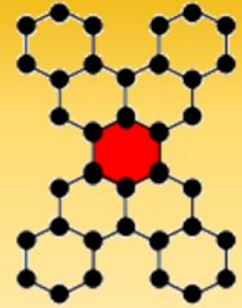


facet.at

by Dr. Roland Schwaiger

- Methoden, Muster und Konzepte
- SAP Development Schulungen
- SAP Development Fachbücher
- SAP Development Projekte
- Akademische Vorlesungen



UV Enterprise Computing WS 2021/2022

SW-Entwicklung mit SAP-ABAP

sapserver.cosy.sbg.ac.at/00/001
BCUSER
minisap
EN

Dr. Schwaiger Roland

Vorstellung



roland.schwaiger@facet.at

Dr. Roland Schwaiger

Located

Bad Dürrenberg, Hallein, AT

Background

Mathematics (University Salzburg)

Computer Sciences (University Salzburg, Bowling Green State University)

Project & Process Management (SMBS – University of Salzburg Business School)

Profession

SAP Technical Consultant (Cert. SAP Development Consultant)

SAP Trainer

Project Coach (Cert. Scrum Master)

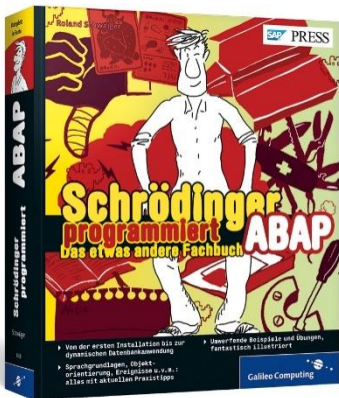
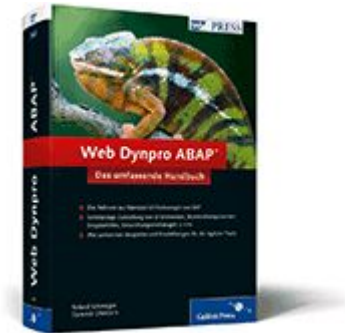
Software Architect

Software Developer (SAP AG, Walldorf, DE and Customer Development Projects)

Author (check out Amazon and/or www.citeseer.com)

Lector (Rheinwerk Verlag)

Lecturer (University Salzburg, FH Salzburg)



Begriffsklärung



“Enterprise computing involves the development, deployment and maintenance of the information systems required for survival and success in today’s business climate.”

Yen-Ping Shan & Ralph H. Earle,
Enterprise Computing with Objects,
Addison-Wesley, 1998.

Motivation Enterprise Computing

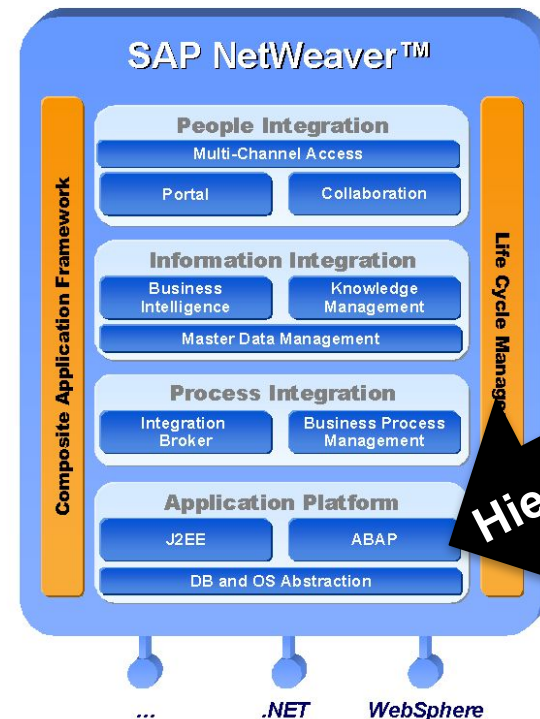
Custom Development in Unternehmen, die SAP einsetzen



Die zwei Aspekte, die wir betrachten wollen:

1. Das SAP System
2. Das ABAP

und das anhand ein Anwendung objektorientiert aufgebaut wird.



