

Willkommen / Welcome

December 5th 2024



```
DATA(ls_structure) = get_structure( 'ZABAPCONF' ).
ASSERT ls_structure IS NOT INITIAL.
get_sadl( IMPORTING es_sadl = DATA(ls_sadl) ).
DELETE ls_sadl-attributes USING KEY structure_id WHERE structure_id = ls_structure-id.

LOOP AT lt_property_binding ASSIGNING ls_property_binding WITH PROPERTY BOUNDING ID
  ls_attribute = VALUE #( |<
    name      = ls_property_binding-name
    binding   = <ls_property_binding-binding
    structure_id = ls_structure-id
  > ).
  IF <ls_property_binding>-is_key = abap_true.
    ls_attribute-is_key = if_sadl_type( 'is_key' ).
  ENDIF.
  INSERT ls_attribute INTO TABLE ls_sadl-attributes.
ENDLOOP.
```



ABAPConf orga

08:45 – 09:00 CET

Channel 1

German & English

5. Dezember 2024

abap
conf2024

WILLKOMMEN

```
DATA(ls_struct) = ...  
ASSERT ls_struct IS NOT INITIAL.  
get_sadl( IMPORTING es_sadl = DATA(es_sadl) )  
DELETE ls_sadl-attributes USING KEY structure_id WHERE structure_id = ls_struct->id.  
  
LOOP AT lt_property_binding ASSIGNING <ls_property_binding> WITH PROPERTY BINDING ID lt_struct->id.  
  ls_attribute = VALUE #( |<ls_struct->name| |<ls_struct->id| |<ls_struct->id| )  
  name = <ls_struct->name  
  binding = <ls_struct->binding  
  structure_id = <ls_struct->id  
  
  IF <ls_struct->is_key = abap_true.  
    ls_attribute-is_key = if_sadl_type-ot_is_key.  
  ENDIF.  
  INSERT ls_attribute INTO TABLE ls_sadl-attributes.  
ENDLOOP.
```

ABAPConf

LIVE | Online | On-Site | Free

MAFINEX, Mannheim, Deutschland
Cadaxo, Wien, Österreich
Live Streams auf YouTube!

ABAPConf

3 Channels

2 Keynotes

2 Sprachen

21 Vorträge

34 Sprecher:in

Start: 8:30 CEST

WATCH LIVE & FOR FREE youtube.com/@abapconf

Amsterdam

Watch Party

Mannheim

Main event

Vienna

Main event



**BROADCAST
START TIME**

5. DECEMBER 2024

Thank you to our Sponsors!

December 5th 2024

abap
conf2024

```
DATA(ls_structure) = get_structure(
ASSERT ls_structure IS NOT INITIAL.
get_sadl( IMPORTING es_sadl = DATA
DELETE ls_sadl-attributes USING KEY

LOOP AT lt_property_binding ASSIGN
ls_attribute = VALUE #( |<
name
binding
structure

IF <ls_property_binding>-is_key =
ls_attribute-is_key = if_sadl_?
ENDIF.
INSERT ls_attribute INTO TABLE ls
ENDLOOP.
```



ESPRESSO
TUTORIALS



Brandeis
Consulting

S/4FIT

cadaxo

Supported by



ABAPConf

Streaming

YouTube

Chat

YouTube

Probleme

Chat, Twitter, E-Mail

```
DATA(ls_structure) = get_structured_ls_entity_get(ls_entity)
ASSERT ls_structure IS NOT INITIAL.
get_sadl( IMPORTING es_sadl = DATA(ls_sadl) )
DELETE ls_sadl-attributes USING KEY structure IF WHERE structure_id = ls_structure-id.

LOOP AT it_ls_attributes ASSIGNING ls_attribute WITH PROGRESS INDICATOR:
  name = <ls_attribute-name>
  binding = <ls_attribute-binding>
  structure_id = ls_structure-id.
  IF <ls_attribute-binding> IS KEY = 'sap' AND
    ls_attribute-is_key = if_sadl_type-not_is_key.
    ENDIF.
  INSERT ls_attribute INTO TABLE ls_sadl-attributes.
ENDLOOP.
```


