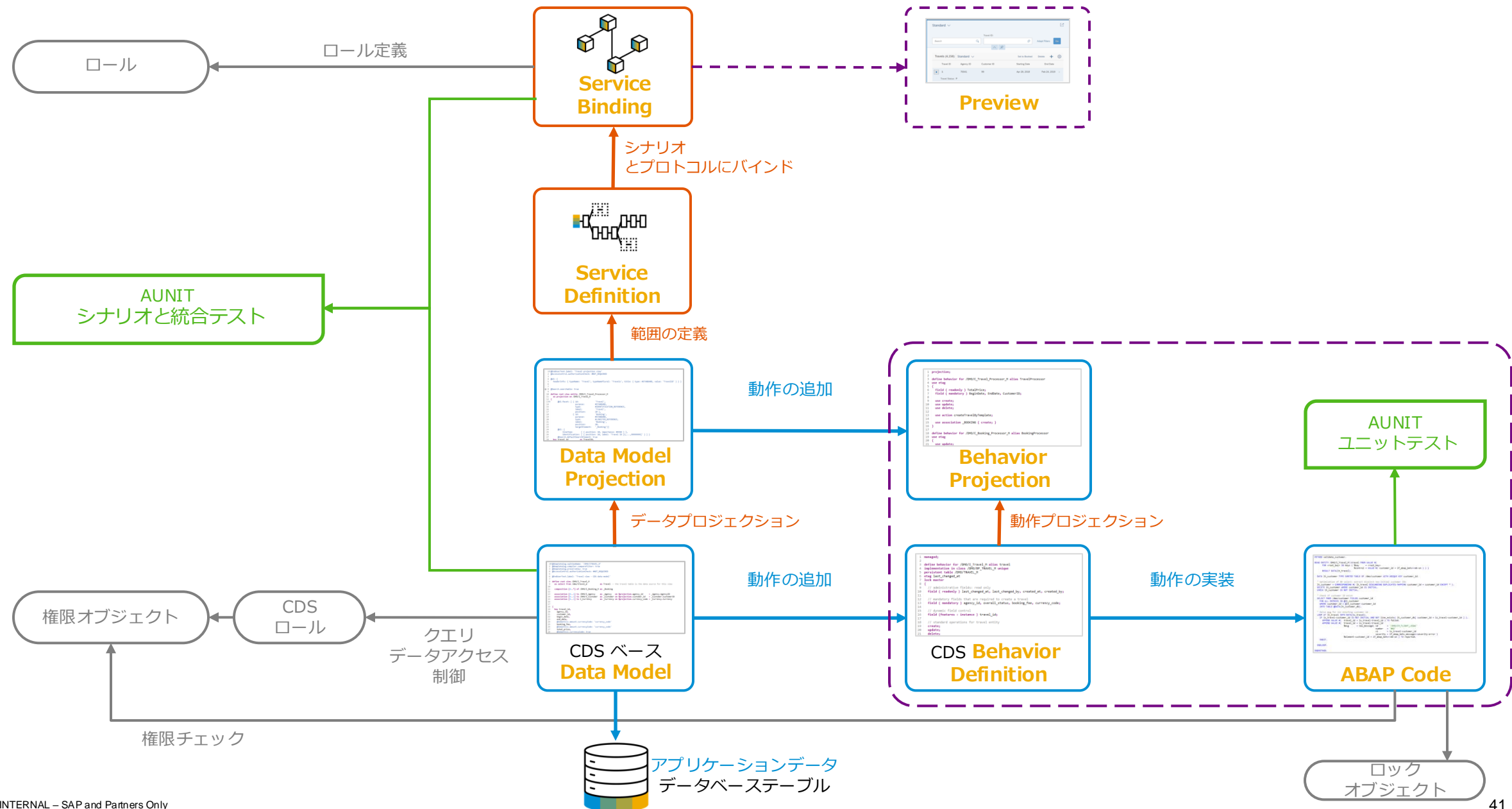
エラー処理
(失敗 & レポート済)

適切な LUW 処理
のための
COMMIT 文

Agenda

- ❑ “クラウドとオンプレにおけるSAP S/4HANAのためのクラウド拡張オプションの”超”概要” (2023年6月8日開催のパートナー様向けWebiner セッション)の振り返り
- ❑ 開発者拡張の中核技術: SAP RESTful Application Programing Model (RAP)概要
- ❑ RAP ファサード / RAP BO Interface を用いるための EML (Entity Manipulation Language)
- ❑ **RAP における Behavior Definition および Behavior Implementationの実装**
- ❑ ご参考情報