Hier sind wir

Quelle: [7]

Inhalte/Organisation



Infrastruktur Client-Server, Laufzeitumgebung, Arbeitsprozesse

ABAP Programme Ausführbares Programm, Module Pool

Werkzeuge Object Navigator, Class Builder, ABAP Editor, ABAP Dictionary, [Eclipse ADT, Web IDE, BAS, ...]

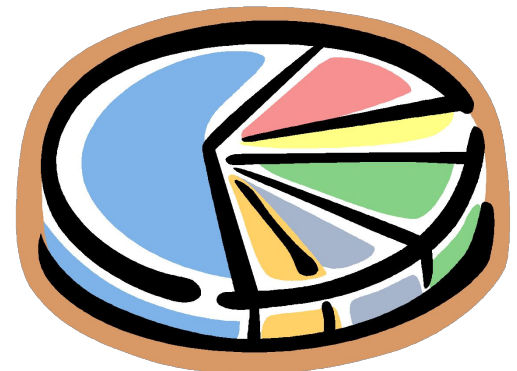
Datentypen und Datenobjekte DATA, TYPES

Modularisierungen Ereignisblock, Unterprogramm, Funktionsbaustein, [REST Services, ...]

ABAP OO Klasse definieren, Objekt instanzieren, Methode, Attribut, ...

GUIs Selektionsbild, Liste, Dynpro, [Web Dynpro, SAP UI5, ...]

DB Tabellen erstellen, auslesen



Abschluss LV



Der Abschluss der LV findet im Rahmen des LV Blocks statt. Dieser besteht aus:

- Implementierung von Repository Objekten im Rahmen der LV
- Abschlussprüfung
 - Theoretische Prüfung mit zehn Fragen aus den Inhalten der LV
 - Entwicklung von Repository Objekten

Einordnung SAP System



- The first version of SAP's **enterprise software** was a **financial Accounting system** named **RF** or better known as **R/1**. (The "R" was for "Real-time data processing")
- This was replaced by **R/2** at the end of the 1970s. **SAP R/2** was a **mainframe based** business application software suite that was very successful in the 1980s and early 1990s
- The client-server based solution **SAP R/3** was officially launched on 6 July 1992. (The 3 stands for the **three layer architecture**)
- In 2008 SAP HANA, an in-memory, column-oriented RDBMS was introduced.
- S/4HANA: S = Simplified, /4 kommt nach /3, HANA

Einordnung

Einordnung SAP



SAP History

- „SAP Systemanalyse und Programmentwicklung“ was founded in 1972 by 5 former IBM engineers
- In 1976 the SAP GmbH was founded, they moved to Walldorf
- In 1988 the SAP Aktiengesellschaft *Systeme, Anwendungen und Produkte in der Datenverarbeitung* was founded

Einordnung SAP History



- SAP is the **largest European** and the third largest software producer world-wide
- SAP has more than **99.700** employees(19.000 software development) (2019)
- Annual Revenue (IFRS) 2018 was **24.7bn** Euros
- SAP products are used by ~ **335.000** customers in 120 subsidiaries.
- SAP has about **12** dev centers worldwide

