

4./5. Juni 2025

abap  
conf2025

WILLKOMMEN

```
DATA(ls_struct) = ...  
ASSERT ls_struct IS NOT INITIAL.  
get_sadl( IMPORTING eo_sadl = DATA(ls_sadl) ).  
DELETE ls_sadl-attributes USING KEY structure.  
  
LOOP AT it_property_binding ASSIGNING <ls_property_binding> WHERE property_binding IS NOT INITIAL.  
  ls_attribute = VALUE #( ls_ = ( ls_struct-attr = <ls_property_binding> ) )  
    name      = <ls_property_binding>-name  
    binding   = <ls_property_binding>-binding  
    structure = <ls_struct>-id .  
  
  IF <ls_property_binding>-is_key = abap_true.  
    ls_attribute-is_key = if_sadl_type-is_key_true.  
  ENDIF.  
  INSERT ls_attribute INTO TABLE ls_sadl-attributes.  
ENDLOOP.
```

# ABAPConf

LIVE | Online | On-Site | Free

Mannheim

Wien

Hamburg

Dresden

Deutschland

Österreich

Deutschland

Deutschland

```
DATA(ls_structure) = get_structure( 'ZABAP_CONF' ).  
ASSERT ls_structure IS NOT INITIAL.  
get_sadl( IMPORTING es_sadl = DATA(es_sadl) ).  
DELETE ls_sadl-attributes USING es_sadl-attributes.  
  
LOOP AT it_property_binding ASSIGNING ls_property_binding.  
  ls_attribute = VALUE #( ls_attribute = |ZABAP_CONF-  
    name = ls_property_binding-name  
    binding = ls_property_binding-binding  
    structure = ls_property_binding-structure ).  
  
  IF <ls_property_binding>-is_key = abap_true.  
    ls_attribute-is_key = if_sadl_type_is_key( ls_property_binding ).  
  ENDIF.  
  INSERT ls_attribute INTO TABLE ls_sadl-attributes.  
ENDLOOP.
```

# ABAPConf

4 Orte

7 Location Keynotes

2 Sprachen

21 Vorträge

2 Keynotes

4 Workshops

```
DATA(ls_structure) = get_structure( ls_entity_get ).  
ASSERT ls_structure IS NOT INITIAL.  
get_sadl( IMPORTING es_sadl = DATA(ls_sadl) ).  
DELETE ls_sadl-attributes USING KEY structure IS WHERE structure IS = ls_structure.  
  
LOOP AT lt_property_binding ASSIGNING ls_property_binding WHERE property_binding IS NOT INITIAL.  
  IF attribute IS NOT INITIAL.  
    IF ls_structure-key IS NOT INITIAL.  
      binding = <ls_property_binding-attribute IS = ls_structure-key>.  
      structure IS = ls_structure-attribute.  
    ENDIF.  
  ENDIF.  
  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.
```