コード解説: データの更新を含むBO機能の実装



Behavior definition –サンプル

```
1 managed implementation in class ZBP_I_TRAVEL_M_510 unique;
2
3 define behavior for ZI_TRAVEL_M_510 alias Travel
4 persistent table ztravel_510
5 etag master last_changed_at
6 lock master
7 {
8   // key that will be automatically generated by the framework
9   field ( readonly, numbering : managed ) mykey;
10
11   // semantic key calculated in a determination
12   field ( readonly ) travel_id;
13
14   // administrative fields (read only)
15   field ( readonly ) last_changed_at, last_changed_by, created_at, created_by
16
17   // mandatory fields that are required to create a travel
18   field ( mandatory ) agency_id, overall_status, booking_fee, currency_code;
19
20   // mandatory fields that are required to create a travel
21   field ( mandatory ) Begin_Date, End_Date, Customer_ID;
22
23   // standard operations for travel entity
24   create;
25   update;
26   delete;
27
28   // instance action and dynamic action control
29   action ( features : instance ) acceptTravel result [1] $self;
30
31   // validations
32   validation validateCustomer on save { field customer_id; }
33   validation validateDates on save { field begin_date, end_date; }
34   validation validateAgency on save { field agency_id; }
35
36   // determination
37   determination CalculateTravelKey on modify { create; }
38
39 }
40
41 }
```

myKey, travel_id などのビジネスオブジェクトの管理項目をReadonly に設定

myKey項目はフレームワークが自動的にセット

その他の入力項目を必須項目に設定

Create, update, deleteといったスタンダードなオペレーションを利用を宣言

標準ではないオペレーションであるActionを定義。
Feature: instanceは特定のインスタンスを選択した場合に動作。\$selfはパラメータしてインスタンスをAction に受け渡す。

Validation はビジネスオブジェクトがOn Saveの場合に動作する機能。文字通り値のチェックで利用

Determinationは値がOn Modify のオペレーションの中で動作

Behavior Projection –サンプル

```
1 projection;  
2 //strict ( 1 ); //Uncomment this line in order to enable st  
3  
4 define behavior for ZC_TRAVEL_M_510 alias TravelProcessor  
5 use etag  
6 {  
7     use create;  
8     use update;  
9     use delete;  
10  
11     use action acceptTravel;  
12 }
```

Behavior Implementation –サンプル

→ 2.Save シーケンス(On Save)で動作

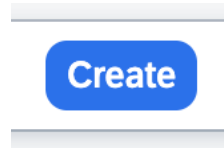
```
1  """ use this source file for the definition and implementation of
2  """ local helper classes, interface definitions and type
3  """ declarations
4
5  CLASS lhc_travel DEFINITION INHERITING FROM cl_abap_behavior_handler.
6
7      PRIVATE SECTION.
8
9          TYPES tt_travel_update TYPE TABLE FOR UPDATE zi_travel_m_510.
10
11         METHODS validate_customer      FOR VALIDATE ON SAVE IMPORTING keys FOR travel~validateCustomer.
12         METHODS validate_dates         FOR VALIDATE ON SAVE IMPORTING keys FOR travel~validateDates.
13         METHODS validate_agency       FOR VALIDATE ON SAVE IMPORTING keys FOR travel~validateAgency.
14
15         METHODS set_status_completed  FOR MODIFY IMPORTING keys FOR ACTION travel~acceptTravel RESULT result.
16         METHODS get_features          FOR FEATURES IMPORTING keys REQUEST requested_features FOR travel RESULT result.
17
18         METHODS CalculateTravelKey    FOR DETERMINE ON MODIFY IMPORTING keys FOR Travel~CalculateTravelKey.
19
20
21     ENDCLASS.
22
```