Motivation

SAP

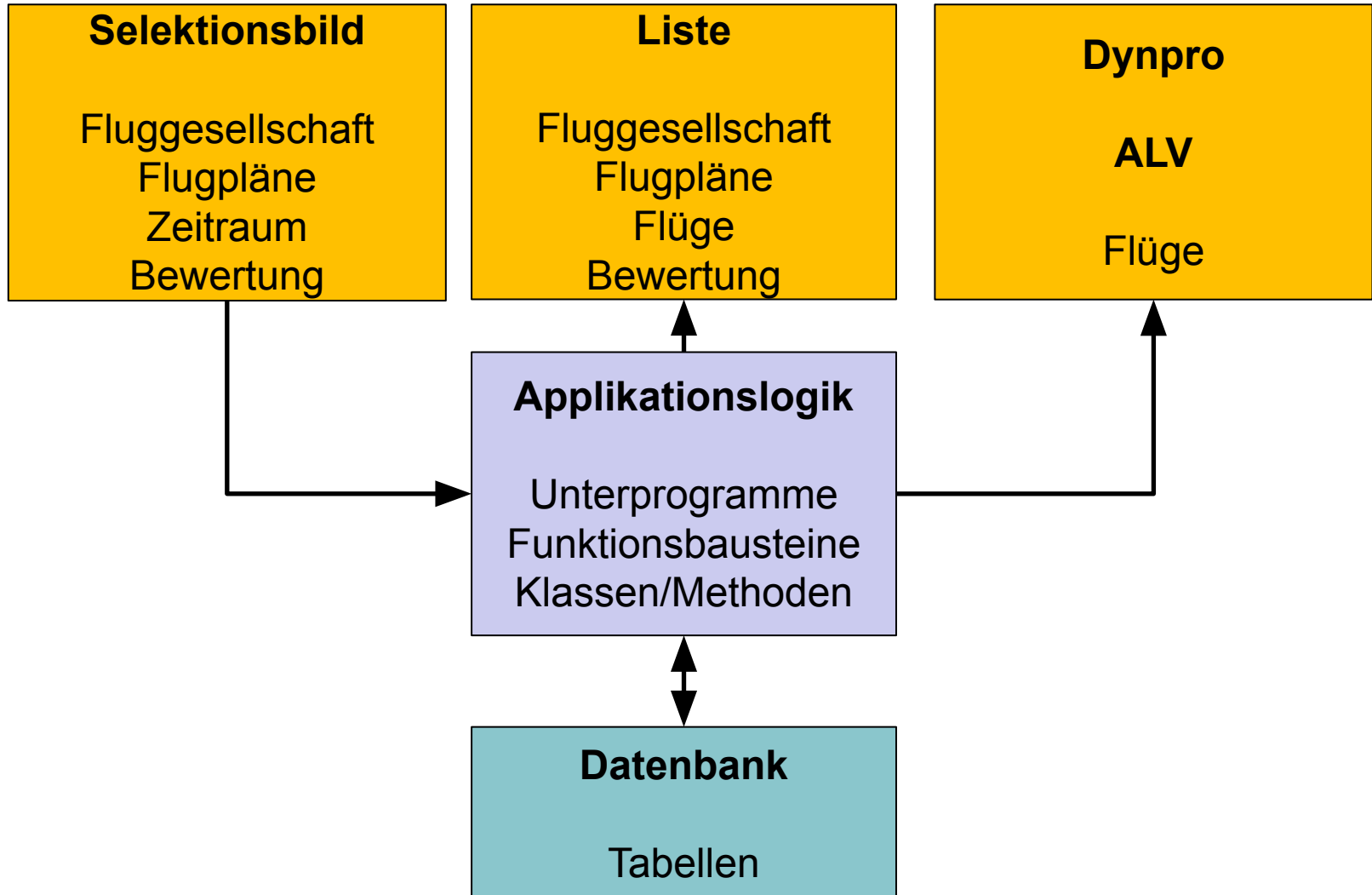
SAP is ...



- the corporation SAP
- the family of products of the SAP, referred to as **THE SAP SYSTEMS**