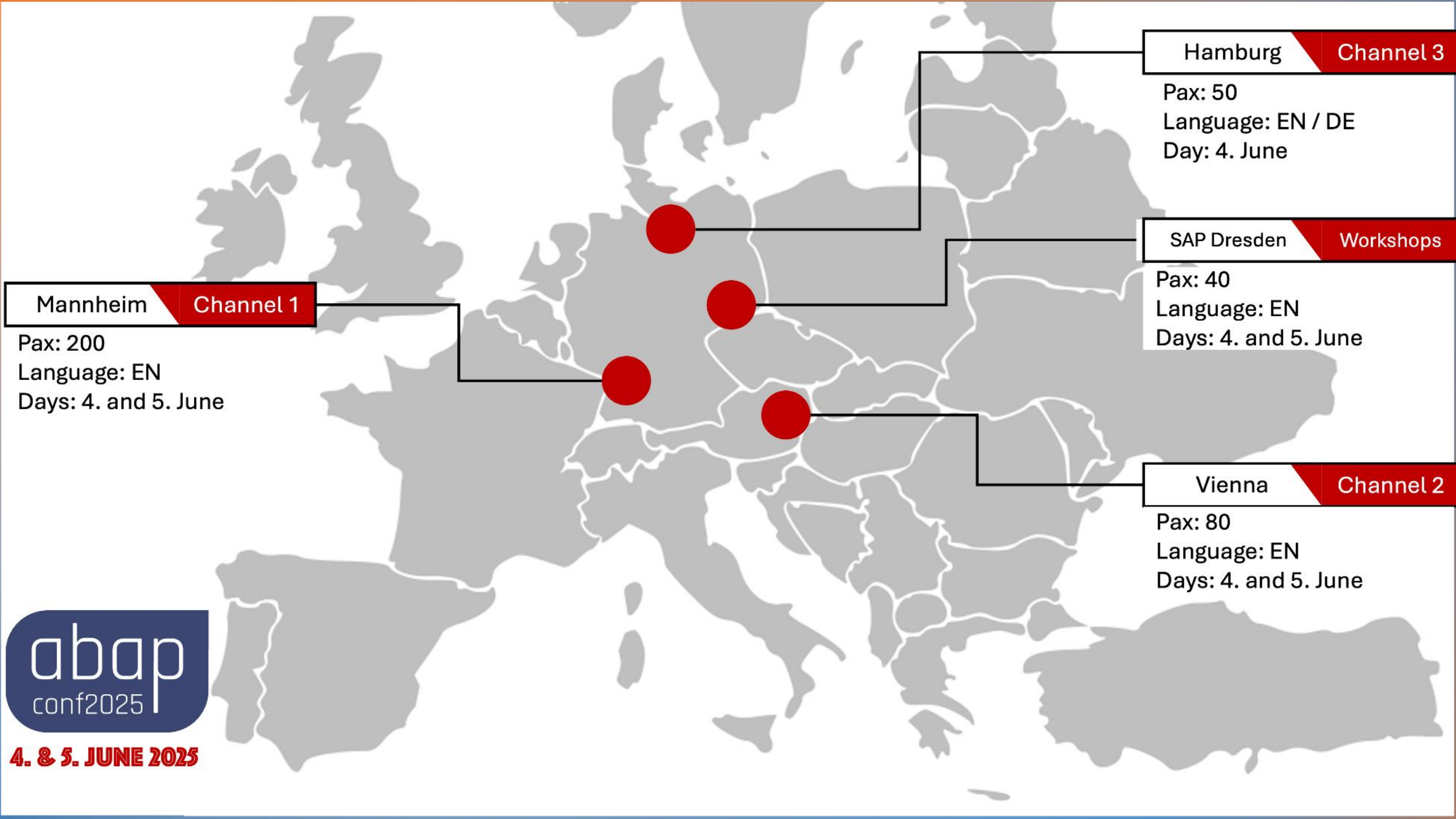
# ABAPConf

# 2 Tage

# +ABAPConf Community Day

# 7 Veranstaltungen ABAPConf

## 8 Veranstaltungen insgesamt



Mannheim Channel 1

Pax: 200  
Language: EN  
Days: 4. and 5. June

Hamburg Channel 3

Pax: 50  
Language: EN / DE  
Day: 4. June

SAP Dresden Workshops

Pax: 40  
Language: EN  
Days: 4. and 5. June

Vienna Channel 2

Pax: 80  
Language: EN  
Days: 4. and 5. June



4. & 5. JUNE 2025

Thank you to our Sponsors!



June 4./5. 2025



```
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 it_property_binding ASSIGN  
    ls_attribute = VALUE #( ( ls  
      name  
      binding  
      structure  
  
  IF <ls_property_binding>-is_key =  
    ls_attribute-is_key = if_sadl ?  
  ENDIF.  
  INSERT ls_attribute INTO TABLE ls  
  ENDLOOP.
```



	Day 1		
	Mannheim	Vienna	Hamburg
08:00	Registration		
09:00	Welcome		
09:30	Keynote		
10:30	Location Keynote	Location Keynote	Location Keynote
11:30	Ask the Experts	Ask the Experts	Ask the Experts
12:00	Lunch		Clean Core und ABAP
13:30	Optimized Tools for Building and Adapting SAP Fiori Elements Apps	tbd	Lunch
	Navigating the Clean Core Journey at SBB: Building Modern Fiori Elements Applications with RAP and ABAP Cloud Model for SAP S/4HANA On-Premise	CDS-only Data Modeling - Harnessing New ABAP CDS Object Types for Modern RAP Applications	Automatische SAP Paket Dokumentation mit AI
14:30	Coffee break		
15:30	Behind XCO and much further	Smart ABAP: Design Patterns for Real-World Scenarios	Hack dein ABAP: Grundsätze für code clean up
16:30	Müssen wir das tun, oder wollen wir das auch?	Closing	Closing
17:30	Closing		