→ 1.Create オペレーション (On Modify)で動作

3.Actionで動作 ←

Behavior Implementation – Modify Create オペレーション

Create 時のランタイムダイアグラム



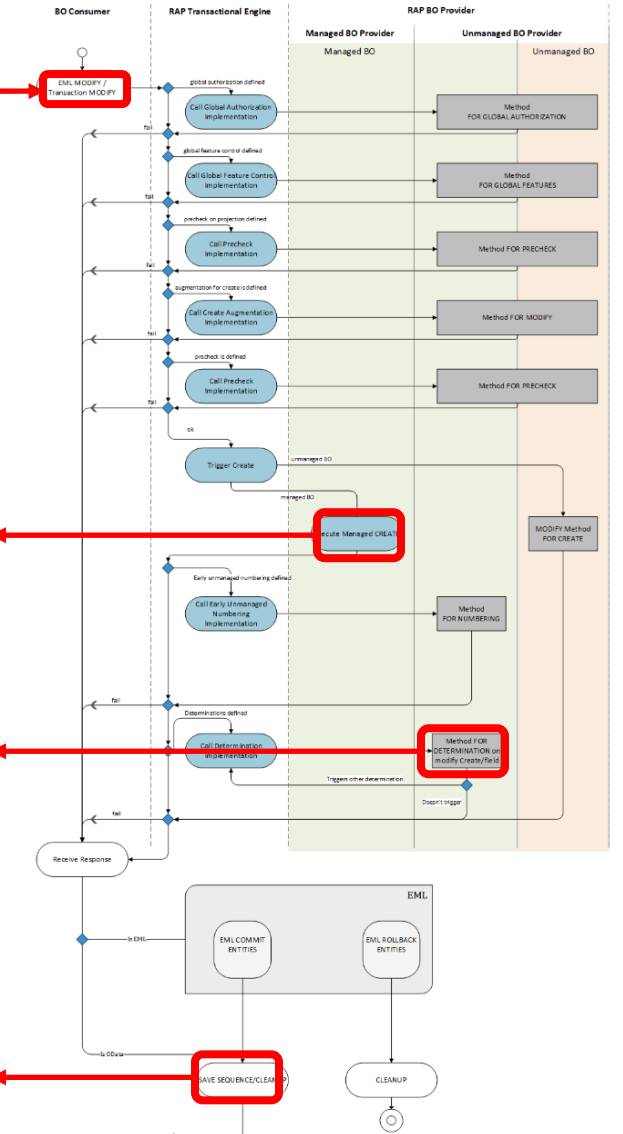
1. Modify Create
オペレーション起動

2. Managed Create
処理実行

3. On Modify 時の
Determination 処理実行

4. Save シークエンス
への移行 (次ページへ)

METHODS CalculateTravelKey



Behavior Implementation – Save シーケンス

Save シーケンスのランタイムダイアグラム

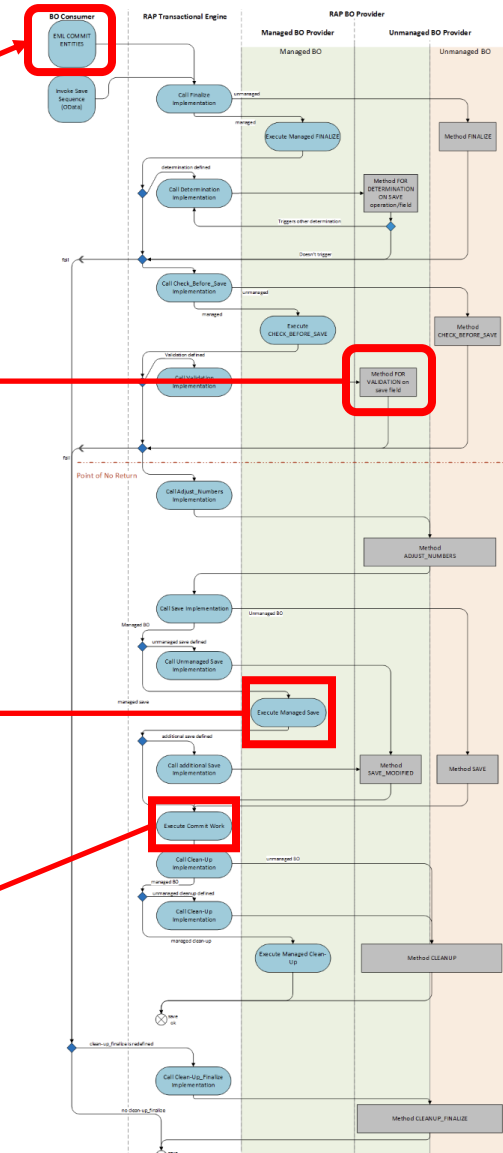
```
METHODS validate_customer
METHODS validate_dates
METHODS validate_agency
```