Unser Beispiel

Anwendung ToDev



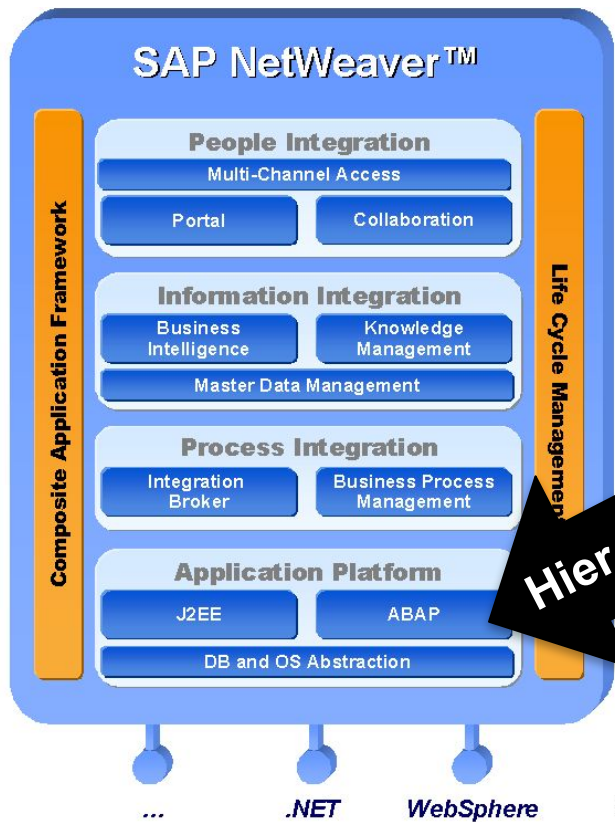
Begriffe und Handgriffe

Begriffe und Handgriffe

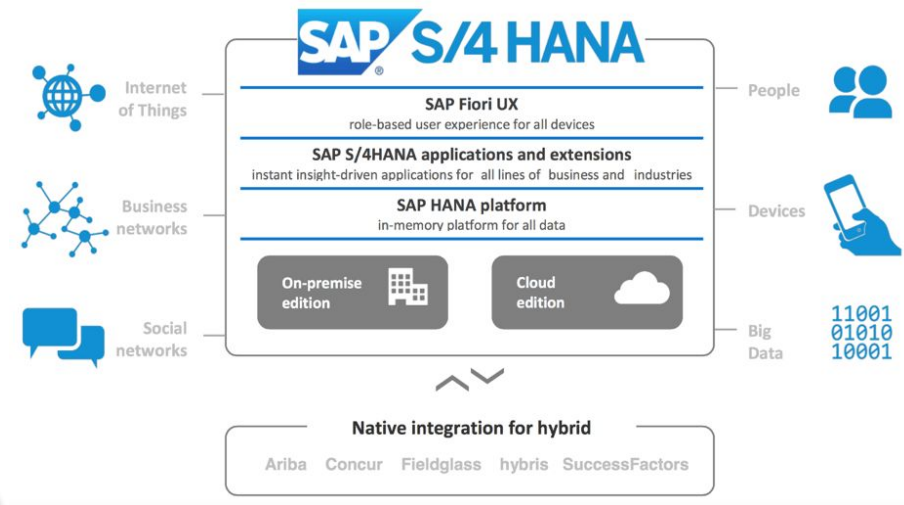


SAP NetWeaver

S/4HANA Cloud



Hier sind wir



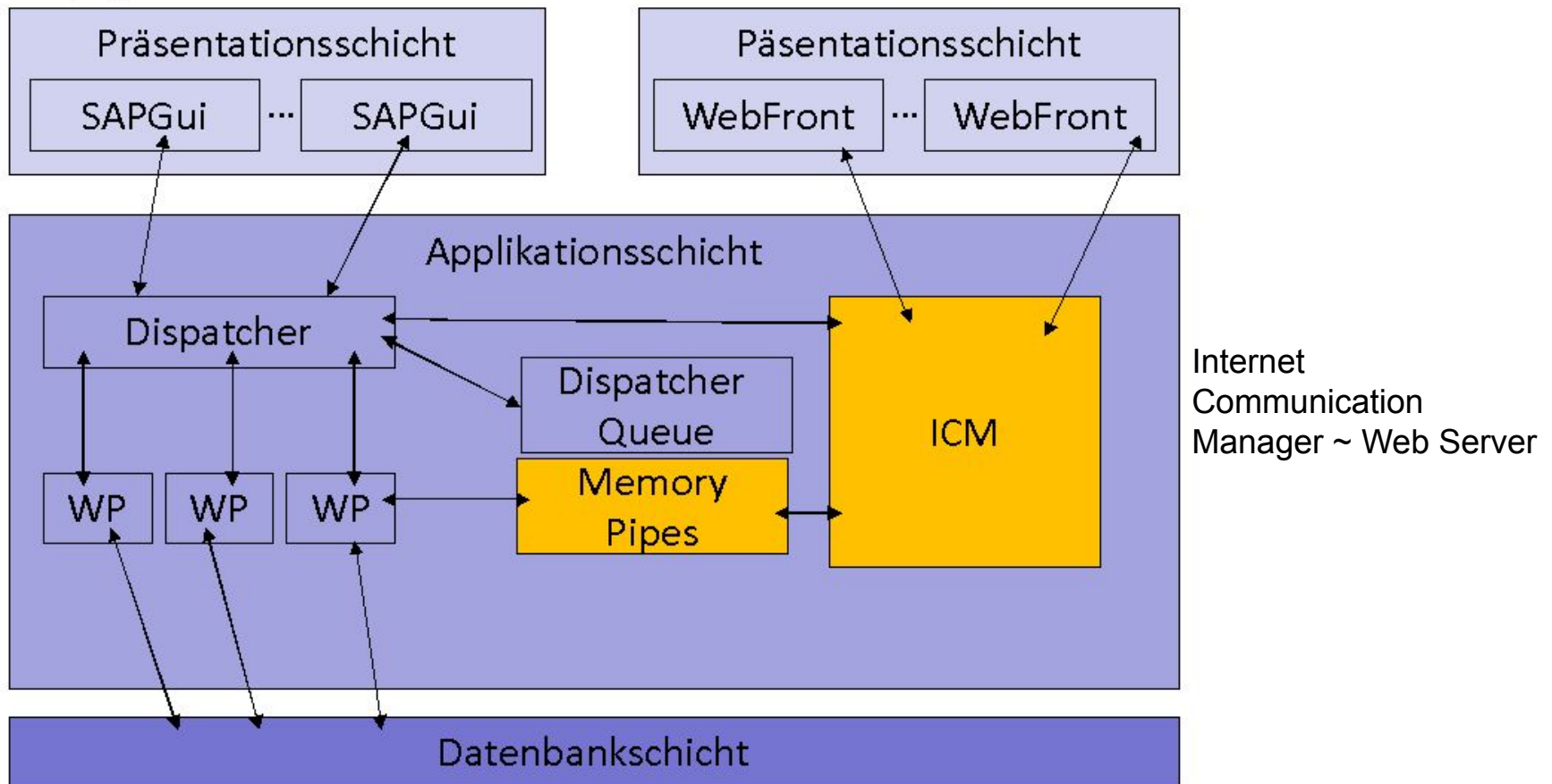
