

ABAP Cloud

A Life after the Report

Agenda

1. ABAP Cloud
2. Possibilities
3. Hands-On
4. Summary

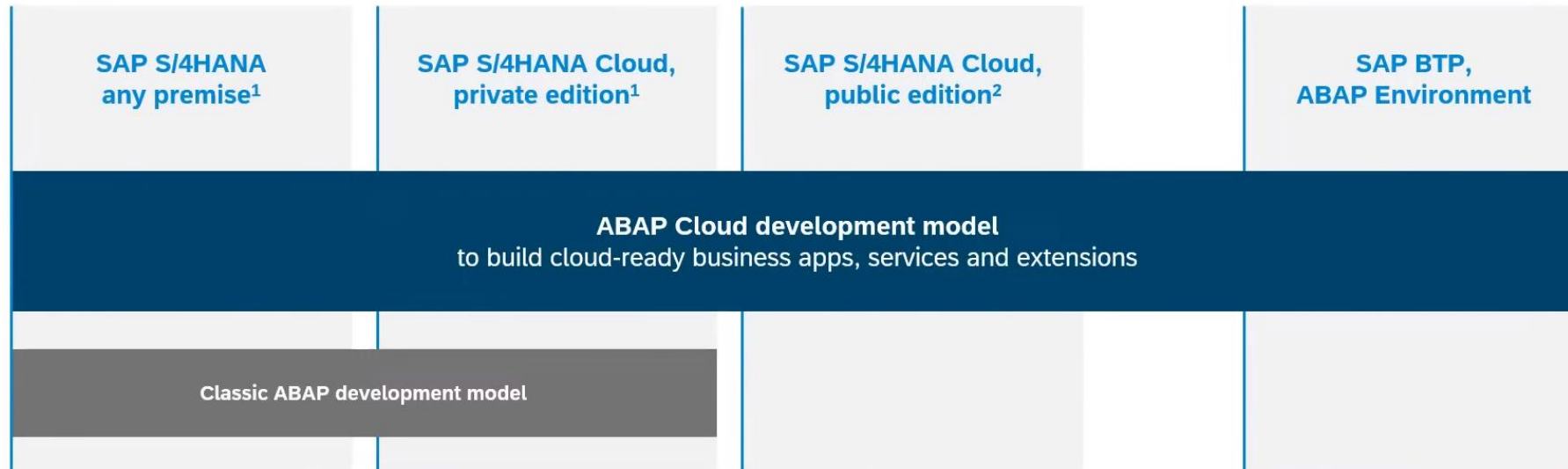
ABAP Cloud



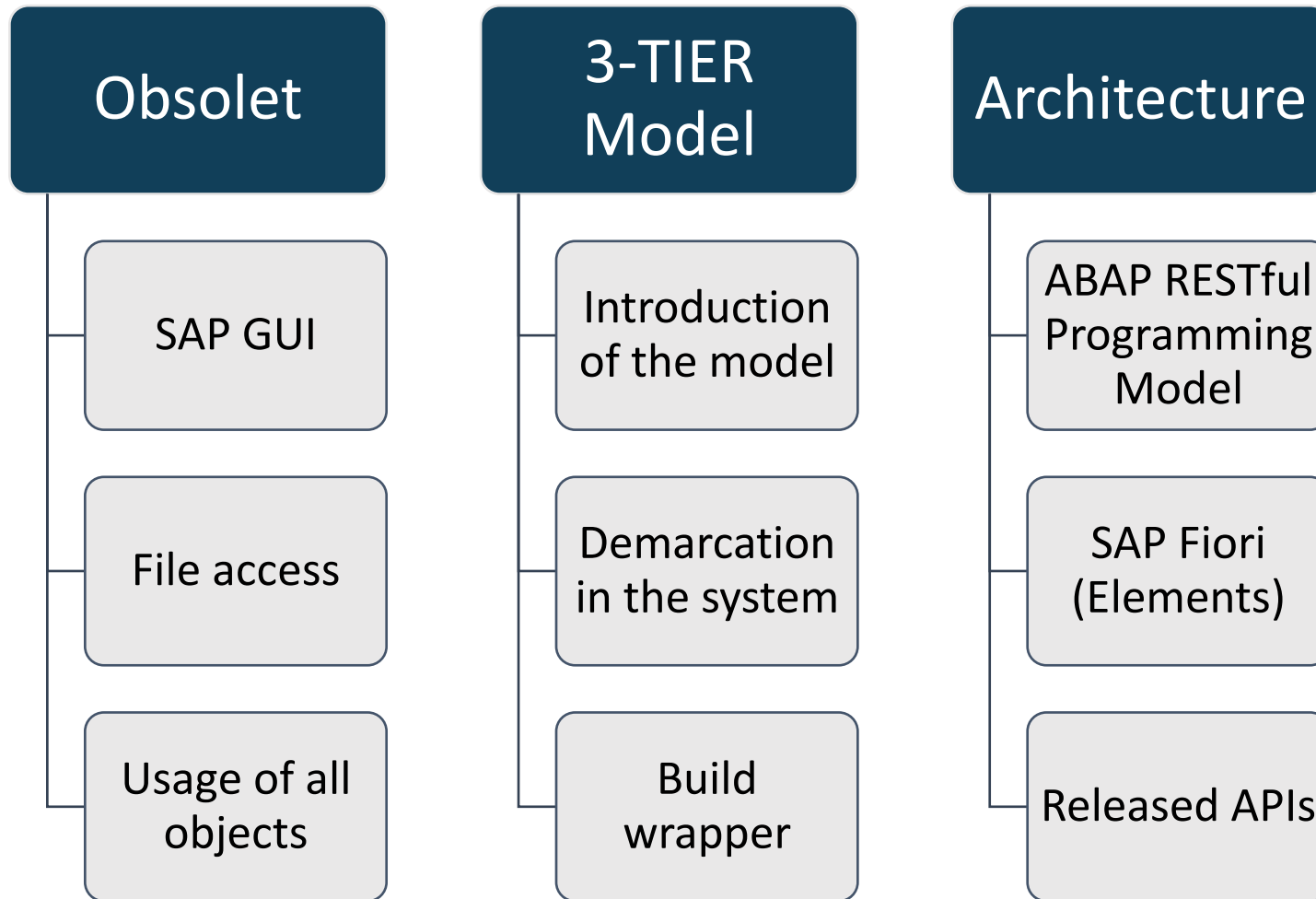