	Day 2	
	Mannheim	Vienna
08:00	Registration	
09:00	Welcome	
09:30	Keynote	
10:30	Location Keynote	Location Keynote
11:30	Ask the Experts	Ask the Experts
12:00	Lunch	
13:30	Joule's ABAP developer capabilities in ABAP Developer Tools	The Magical World of SAP EWM for ABAPers
	Extending an API via BTP, ABAP environment	Effortless ABAP dependency management with TRM
14:30	Coffee break	
15:30	The last Overview you need on how to Extend S/4HANA On-Premise (as a developer)	RAP Events: Local and External Consumption
16:30	SAP Fiori Elements V4 & RAP: The Latest Innovations of V4 and the Evolution of the FPM Explorer	Closing
17:30	Closing	

Deutsch

Englisch

Mannheim

Wien

Hamburg

We are here

Keynote

Location Keynotes

Location Keynotes

Ask the Experts

Ask the Experts

Lunch

Lunch

Coffee break

Coffee break

Food for  
thoughts

Closing

Closing

Sessions

Sessions

Sessions

Sessions

Sessions

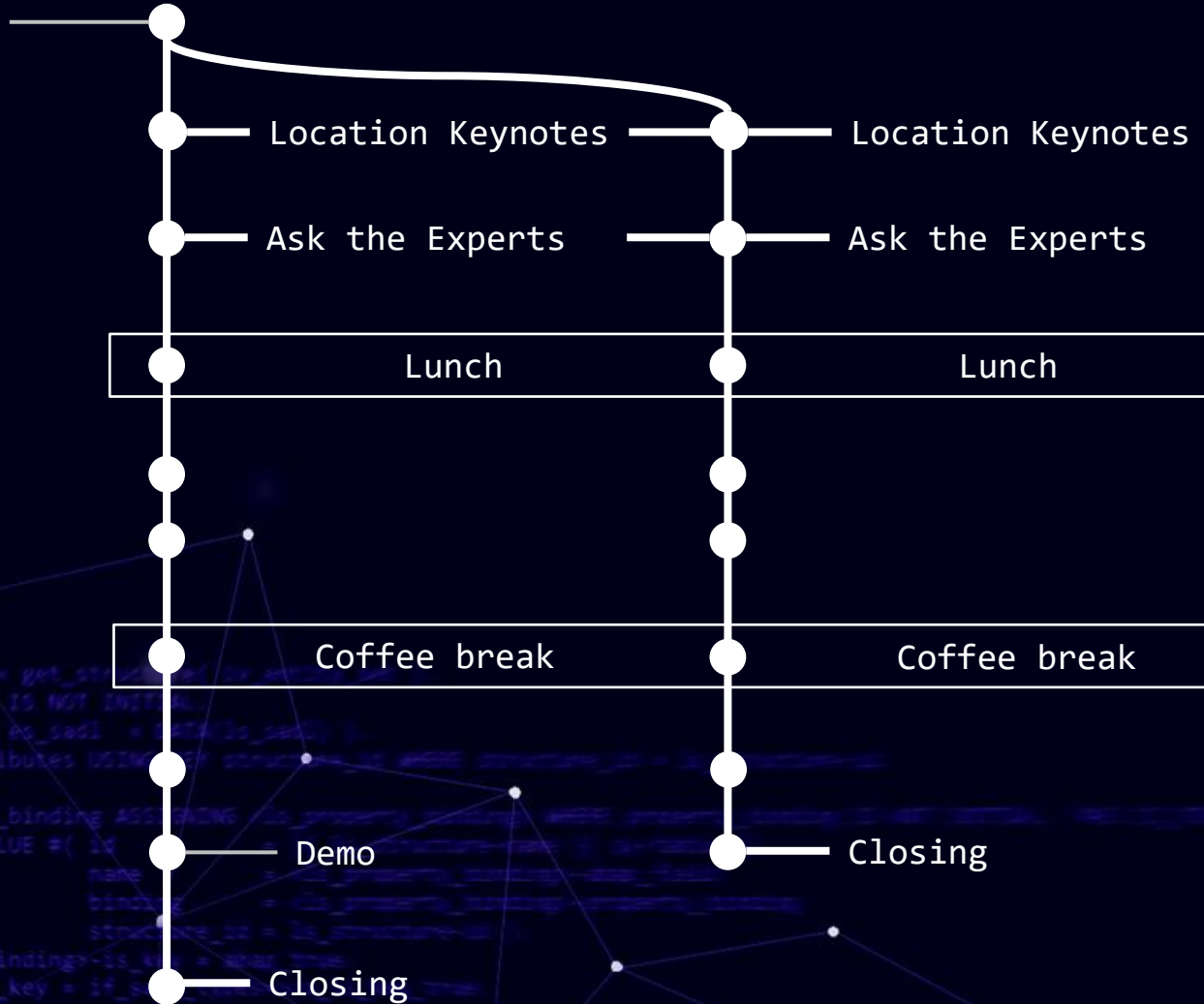
```
DATA(ls_structure) = get_structure(ls_structure)
ASSERT ls_structure IS NOT INITIAL.
get_sadl( IMPORTING es_sadl = DATA(ls_sadl) ).
ls_attribute = VALUE #( ls_attribute = VALUE #( ls_attribute = VALUE #(
  name = 'Food for thoughts'
  binding = 'Food for thoughts'
  structure = 'Food for thoughts'
) ) )
IF <ls_property_binding> IS NOT INITIAL.
  ls_attribute-is_key = if_<ls_property_binding>.
ENDIF.
INSERT ls_attribute INTO TABLE es_sadl-attributes.
ENDLOOP.
```