Quelle: [7]

© SAP

Begriffe und Handgriffe

Application Plattform ABAP

Schlagwort: "FIORI"

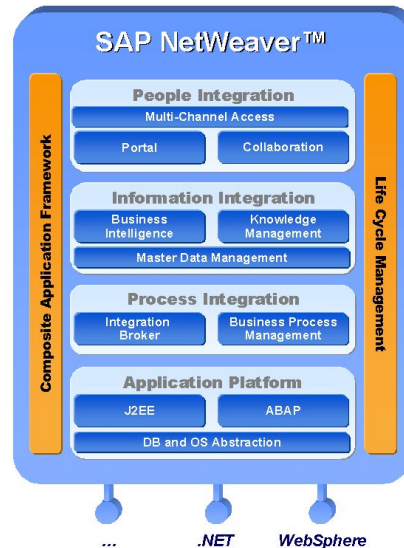


Begriffe und Handgriffe



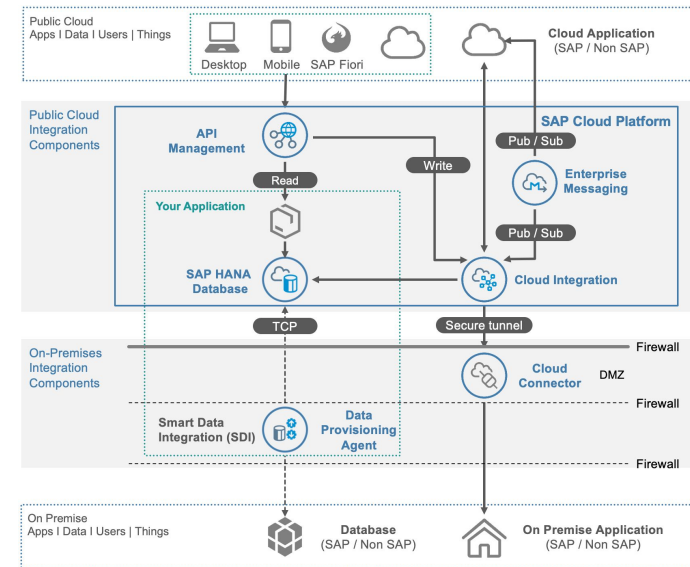
SAP Integrationsmöglichkeiten

OnPrem



Quelle: [7]