ABAPConf

Manifesto

<https://abapconf.org/abapconf2024/#/manifesto>

```
DATA(ls_structure) = get_structured( ls_entity get ls )
ASSERT ls_structure IS NOT INITIAL.
get_sadl( IMPORTING es_sadl = DATA(ls_sadl) )
DELETE ls_sadl-attributes USING KEY structure IF WERE structure IS = ls_structure.

LOOP AT lt_property_binding ASSIGNING ls_property_binding WITH PROPERTY BINDING (NOT INITIAL) REVERSE ORDER
  ls_attribute = VALUE #( |<
    name      = <ls_property_binding-aaa_name>
    binding   = <ls_property_binding-aaa_binding>
    structure IS = ls_structure
  > ).

IF <ls_property_binding>-is_key = abap_true.
  ls_attribute-is_key = if_sadl_type( not ls_attribute-is_key ).
ENDIF.
INSERT ls_attribute INTO TABLE ls_sadl-attributes.
ENDLOOP.
```




Time			Channel 1	Channel 2	Channel 3
			Location Mannheim	Location Vienna	Location Mannheim
08:00 - 09:00			Registrierung Coffee & Breakfast		
08:00 - 09:00			60 Min		
08:45 - 09:00			15 Min		
			Willkommen / Welcome		
			ABAPConf Orga Team Language: DE / EN		
09:00 - 09:50			Keynote K 1 Stephan Heinberg University of Zurich Language: EN		
09:50 - 10:40			V 101 ABAP Cloud - A life after the report Björn Schulz Language: DE	V 201 Streamlining ABAP Code reviews with AI: ABAP GenAI Co-pilot on Microsoft Teams for S/4HANA on-premise Pankaj Yadav Language: EN	V 301 Key User and Developer Extensibility Options - Best practices Thomas Schneider / Kai Dehmann Language: EN
10:40 - 11:30			V 102 Release management mit gCTS Christopher Graw Language: DE	V 202 Making an OnPremise ABAP CI/CD with Github Actions Simone Gaffurini Language: EN	V 302 Implementing Extension Fields using Key user and Developer Extensibility-Do's and Don'ts Karsten Schaser / Daniel Wachs Language: EN
11:30 - 12:20			V 103 Quo vadis ABAP Devs? Sören Schlegel Language: DE	V 203 Lessons learned building a product on Clean Core ABAP on S/4HANA public cloud Robert Eijpe / Octavian Mihail Savu Language: EN	V 303 Enable OData Streams in RAP Pavan Kumar Reddy / Ankita Singh Language: EN
12:20 - 12:30			Meet the Speakers		
12:30 - 13:30			Mittagessen / Lunch		
13:30 - 14:20			V 104 Unit Test your Business Objects in RAP Martin Jonen Language: EN	V 204 Dokumentengenerierung mit dox42 in ABAP RAP Lisa Pulsinger / Johann Fößleitner Language: DE	V 304 Clean core extensibility with ABAP Cloud in SAP S/4HANA Cloud Private Edition Volker Drees / Fabian Fellhauer Language: EN
14:20 - 15:10			V 105 apm - From Vision to Reality Marc Bernard Language: EN	V 205 Verloren im CDS Dschungel? Dusan Sacha / Oliver Wahrstötter Language: DE	V 305 ABAP Test Cockpit (ATC) goes Cloud – ATC on SAP BTP Olga Dolinskaja / Thomas Fiedler Language: EN
15:10 - 16:00			V 106 Von SAPscript zu Adobe Forms - Freud und Leid in der Formularentwicklung Frank Engert / Michael Keller Language: DE	V 206 Secure custom SAP development - Typical security issues in ABAP code Ivan Mans Language: EN	V 306 Advanced New ABAP George Drakos Language: EN
16:00 - 16:10			Meet the Speakers		
16:10 - 16:30			Kaffeepause / Coffee break		
16:30 - 17:20			V 107 ABAP Hybrid - Magie mit Graph Datenmodellen Jörg Müller Language: DE	V 207 Programming Glitches and ABAP Hacks Enno Wulff Language: DE	V 307 Integration Troubleshooting in the SAP BTP, ABAP Environment Ingo Bräuninger Language: EN
17:20 - 18:10			Closing Keynote K 2 The Modern SAP Full-Stack Developer Speaker: Sören Schlegel Fiori Elements meets ABAP Cloud: Demos and Outlook Tool Bridge (BAS/ADT), Analytical List Page (ALP), Treeview Edit (RAP), AI Support Speakers: Christoph Gollmick, Volker Drees, Fabian Fellhauer Language: EN		
18:10 - 18:20			Meet the Speakers		
18:20 - 18:25			This was the last ABAPConf		