1. Save シークエンス (前ページより)

2. On Save 時の Validation 処理実行

3. Managed Save 処理実行

4. Commit Work 処理実行



Behavior Implementation – Action

Actionのランタイムダイアグラム

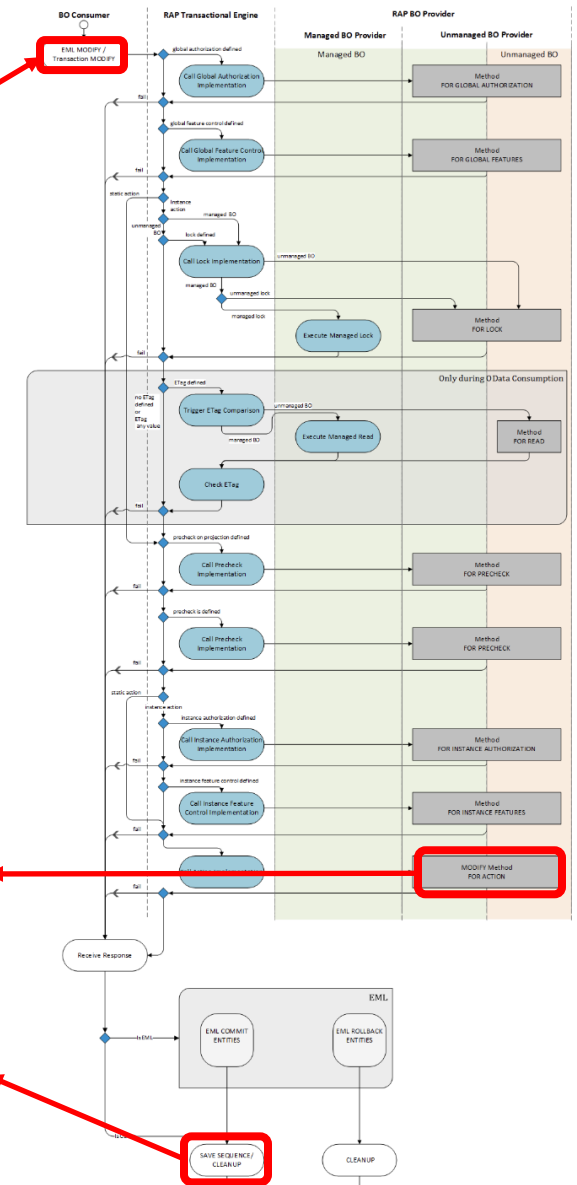
Accept Travel

1. Action起動

`METHODS set_status_completed`

2. Action処理実行

3. Save シークエンス
への移行



Agenda

- ❑ “クラウドとオンプレにおけるSAP S/4HANAのためのクラウド拡張オプションの”超”概要” (2023年6月8日開催のパートナー様向けWebiner セッション)の振り返り
- ❑ 開発者拡張の中核技術: SAP RESTful Application Programing Model (RAP)概要
- ❑ RAP ファサード / RAP BO Interface を用いるための EML (Entity Manipulation Language)
- ❑ RAP における Behavior Definition および Behavior Implementationの実装
- ❑ **ご参考情報**

Agenda

- ❑ “クラウドとオンプレにおけるSAP S/4HANAのためのクラウド拡張オプションの”超”概要”
(2023年6月8日開催のパートナー様向けWebiner セッション)の振り返り
- ❑ 開発者拡張の中核技術: SAP RESTful Application Programing Model (RAP)概要
- ❑ RAP ファサード / RAP BO Interface を用いるための EML (Entity Manipulation Language)
- ❑ RAP における Behavior Definition および Behavior Implementationの実装
- ❑ **ご参考情報**

ご参考情報

- SAP Community: RAP 関連情報のリンク集 (解説、ツール、ビデオおよび学習コンテンツのリンクなど)
 - <https://pages.community.sap.com/topics/abap/rap>
- 本資料作成のために引用したSAP RESTful Application Programming Model (RAP)のコード解説SAP Blog
 - Get Know about RAP : [Introduction](#)
 - Get Know about RAP : [Define Data Model – Part 1](#)
 - Get Know about RAP : [Define CDS-based data model – Part 2](#)
 - Get Know about RAP : [CDS Behavior Definition](#)
 - Get Know about RAP : [Enhance Behavior with action and validation](#)
- SAP Tutorial: RAPでのビジネスオブジェクト開発からFiori Element までの開発のチュートリアル (初級編 & 中級編)
 - [Build an SAP Fiori Elements App Using the SAP RESTful Application Programming Model \(RAP\)– Beginners](#)
 - [Build an SAP Fiori Elements App Using the SAP RESTful Application Programming Model \(RAP\) – Intermediate](#)
- SAP Help – Behavior definition/Metadata Extension 関連のアノテーション
 - [UI Annotation](#)
- SAP Help - Behavior implementation関連
 - [Operations](#) – managed operation やunmanaged operationなど
 - Managed operation に関連処理 ([Determinations](#), [Validations](#), [Save Options](#)など)
 - [Action](#) – Modify 処理を定義する標準ではないオペレーション
 - [Entity Manipulation Language \(EML\)](#)

補足情報

- ❑ RAPを用いたExcel Upload の開発(SAP Blog)
 - ❑ <https://blogs.sap.com/2022/01/19/excel-upload-using-rap-part-1/>
- ❑ RAPのLOCK機構について (SAP Help)
 - ❑ <https://help.sap.com/docs/btp/sap-abap-restful-application-programming-model/concurrency-control?locale=en-US>
 - ❑ <https://help.sap.com/docs/btp/sap-abap-restful-application-programming-model/update-operation-runtime?locale=en-US>
- ❑ RAP BO Interface Product Description(Create オペレーションがない (Business Accelerator Hub))
 - ❑ https://api.sap.com/bointerface/I_PRODUCTTP_2
- ❑ EMLのModify Entity の書式 (SAP Help)
 - ❑ https://help.sap.com/doc/abapdocu_cp_index.htm/CLOUD/en-US/index.htm?file=abapmodify_entity_entities.htm
- ❑ If_oo_adt_classrunのSAP Blog (SAP Blog)
 - ❑ <https://blogs.sap.com/2021/02/01/printing-to-abap-console/>
- ❑ Draft データの削除 (SAP Help)
 - ❑ <https://help.sap.com/docs/btp/sap-abap-restful-application-programming-model/deleting-instances-of-draft-bo?locale=en-US&version=Cloud>
- ❑ SAP Tutorial : Develop an SAP Fiori App to Trigger Purchase Requisitions API (応用編チュートリアル。RAP作成からFiori App 作成を経てS/4HANA環境へデプロイするシナリオ)
 - ❑ <https://developers.sap.com/group.sap-fiori-app-purchase-req.html>