Foto von Arteum.ro auf Unsplash

ABAP Cloud

Development Model for all systems

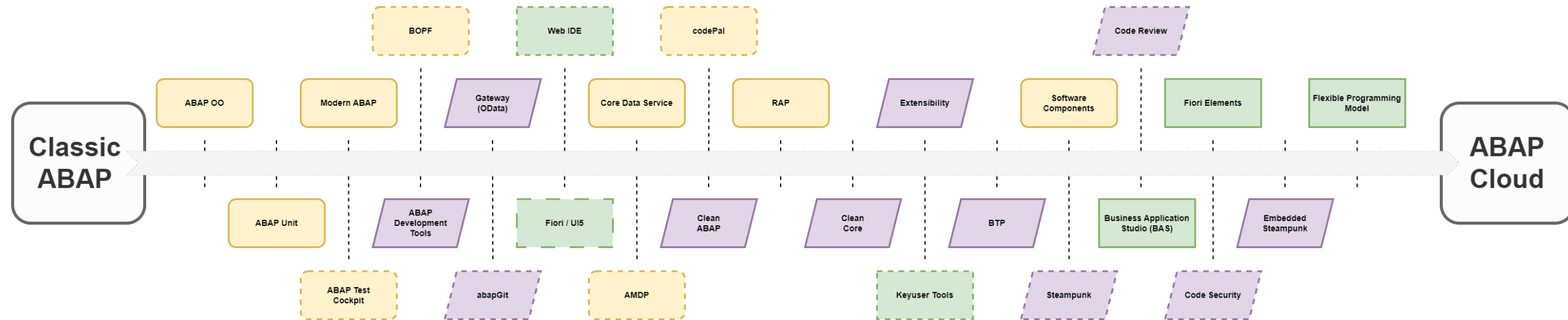


ABAP Cloud



ABAP Cloud

A journey starts ...