# Mannheim

# Wien

# Keynote



## Sessions

## Sessions

4./5. June 2025

abap  
conf2025

2025+

```
DATA(ls_struct) = ...  
ASSERT ls_struct IS NOT INITIAL.  
get_sadl( IMPORTING eo_sadl = DATA(ls_sadl) ).  
DELETE ls_sadl-attributes USING KEY structure.  
  
LOOP AT it_property_binding ASSIGNING <ls_property_binding> WHERE property_binding IS NOT INITIAL.  
  ls_attribute = VALUE #( ls_ = ( |<ls_structname>-<ls_property_binding>| ) )  
    name      = <ls_property_binding>-name  
    binding   = <ls_property_binding>-binding  
    structure = <ls_structname> .  
  
  IF <ls_property_binding>-is_key = abap_true.  
    ls_attribute-is_key = if_sadl_type-is_key_true.  
  ENDIF.  
  INSERT ls_attribute INTO TABLE ls_sadl-attributes.  
ENDLOOP.
```

# ABAPConf

Es wird keine Veranstaltung im Dezember geben

```
DATA(ls_structure) = get_structure( ls_entity_get ).  
ASSERT ls_structure IS NOT INITIAL.  
get_sadl( IMPORTING es_sadl = DATA(ls_sadl) ).  
DELETE ls_sadl-attributes USING KEY structure WHERE structure_id = ls_structure-id.  
  
LOOP AT it_property_binding ASSIGNING ls_property_binding WHERE property_binding IS NOT INITIAL.  
  ls_attribute = VALUE #( ls_ = (| ls_structure-name |) <data> )  
    name      = <ls_property_binding-name_field>  
    binding   = <ls_property_binding-structure_id>  
    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.
```

# Das war ... Deutsch

```
DATA(ls_structure) = get_structure( <ls_entity_get> ).  
ASSERT ls_structure IS NOT INITIAL.  
get_sadl( IMPORTING es_sadl = DATA(ls_sadl) ).  
DELETE ls_sadl-attributes USING KEY structure IS WHERE structure IS = ls_structure.  
  
LOOP AT it_property_binding ASSIGNING <ls_property_binding> WHERE property_binding IS NOT INITIAL. WHERE ( <ls_property_binding> )  
  ls_attribute = VALUE #( ls  
    = ( <ls_structure-name> ) <ls_entity_get> )  
    name  
      = <ls_property_binding-name>_field  
    binding  
      = <ls_property_binding-structure_binding>  
    structure IS = ls_structure-  
  ).  
  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.
```