ご参考資料:

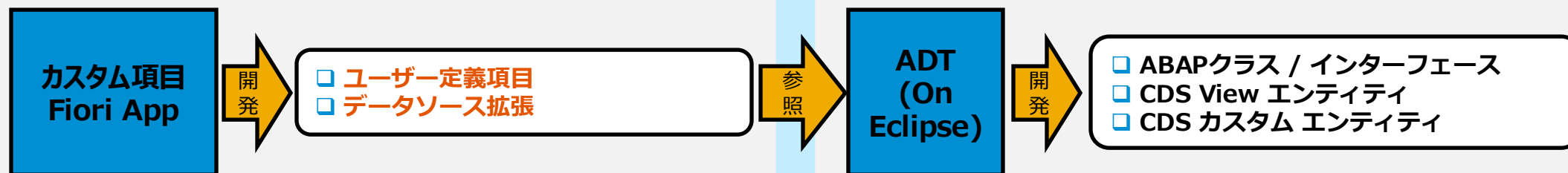
Key-User拡張オブジェクトと開発者拡張オブジェクトの相互アクセス

S/4HANA

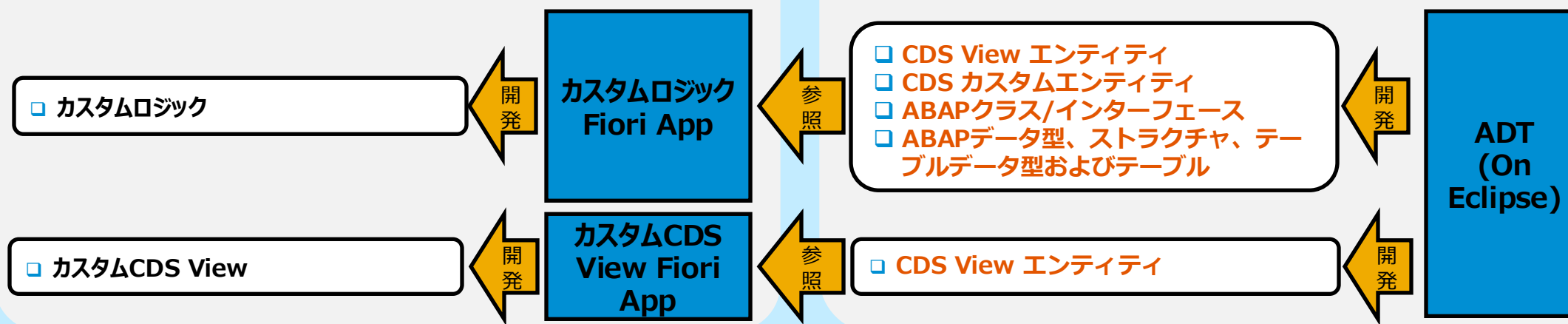
Key-User拡張テナント

開発者拡張テナント

1. 開発者拡張(ADT)からIn-App拡張オブジェクトの参照



2. Key-User拡張ツールから開発者拡張オブジェクトの参照*

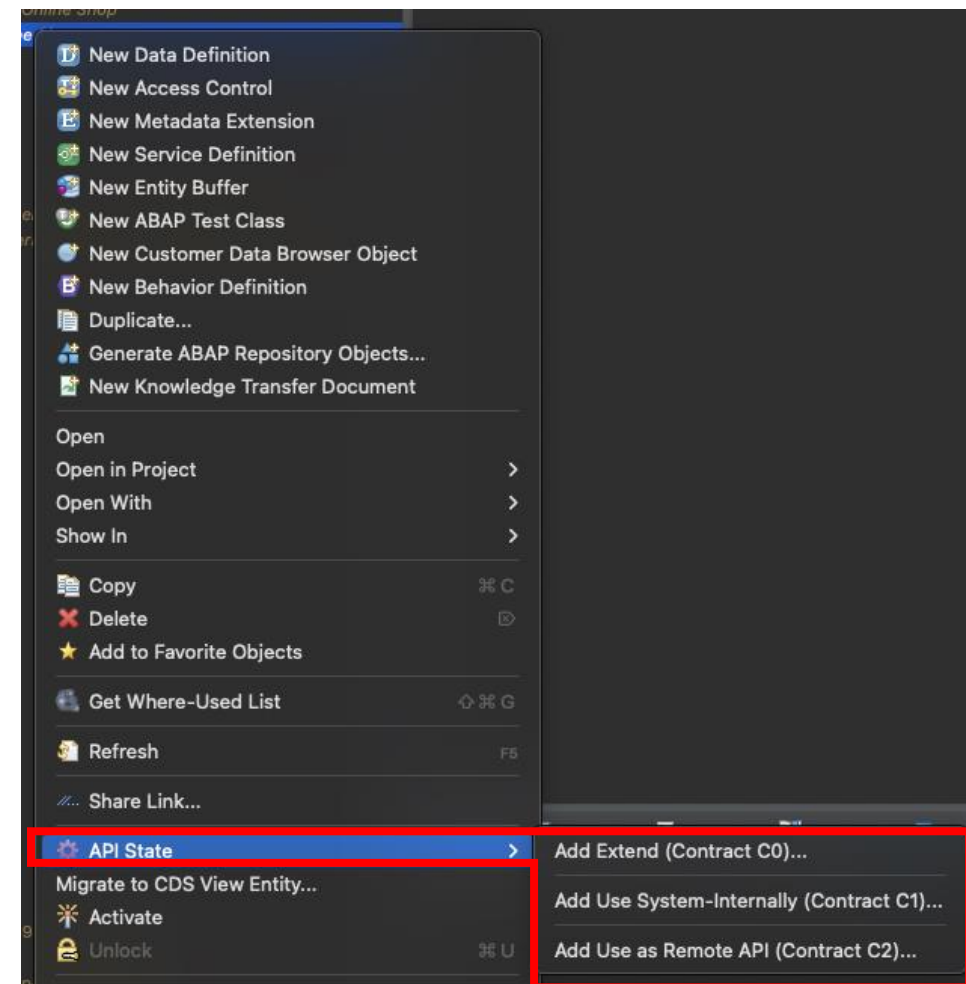


*ただしADT上でそれぞれのオブジェクトに対して"API State" を"Use System-Internally(C1)"に設定することでKey-User拡張で利用できるように上でリリースする必要があります。(詳細は[こちら](#))

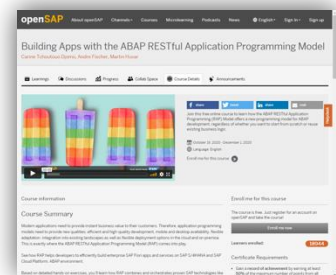
ご参考資料: 開発者拡張オブジェクトのAPI Stateについて

オブジェクトの種類ごとに関連するRelease Contractの中から用途に応じて適切な用途を選択

Release Contract	説明	対象オブジェクト
Extend (C0)	拡張が可能なAPIの専用の安定した拡張ポイント	例: BO Type → それ以外のオブジェクト
Use System-Internally(C1)	In-App拡張で利用するための安定したパブリックインターフェース	例: 権限項目, 権限オブジェクト → それ以外のオブジェクト
Use as Remote API (C2)	リモートAPIとして利用するための安定したパブリックインターフェース	例: CDSエンティティ → それ以外のオブジェクト
Manage Configuration Content (C3)	専用のAPIとエディタを利用したインポート/エクスポート/表示および編集可能な独自の設定コンテンツのパースタンス	例: データベーステーブル → その他詳細
Use in ABAP-Managed Database Procedures (C4)	ABAP-Managed Database Procedures (AMDP)で利用するためのパブリックインターフェース	例: AMDP関連のBAdI定義 → それ以外のオブジェクト
Deprecation	非推奨の開発オブジェクト	例: CDSエンティティなど多数 → それ以外のオブジェクト



ADT上でのAPI State設定



無料の SAP Learning コース



Building Apps with the ABAP RESTful Application Programming Model (英語コース)



セルフペースモード



- 第 1 週目：導入
- 第 2 週目：読み取り専用一覧レポートアプリの開発
- 第 3 週目：アプリの開発
- 第 4 週目：既存のコードトランザクション動作の有効化
- 第 5 週目：サービス利用と Web API を処理
- 第 6 週目：最終試験