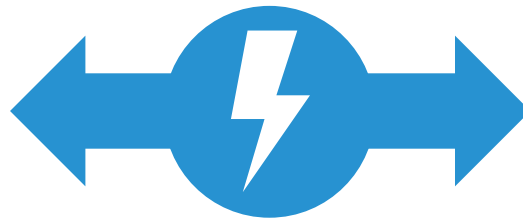
→ [Learning Roadmap](#)

ABAP Cloud



Fear

- Development style must be changed
- Not everything in the system can be used anymore
- Release completely



Chance

- Make ABAP lean again
 - Statements
 - APIs
 - Technology
- Easy entry



Foto von Nick Fewings auf Unsplash

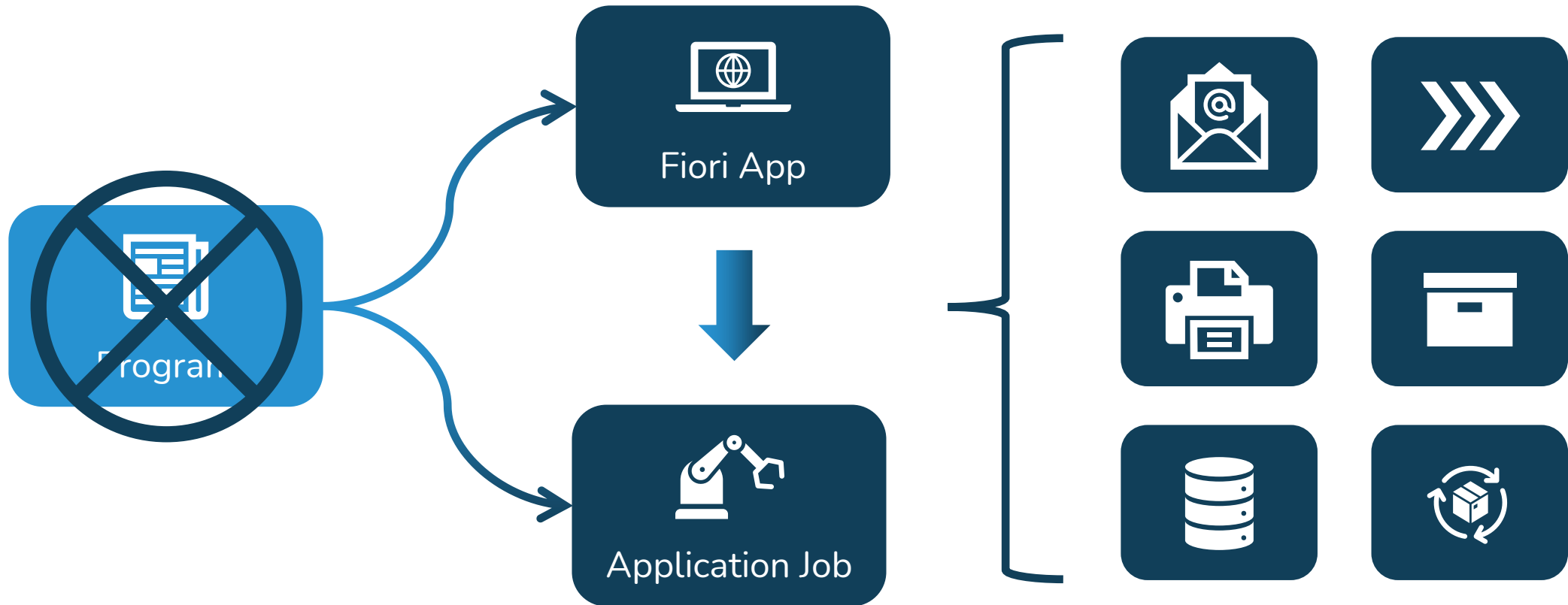
Possibilities

Possibilities

Classic ABAP

ABAP Cloud

Released APIs



Possibilities



Fiori App

- ABAP RESTful Programming Model (RAP) with Fiori (Elements)
- Various patterns for integration possible
 - Classic
 - Report
 - Custom
 - Hybrid
 - API
 - Singleton