4./5. Juni 2025

abap  
conf2025

WELCOME

```
DATA(ls_struct) = ...  
ASSERT ls_struct IS NOT INITIAL.  
get_sadl( IMPORTING eo_sadl = DATA(ls_sadl) ).  
DELETE ls_sadl-attributes USING KEY structure.  
  
LOOP AT it_property_binding ASSIGNING <ls_property_binding> WHERE property_binding IS NOT INITIAL.  
  ls_attribute = VALUE #( ls_ = ( ls_struct-attr = <ls_property_binding>-attr ) )  
    name      = <ls_property_binding>-attr_name  
    binding   = <ls_property_binding>-binding  
    structure = <ls_struct>-id .  
  
  IF <ls_property_binding>-is_key = abap_true.  
    ls_attribute-is_key = if_sadl_type-is_key_true.  
  ENDIF.  
  INSERT ls_attribute INTO TABLE ls_sadl-attributes.  
ENDLOOP.
```

# ABAPConf

LIVE | Online | On-Site | Free

Mannheim

Germany

Vienna

Austria

Hamburg

Germany

Dresden

Germany

```
DATA(ls_structure) = get_structur  
ASSERT ls_structure IS NOT NULL  
get_sadl( IMPORTING es_sadl = S  
DELETE ls_sadl-attributes USING KEY structur  
LOOP AT it_property_binding ASSI  
  ls_attribute = VALUE #( ls  
    name  
    binding  
    structur  
IF <ls_property_binding>-is_key  
  ls_attribute-is_key = if_sadl  
ENDIF.  
INSERT ls_attribute INTO TABS  
ENDLOOP.
```



# ABAPConf

4 Locations

7 Location Keynotes

2 Languages

21 Sessions

2 Keynotes

4 Workshops

```
DATA(ls_structure) = get_structure( ls_entity_get ).  
ASSERT ls_structure IS NOT INITIAL.  
get_sadl( IMPORTING es_sadl = DATA(ls_sadl) ).  
DELETE ls_sadl-attributes USING KEY structure IS WHERE structure IS = ls_structure.  
  
LOOP AT lt_property_binding ASSIGNING ls_property_binding WHERE property_binding IS NOT INITIAL.  
  IF attribute IS NOT INITIAL.  
    IF ls_structure-key IS NOT INITIAL.  
      binding = <ls_property_binding-attribute IS NOT INITIAL>.  
      structure IS = ls_structure-at .  
    ENDIF.  
  ENDIF.  
  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.
```