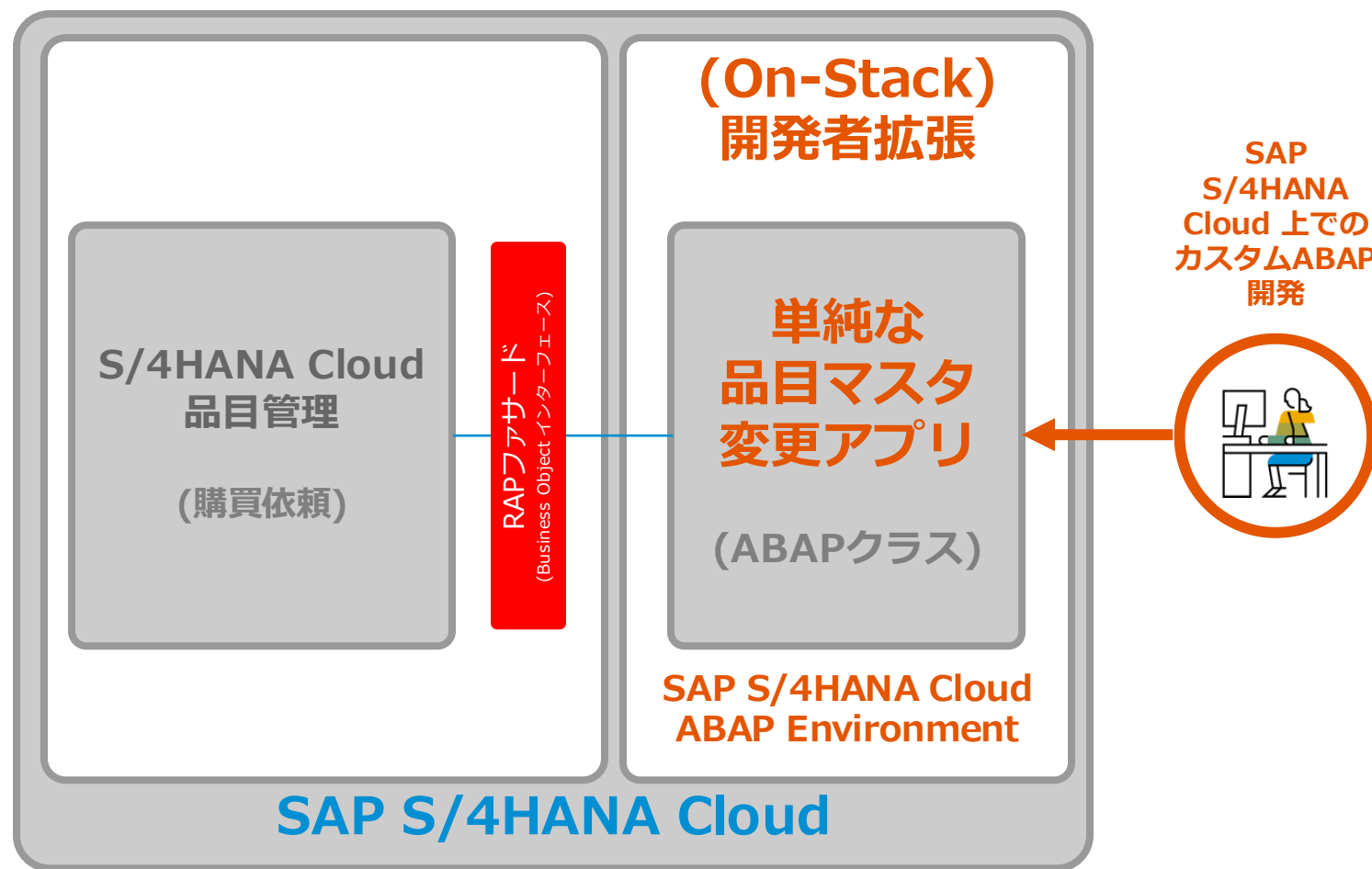
今すぐ受講！

<https://open.sap.com/courses/cp13>



演習 1

S/4HANA Cloud, public edition 拡張のシンプルなABAPクラスへの実装

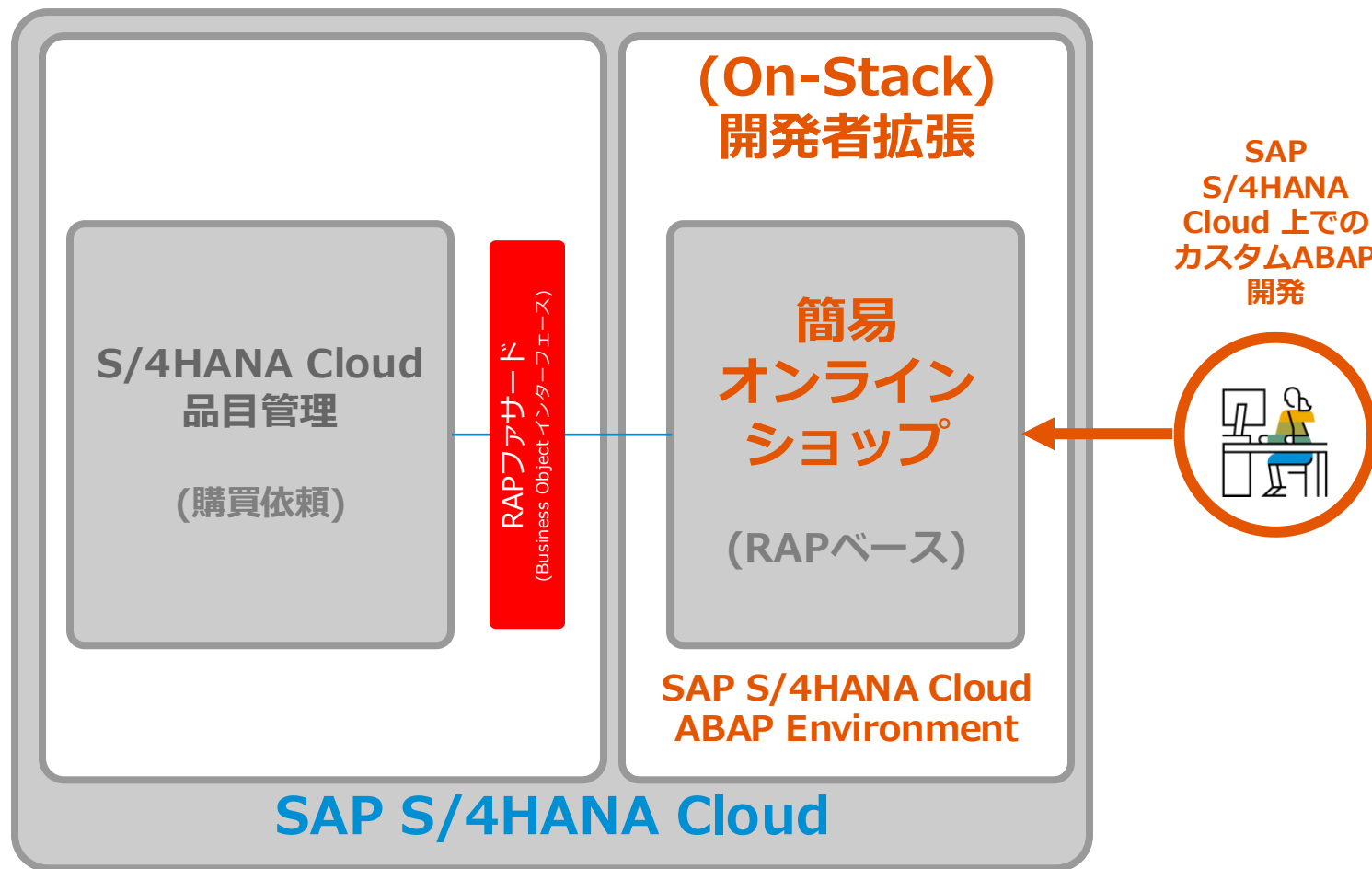


本演習での習得ポイント

- ✓ ADTによるSAP S/4HANA Cloud へのアクセス
- ✓ ABAP Cloud Package の作成
- ✓ SAP S/4HANA Cloud の標準オブジェクトとの連携 (RAP ファサードの利用)

演習2

S/4HANA Cloud, public edition 拡張 + RAPを用いた 簡易オンラインショップアプリの作成



本演習での習得ポイント

- ✓ データベースの作成
- ✓ CDSモデルおよびProjection View 作成
- ✓ Behavior 定義および実装
- ✓ Service 定義とバインディング
- ✓ インプットヘルプの作成
- ✓ SAP Fiori アプリによるプレビュー方法

理解度テスト

本セッションの最後にWebテストを実施します。

Web テストを受けてバッジを獲得しましょう！

- ❑ 要SAP Universal IDもしくはSユーザーID
- ❑ 日本語/英語選択可能
- ❑ 80%以上正解で合格&バッジ獲得
- ❑ 何度でも受講可能



理解度テスト(Git Hub のリンクもしくは→から)



質問	回答
演習3で受信用(Inbound)サービスを生成する際に作成されたService binding はどのタイプのものですか？	OData V2 – Web API