Application Job

- Development of class-based processing
- Output of messages as application log
- Scheduling via job app or API



Foto von Nick Fewings auf Unsplash

Hands-On

Hands-On

The Scenario

- Migrating an ALV application to Fiori Elements
 - Evaluation report of documents
 - Enrichment of additional information
- Implementation of the "Report Pattern"

The screenshot displays the SAP Fiori 'Comment Bookings' application. The main table lists bookings with columns: BuKr., Gesellschaftsname, Belegnummer, Ges..., Pos, Belegdatum, Belegkopftext, Bsc, and Text. The table contains several rows, including two highlighted rows with Belegnummer 4900000018 and 4900000022. A modal window titled 'S4H(1)/100 Comment Bookings' is open, showing a 'Kommentar:' field and a 'Prüfen' button. The modal also displays a small table of bookings.

BuKr.	Gesellschaftsname	Belegnummer	Ges...	Pos	Belegdatum	Belegkopftext	Bsc	Text
1710	Company Code 1710	4900000018	2020	1	16.01.2020		99	
1710	Company Code 1710	4900000018	2020	2	16.01.2020		81	
1710	Company Code 1710	4900000022	2020	2	16.01.2020		81	
1710	Company Code 1710	4900000023	2020	1	16.01.2020		99	
1710	Company Code 1710	4900000023	2020	2	16.01.2020		81	
1710	Company Code 1710	4900000024	2020	1	16.01.2020		99	

Hands-On



Hands-On (Backup)

- Step: Migration of the table

[S4H,100] ZFI_COMMENTS	
General	
API State	
Transport	
Package:	ZFI_CEXT COMMENTS -> ZFI_CEXT
Version:	Active
Description:	Comments for FI documents
Responsible:	(Björn Schulz)
Created on:	Friday, November 29, 2024
Last Changed by:	(Björn Schulz)
Last Changed on:	Friday, November 29, 2024, 3:30:57 PM
Original Language:	DE
Original System:	S4H
Application Component:	-
ABAP Language Version:	ABAP für Cloud-Entwicklung

Hands-On (Backup)

- Step: Create Base-View

```
D [S4H,100] ZFI_B_BKPCOMMENTS X
1 @AccessControl.authorizationCheck: #NOT_REQUIRED
2 @EndUserText.label: 'Comments'
3 define view entity ZFI_B_BKPComments
4   as select from zfi_comments
5   {
6     key buhrs      as CompanyCode,
7     key belnr      as AccountingDocument,
8     key gjahr      as FiscalYear,
9     key buzei      as AccountingDocumentItem,
10    commt          as DocumentComment,
11    create_user    as CreationUser,
12    create_time    as CreationTime
13  }
```

Hands-On (Backup)

- Step: Create ROOT of the app

```
D [S4H,100] ZFI_R_BKPDOCUMENTCOMMENTS X
1 @AccessControl.authorizationCheck: #NOT_REQUIRED
2 @EndUserText.label: 'Comments for FI Documents'
3 define root view entity ZFI_R_BKPDocumentComments
4   as select from I_OperationalAcctgDocItem
5   association [0..1] to ZFI_B_BKPComments as _Comment on _Comment.CompanyCode           = $projection.CompanyCode
6                                                         and _Comment.AccountingDocument       = $projection.AccountingDocument
7                                                         and _Comment.FiscalYear           = $projection.FiscalYear
8                                                         and _Comment.AccountingDocumentItem = $projection.AccountingDocumentItem
9   {
10    key CompanyCode,
11    key AccountingDocument,
12    key FiscalYear,
13    key AccountingDocumentItem,
14    _CompanyCode.CompanyCodeName,
15    _JournalEntry.DocumentDate,
16    _JournalEntry.AccountingDocumentHeaderText,
17    PostingKey,
18    DocumentItemText,
19    CostCenter,
20    _Comment.DocumentComment
21  }
```