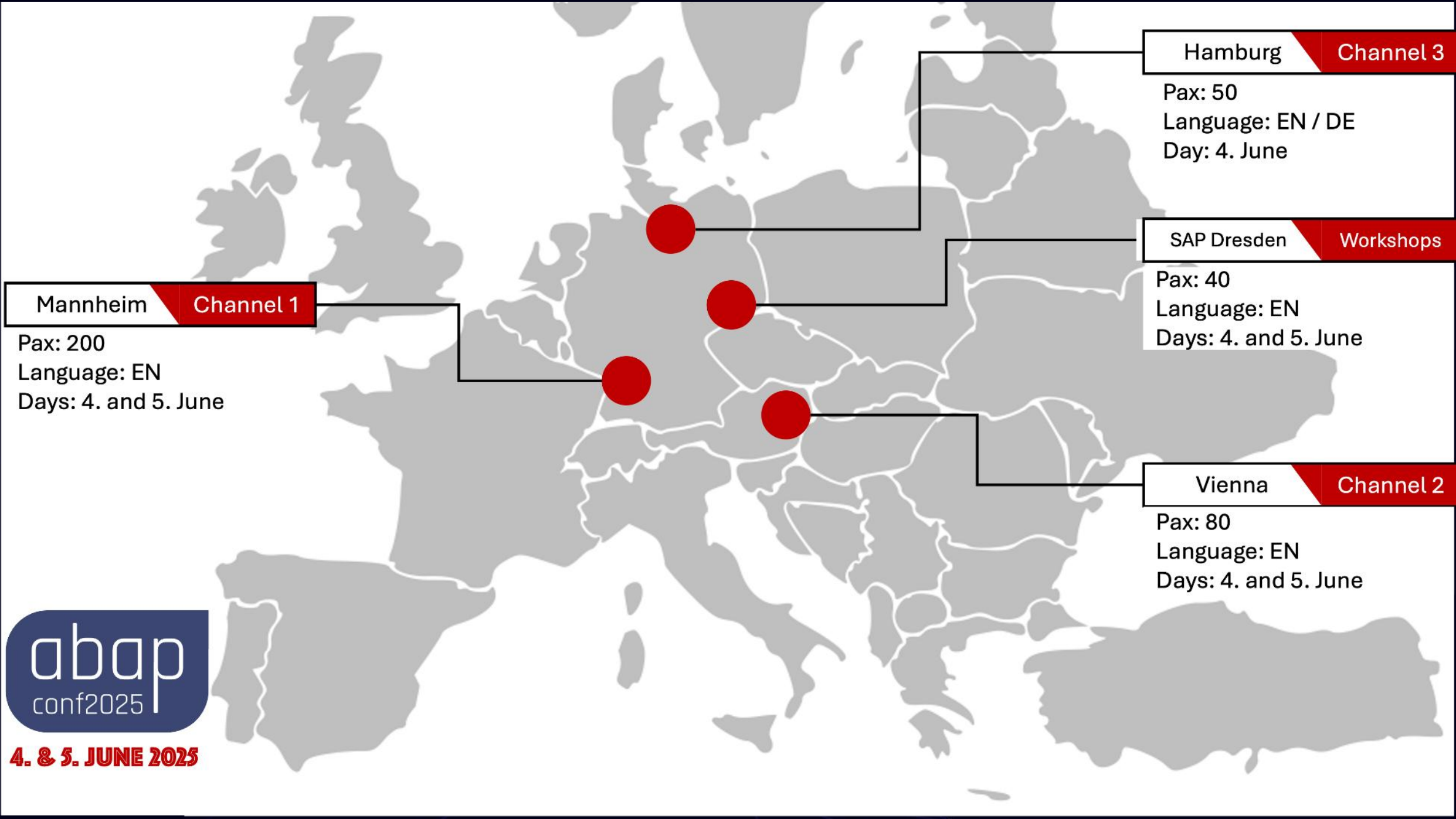
# ABAPConf

2 Days

+ABAPConf Community Day

7 Events ABAPConf  
8 Events in total

```
DATA(ls_structure) = get_structure( ls_entity_get ).  
ASSERT ls_structure IS NOT INITIAL.  
get_sadl( IMPORTING eo_sadl = DATA(ls_sadl) ).  
DELETE ls_sadl-attributes USING KEY struct.  
  
LOOP AT it_properties binding ASSIGN  
  ls_attribute = ME #( ls  
    name  
    binding  
    structure = ls_structure ).  
  
IF <ls_property_binding>-is_key  
  ls_attribute-key = if_sadl_key( ls_attribute-key ).  
ENDIF.  
INSERT ls_attribute INTO TABLE ls_sadl-attributes.  
ENDLOOP.
```



Mannheim Channel 1

Pax: 200  
Language: EN  
Days: 4. and 5. June

Hamburg Channel 3

Pax: 50  
Language: EN / DE  
Day: 4. June

SAP Dresden Workshops

Pax: 40  
Language: EN  
Days: 4. and 5. June

Vienna Channel 2

Pax: 80  
Language: EN  
Days: 4. and 5. June



4. & 5. JUNE 2025



## Day 1

**Mannheim**

Vienna

## Hamburg

## Day 2

**Mannheim**

## Vienna

Thank you to our Sponsors!



June 4./5. 2025

abap  
conf2025



```
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 it_property_binding ASSIGN  
ls_attribute = VALUE #( ( ls  
name  
binding  
structure  
  
IF <ls_property_binding>-is_key =  
ls_attribute-is_key = if_sadl ?  
ENDIF.  
INSERT ls_attribute INTO TABLE ls  
ENDLOOP.
```