Deutsch

Englisch

Speaker: On-Site

Speaker: Remote

Speaker: SAP



Tobias Hofmann



Sören Schlegel



Johann Föbleitner



Domi Bigl



Stephan Heinberg



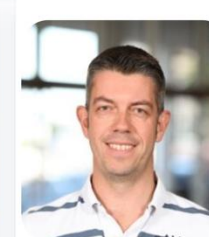
Björn Schulz



Christopher Graw



Fabian Fellhauer



Thomas Fiedler



Olga Dolinskaja



Martin Jonen



Marc Bernard



Frank Engert



Michael Keller



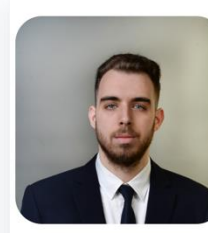
Jörg Müller



Pankaj Yadav



Simone Gaffurini



George Drakos



Ingo Bräuninger



Christoph Gollmick



Robert Eijpe



Octavian Savu



Regina Hochecker



Dusan Sacha



Oliver Wahrstötter



Ivan Mans



Enno Wulff



Thomas Schneider



Kai Dehmann



Karsten Schaser



Daniel Wachs



Pavan Reddy



Ankita Singh



Volker Drees

Channel 1

Channel 2

Channel 3

We are here

Keynote

Meet the speakers

Meet the speakers

Lunch

Lunch

Meet the speakers

Meet the speakers

Coffee break

Coffee break

Closing Keynote

Meet the speakers

ABAPConf 2025+

Development
S/4HANA

ABAP
Development

Beyond
Coding

```
DATA(ls_structure) = get_structured_data( 'S/4HANA' )
ASSERT ls_structure IS NOT INITIAL
get_sapl( IMPORTING es_sapl = ls_structure )
DELETE ls_sapl-attributes WHERE name = 'S/4HANA'

LOOP AT lt_property_binding WHERE name = 'S/4HANA'
  ls_attribute = VALUE #(
    name = 'S/4HANA'
    binding = 'S/4HANA'
    structure = 'S/4HANA'
  )
  IF <ls_property_binding> IS NOT INITIAL
    ENDIF.
  INSERT ls_attribute INTO TAB
ENDLOOP.
```


5. Dezember 2024

abap
conf2024

2025+

```
DATA(ls_struct) = ...  
ASSERT ls_struct IS NOT INITIAL.  
get_sadl( IMPORTING es_sadl = DATA(es_sadl) )  
DELETE ls_sadl-attributes USING KEY structure_id WHERE structure_id = ls_struct->id.  
  
LOOP AT lt_property_binding ASSIGNING <ls_property_binding> WHERE property_binding_id NOT INITIAL.  
  ls_attribute = VALUE #( |<ls_struct->name| |<ls_struct->id| |<ls_struct->id| )  
  name = <ls_struct->name  
  binding = <ls_struct->binding  
  structure_id = <ls_struct->id  
  
  IF <ls_property_binding>-is_key = abap_true.  
    ls_attribute-is_key = if_sadl_type-ot_is_key_true.  
  ENDIF.  
  INSERT ls_attribute INTO TABLE ls_sadl-attributes.  
ENDLOOP.
```

ABAPConf

Danke!

~100 Teilnehmer vor Ort
~1,500 Anmeldungen

```
DATA(ls_struct) = VALUE #( name = 'ABAPConf' structure = 'ABAPConf' ).  
ASSERT ls_struct IS NOT INITIAL.  
get_sadl( IMPORTING ls_struct = ls_struct ).  
DELETE ls_sadl-attributes USING KEY structure.  
  
LOOP AT lt_property_binding ASSIGNING ls_property_binding.  
  ls_attribute = VALUE #( name = ls_property_binding-name  
                           binding = ls_property_binding-binding  
                           structure_id = ls_struct-structure_id ).  
  
  IF <ls_property_binding>-is_key = 'yes'.  
    ls_attribute-is_key = 'yes'.  
  ENDIF.  
  
  INSERT ls_attribute INTO TABLE ls_sadl-attributes.  
ENDLOOP.
```

Das war ... Deutsch

```
DATA(is_structure) = get_structured(is_entity_get(is_
ASSERT is_structure IS NOT INITIAL.
get_sadl( IMPORTING es_sadl = DATA(is_sadl) )
DELETE is_sadl-attributes USING KEY structure IS WHERE structure_id = is_structure-id.

LOOP AT it_property_binding ASSIGNING is_property_binding WITH property_binding IS NOT INITIAL.
  is_attribute = VALUE #( is_
    name = <is_property_binding-name>
    binding = <is_property_binding-binding>
    structure_id = is_structure-id ).
  IF <is_property_binding>-is_key = abap_true.
    is_attribute-is_key = if_sadl_types-not_is_key_true.
  ENDIF.
  INSERT is_attribute INTO TABLE is_sadl-attributes.
ENDLOOP.
```