Hands-On (Backup)

- Step: Create service

```
[S4H,100] ZFI_BKP_COMMENTS X  
1 @EndUserText.label: 'Comments Service'  
2 define service ZFI_BKP_COMMENTS {  
3   expose ZFI_R_BKPDokumentComments as DocumentComments;  
4 }
```

[S4H,100] ZFI_UI_BKP_COMMENTS_O4 X

Service Binding: ZFI_UI_BKP_COMMENTS_O4

General Information
This section describes general information about this service binding
Binding Type: OData V4 - UI

Services
Define services associated with the Service Binding
Local Service Endpoint: Unpublished Publish

type filter text

Service Name	Version	Service Definition	
▼ ZFI_BKP_COMMENT			Add Service...
	1.0.0	ZFI_BKP_COMMENTS	Remove

Service Version Details
View information on selected service version

Service Information
Local service endpoint is not published. [Publish local service endpoint](#) to view service URL, entity sets and associations.

- Step: Release service

Filter

Systemalias:

LOCAL

ServicegruppenID:

ZFI_UI*

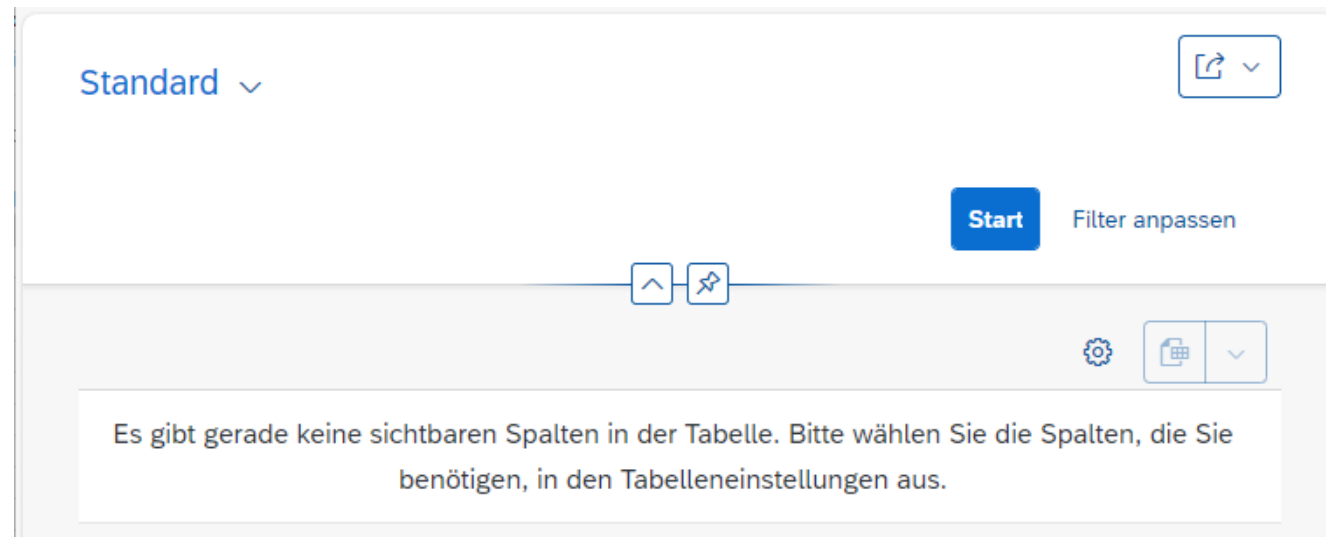
Nicht publizierte Servicegruppen

Servicegruppen publizieren

<input type="checkbox"/> Gruppen-ID	Repository-I...	Service-ID	Vers...	Beschreibung
<input type="checkbox"/> ZFI_UI_BKP_COMMENTS_O4				0

Hands-On (Backup)

- Interim result : Service callable



Hands-On (Backup)