	Day 1		
	Mannheim	Vienna	Hamburg
08:00	Registration		
09:00	Welcome		
09:30	Keynote		
10:30	Location Keynote	Location Keynote	Location Keynote
11:30	Ask the Experts	Ask the Experts	Ask the Experts
12:00	Lunch		Clean Core und ABAP
13:30	Optimized Tools for Building and Adapting SAP Fiori Elements Apps	tbd	Lunch
	Navigating the Clean Core Journey at SBB: Building Modern Fiori Elements Applications with RAP and ABAP Cloud Model for SAP S/4HANA On-Premise	CDS-only Data Modeling - Harnessing New ABAP CDS Object Types for Modern RAP Applications	Automatische SAP Paket Dokumentation mit AI
14:30	Coffee break		
15:30	Behind XCO and much further	Smart ABAP: Design Patterns for Real-World Scenarios	Hack dein ABAP: Grundsätze für code clean up
16:30	Müssen wir das tun, oder wollen wir das auch?	Closing	Closing
17:30	Closing		

	Day 2	
	Mannheim	Vienna
08:00	Registration	
09:00	Welcome	
09:30	Keynote	
10:30	Location Keynote	Location Keynote
11:30	Ask the Experts	Ask the Experts
12:00	Lunch	
13:30	Joule's ABAP developer capabilities in ABAP Developer Tools	The Magical World of SAP EWM for ABAPers
	Extending an API via BTP, ABAP environment	Effortless ABAP dependency management with TRM
14:30	Coffee break	
15:30	The last Overview you need on how to Extend S/4HANA On-Premise (as a developer)	RAP Events: Local and External Consumption
16:30	SAP Fiori Elements V4 & RAP: The Latest Innovations of V4 and the Evolution of the FPM Explorer	Closing
17:30	Closing	



German

English



# Speakers

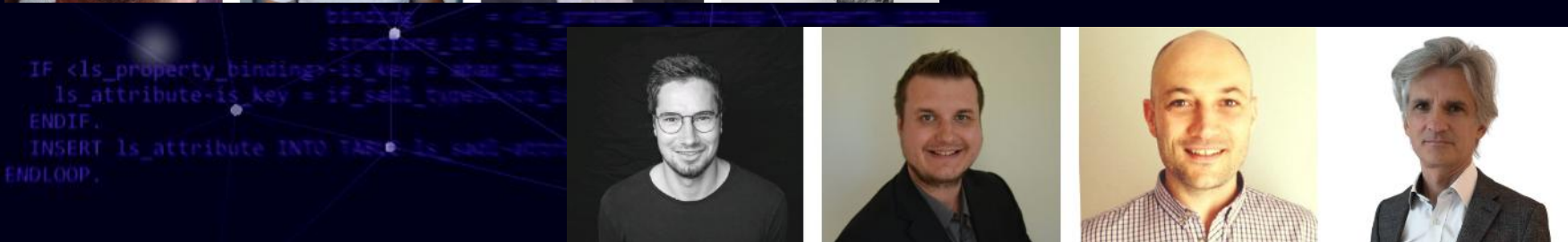
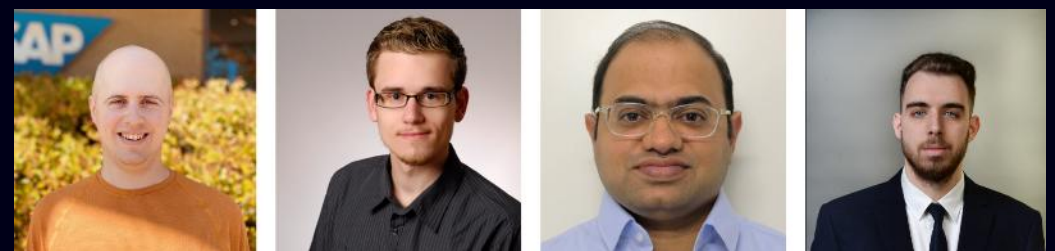
41 speakers

24 new speakers



41 experts

```
DATA(ls_structure) = ...  
ASSERT ls_structure IS NOT INITIAL.  
get_sadl( IMPORTING eo_sadl = DATA(ls_sadl) ).  
DELETE ls_sadl-attributes USING KEY structure WHERE structure IS = ls_structure.  
  
LOOP AT it_property_binding ASSIGNING ls_property_binding WHERE property_binding IS NOT INITIAL.  
  ls_attribute = VALUE #( ls  
    name      = <ls_property_binding-name> <ls_sadl-name>  
    binding   = <ls_property_binding-binding>  
    structure IS = ls_structure-at ).  
  
  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.
```



```
binding
structure is = is_
IF <ls_property_bindings>-is_key = when true
  is_attribute-is_key = if_said_type-is_key
ENDIF.
INSERT is_attribute INTO TABLE is_said_attr
ENDLOOP.
```