演習3でS/4HANA上の管理者としてCommunication arrangement を登録する際に参照される開発オブジェクトは何ですか？	Custom Communication Scenario
演習3でOnline Shop Web APIを呼び出すためにSAP BTP ABAP Environment で作成されるオブジェクトは何ですか？	Service Consumption Model
演習3でOnline Shop Web APIのOdataサービスから作られ、最終的にSAP BTPから公開されるサービスはどのような種類のサービスですか？	HTTP Service
ABAP Test Cockpit でカスタムコードのABAP Cloud への適応可否を確認する際に利用されるバリエーションはどれですか？	ABAP_CLOUD_READINESS
バリエーション ABAP_CLOUD_READINESSでチェックできる項目はどれですか？2つ選択してください。	リリースされていないAPIの利用 ABAP言語バージョン5以上の適応
ABAP Cloud を利用する場合、テーブルMARAを利用することはできません。その理由は…	ABAP Cloud で利用するためにテーブルMARAはリリースされていないためです。

SAP S/4HANA Cloud, public edition ABAP拡張ハンズオンワークショップ



- 本日はご参加ありがとうございました。
本セッションのアンケートへのご協力をお願いいたします。
今後の参考にさせていただきますので、ご意見をお聞かせください。
<https://jp.surveymonkey.com/r/ABAPworkshop20241126>
- 資料は、後日事務局からご案内させていただきます。

ご清聴ありがとうございます！

コンタクト先:

Speaker's Name

Email