- Step: Create projection view

```
[S4H,100] ZFI_C_BKPDOCUMENTCOMMENTS ×
1 @AccessControl.authorizationCheck: #NOT_REQUIRED
2 @EndUserText.label: 'Consumption View'
3 @Metadata.allowExtensions: true
4 define root view entity ZFI_C_BKPDocumentComments
5   provider contract transactional_query
6   as projection on ZFI_R_BKPDocumentComments
7   {
8     key CompanyCode,
9     key AccountingDocument,
10    key FiscalYear,
11    key AccountingDocumentItem,
12      CompanyCodeName,
13      DocumentDate,
14      AccountingDocumentHeaderText,
15      PostingKey,
16      DocumentItemText,
17      CostCenter,
18      DocumentComment
19  }
20
```

```
[S4H,100] ZFI_BKP_COMMENTS ×
1 @EndUserText.label: 'Comments Service'
2 define service ZFI_BKP_COMMENTS {
3   expose ZFI_C_BKPDocumentComments as DocumentComments;
4 }
```


Hands-On (Backup)



- Step: Metadata


```
[S4H,100] ZFI_C_BKPDOCUMENTCOMMENTS X
1 @Metadata.layer: #CUSTOMER
2 annotate entity ZFI_C_BKPDocumentComments
3 with
4 {
5
6 @UI : {
7   selectionField: [{ position: 10 }],
8   lineItem : [
9     { position: 10 },
10    { type: #FOR_ACTION, dataAction: 'setComment',label: 'Set comment' } ]
11 }
12 CompanyCode;
13
14 @UI : {
15   selectionField: [{ position: 20 }],
16   lineItem : [{ position: 30 }]
17 }
18 AccountingDocument;
19
20 @UI : {
21   selectionField: [{ position: 30 }],
22   lineItem : [{ position: 40 }]
23 }
24 FiscalYear;
25
26 @UI : {
27   lineItem : [{ position: 50 }]
28 }
29 AccountingDocumentItem;
30
31 @UI : {
32   lineItem : [{ position: 20 }]
33 }
34 CompanyCodeName;
```


Hands-On (Backup)


- Interim result : UI available



Standard 






 

Buchungskreis: 

Buchungsbeleg: 

Geschäftsjahr:  **Start** [Filter anpassen](#)

[Set comment](#)     

Buchungsk...	Gesellschaftsname	Buchungsbeleg	Geschäftsj...	Buchungssi...	Buchungsbelegdatum	Belegkopftext	Buchungssc...	
1010	Company Code 1010	100000000	2016	1	05.01.2016	Integration-AP	40	>
1010	Company Code 1010	100000000	2016	2	05.01.2016	Integration-AP	50	>
1010	Company Code 1010	100000000	2017	1	15.11.2017		40	>
1010	Company Code 1010	100000000	2017	2	15.11.2017		50	>
1010	Company Code 1010	100000000	2018	1	21.11.2018	Integration-AP	40	>
1010	Company Code 1010	100000000	2018	2	21.11.2018	Integration-AP	50	>

Hands-On (Backup)

- Step: Entity for Popup

```
D [S4H,100] ZFI_S_BKPCOMMENTPOPUP X
1 @EndUserText.label: 'Popup for Comment'
2 define abstract entity ZFI_S_BKPCommentPopUp
3 {
4     @EndUserText.label: 'Comment'
5     DocumentComment : abap.string;
6 }
7
```

Hands-On (Backup)

- Step: Behavior (Root)

```
[B] [S4H,100] ZFI_R_BKPDOCUMENTCOMMENTS X
1 managed implementation in class zbp-fi_bkp_document unique;
2 strict ( 2 );
3
4 define behavior for ZFI_R_BKPDDocumentComments alias Comment
5 with unmanaged save
6 lock master
7 authorization master ( instance )
8 {
9     update;
10
11     action setComment parameter ZFI_S_BKPCommentPopUp result [1] $self;
12 }
```


Hands-On (Backup)