Mannheim

Vienna

Hamburg

We are here

Keynote

Location Keynotes

Location Keynotes

Ask the Experts

Ask the Experts

Lunch

Lunch

Coffee break

Coffee break

Food for thoughts

Closing

Closing

Sessions

Sessions

Sessions

Sessions

Sessions

```
DATA(ls_structure) = get_structure(ls_structure)
ASSERT ls_structure IS NOT NULL
get_sadl( IMPORTING es_sadl = DATA(ls_sadl) )
ls_attribute = VALUE #( ls_name = 'Closing' )
LOOP AT it_property_binding ASSIGNING ls_attribute
  IF <ls_property_binding> IS KEY
    ls_attribute-is_key = if_<ls_property_binding> IS KEY
  ENDIF.
  INSERT ls_attribute INTO TABLE ls_sadl-attributes
ENDLOOP.
```

Mannheim

Vienna

Keynote

Location Keynotes

Location Keynotes

Ask the Experts

Ask the Experts

Lunch

Lunch

Coffee break

Coffee break

Demo

Closing

Closing

Sessions

Sessions

```
DATA(ls_structure) = get_structure(ls_structure)
ASSERT ls_structure IS NOT NULL
get_sadl( IMPORTING es_sadl = DATA(ls_sadl) )
DELETE ls_sadl-attributes USING es_sadl
LOOP AT it_property_binding ASSIGNING ls_property_binding
  ls_attribute = VALUE #( ls_name = it_property_binding-name
    ls_binding = it_property_binding-binding
    ls_structure = ls_structure )
  IF <ls_property_binding>-is_key = true
    ls_attribute-is_key = if <ls_attribute>-is_key = true
      true
    else
      false
    endif
  INSERT ls_attribute INTO TABLE ls_sadl-attributes
ENDLOOP.
```



4./5. June 2025

abap  
conf2025

2025+

```
DATA(ls_struct) = ...  
ASSERT ls_struct IS NOT INITIAL.  
get_sadl( IMPORTING eo_sadl = DATA(ls_sadl) ).  
DELETE ls_sadl-attributes USING KEY structure.  
  
LOOP AT it_property_binding ASSIGNING <ls_property_binding> WHERE property_binding IS NOT INITIAL.  
  ls_attribute = VALUE #( ls_ = ( ls_struct-attr = <ls_property_binding> ) )  
    name      = <ls_property_binding>-name  
    binding   = <ls_property_binding>-binding  
    structure = <ls_struct>-id .  
  
  IF <ls_property_binding>-is_key = abap_true.  
    ls_attribute-is_key = if_sadl_type-is_key_true.  
  ENDIF.  
  INSERT ls_attribute INTO TABLE ls_sadl-attributes.  
ENDLOOP.
```

# ABAPConf

There won't be an event in December

```
DATA(ls_structure) = get_structure( ls_entity_get ).
ASSERT ls_structure IS NOT INITIAL.
get_sadl( IMPORTING eo_sadl = DATA(ls_sadl) ).
DELETE ls_sadl-attributes USING KEY structure WHERE structure_id = ls_structure-id.

LOOP AT it_property_binding ASSIGNING ls_property_binding WHERE property_binding IS NOT INITIAL.
  ls_attribute = VALUE #( ls_
    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-not_is_key_true.
  ENDIF.
  INSERT ls_attribute INTO TABLE ls_sadl-attributes.
ENDLOOP.
```



# ABAPConf

be a speaker

be a sponsor

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

LOOP AT it_property_binding ASSIGNING ls_property_binding WHERE property_binding IS NOT INITIAL.
  ls_attribute = VALUE #( ls = ( ls_structure-name ) derivative ( )
    name = <ls_property_binding-name>_field
    binding = <ls_property_binding-structure_binding>
    structure_id = ls_structure-id ).
  IF <ls_property_binding>-is_key = abap_true.
    ls_attribute-is_key = if_sadl_type-not_is_key_true.
  ENDIF.
  INSERT ls_attribute INTO TABLE ls_sadl-attributes.
ENDLOOP.
```