- Step: Behavior implementation

```
METHOD setComment.  
  READ ENTITIES OF ZFI_R_BKPDDocumentComments IN LOCAL MODE  
    ENTITY Comment ALL FIELDS WITH CORRESPONDING #( keys )  
    RESULT DATA(lt_data).  
  
  LOOP AT lt_data INTO DATA(ls_data).  
    DATA(ls_key) = keys[ %tky = ls_data-%tky ].  
  
    MODIFY ENTITIES OF ZFI_R_BKPDDocumentComments IN LOCAL MODE  
      ENTITY Comment UPDATE FROM VALUE #( ( %tky  
        DocumentComment = ls_data-%tky  
        DocumentComment = ls_key-%param-DocumentComment  
        %control-DocumentComment = if_abap_behv=>mk-on ) ).  
  
  ENLOOP.  
  
  READ ENTITIES OF ZFI_R_BKPDDocumentComments IN LOCAL MODE  
    ENTITY Comment ALL FIELDS WITH CORRESPONDING #( keys )  
    RESULT DATA(lt_update).  
  
  result = VALUE #( FOR ls_update IN lt_update  
    ( %tky = ls_update-%tky %param = CORRESPONDING #( ls_update ) ) ).  
ENDMETHOD.
```

```
METHOD save_modified.  
  LOOP AT update-comment INTO DATA(ls_comment).  
    MODIFY zfi_comments FROM @( VALUE #( bukrs = ls_comment-CompanyCode  
      belnr = ls_comment-AccountingDocument  
      gjahr = ls_comment-FiscalYear  
      buzei = ls_comment-AccountingDocumentItem  
      commt = ls_comment-DocumentComment  
      create_user = cl_abap_context_info=>get_user_technical_name( )  
      create_time = utclong_current( ) ) ).  
  
  ENLOOP.  
ENDMETHOD.
```

```
@UI      : {  
  selectionField: [{ position: 10 }],  
  lineItem      : [  
    { position: 10 },  
    { type: #FOR_ACTION, dataAction: 'setComment', label: 'Set comment' }  
  ]  
}  
CompanyCode;
```

Hands-On (Backup)

- Step: Behavior (Projektion)

```
[B] [S4H,100] ZFI_C_BKPDOCUMENTCOMMENTS ×  
1 projection;  
2 strict ( 2 );  
3  
4 define behavior for ZFI_C_BKPDocumentComments alias Comments  
5 {  
6   use action setComment;  
7 }
```

Hands-On (Backup)

- Step: Access Control (Root)

```
[S4H,100] ZFI_R_BKPDOCUMENTCOMMENTS x
1 @EndUserText.label: 'ACL for Root'
2 @MappingRole: true
3 define role ZFI_R_BKPDocumentComments {
4     grant
5     select
6     on
7     ZFI_R_BKPDocumentComments
8     where
9     ( CompanyCode ) = aspect pfcg_auth ( F_BKPF_BUK, BUKRS , ACTVT = '03' );
10
11 }
```

Hands-On (Backup)

- Step: Access Control (Consumption)

```
[S4H,100] ZFI_C_BKPDOCUMENTCOMMENTS X
1 @EndUserText.label: 'ACL for Consumption'
2 @MappingRole: true
3 define role ZFI_C_BKPDocumentComments {
4   grant
5     select
6       on
7         ZFI_C_BKPDocumentComments
8         where
9           inheriting conditions from entity ZFI_R_BKPDocumentComments;
10
11 }
```

Hands-On

The result

- Implementation of the Fiori application
- Implementation of the functionality

→ Clean Core / Cloud Ready

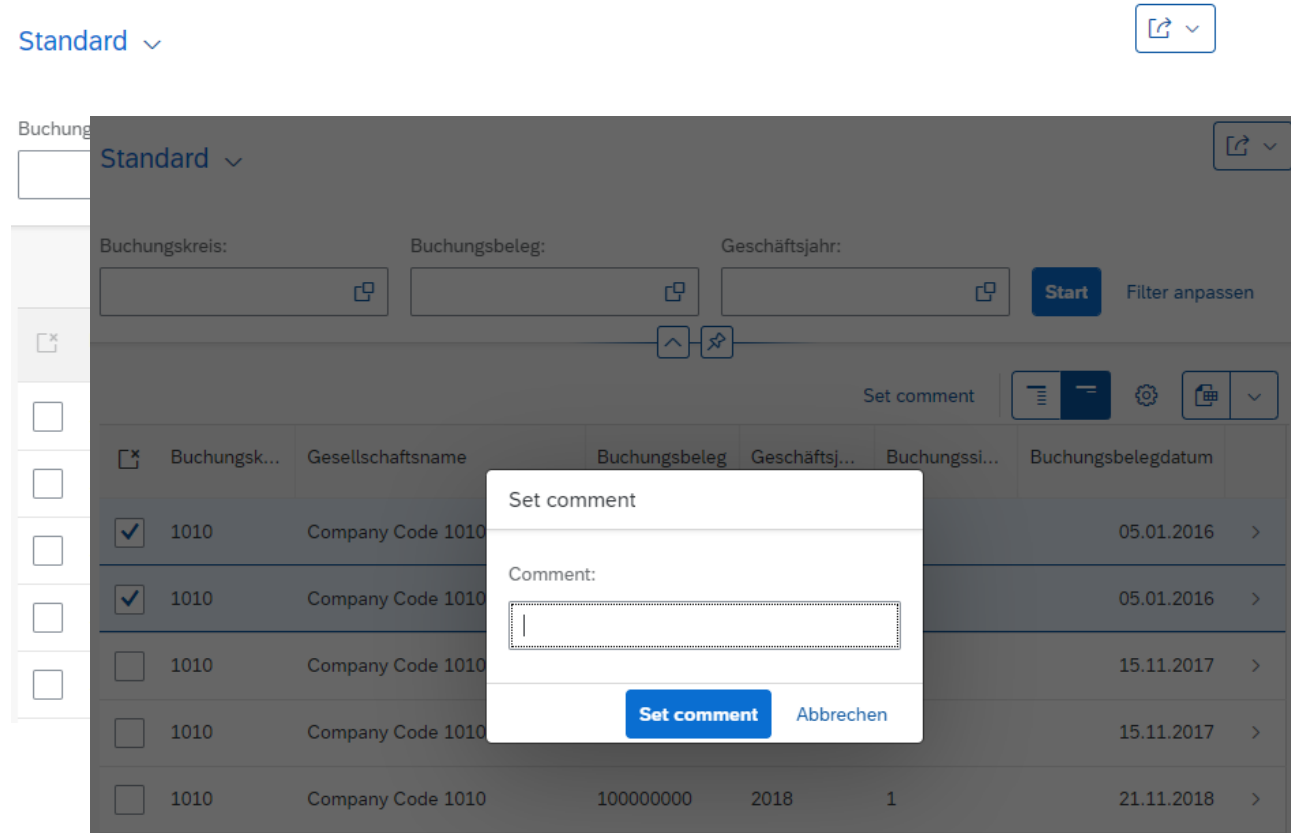




Foto von Joshua Hoehne auf Unsplash

Summary

Summary



Developer

- Developer training
- Understanding RAP
- Development tools



Strategy

- Breaking down complex applications
- Using the 3-TIER model



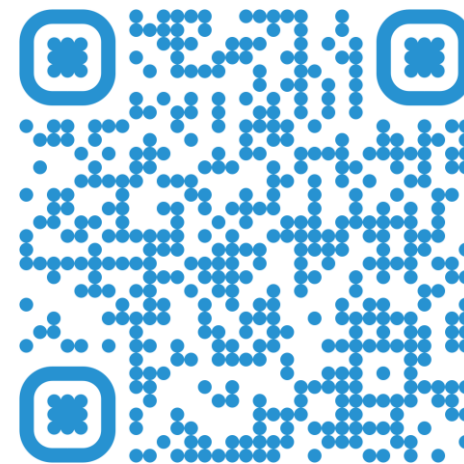
Migration

- Use common paths and patterns
- Fiori Elements for fast results

Thanks



Download



LinkedIn