Cloud



© SAP

Klassisch

- Java Connector (Jco)
- .Net Connector

Aktuell

- Web Services
- OData Rest Services

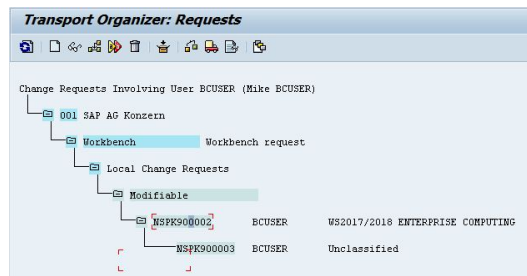
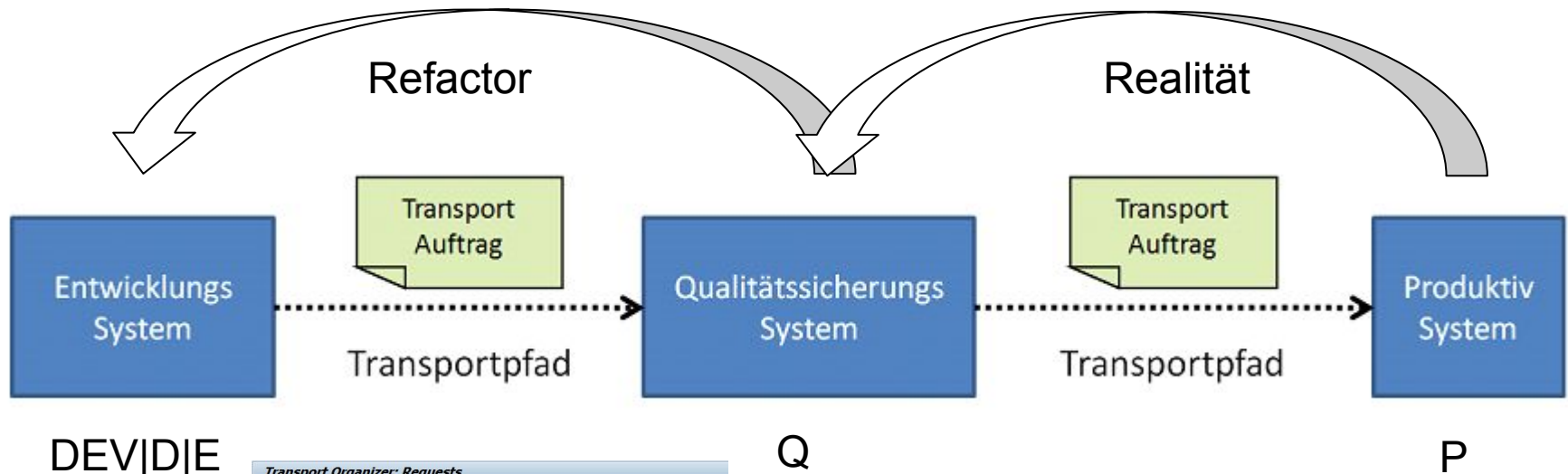
- ...

Weiters

- Operationsmodus Server oder Client

Begriffe und Handgriffe

SAP 3-System Landschaft

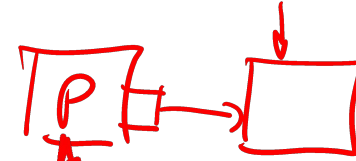


Begriffe und Handgriffe

SAP Upgrade

- Original (von SAP oder Kunde)
- Kopie (Ist das Original im Folgesystem)
- Modifikation (Änderung an der Kopie der SAP) ~~X~~
- Modifikationsableich (Abgleich der Modifikationen mit den neuen Auslieferungen der SAP)

ERWERTERUNG ✓



Auslieferung

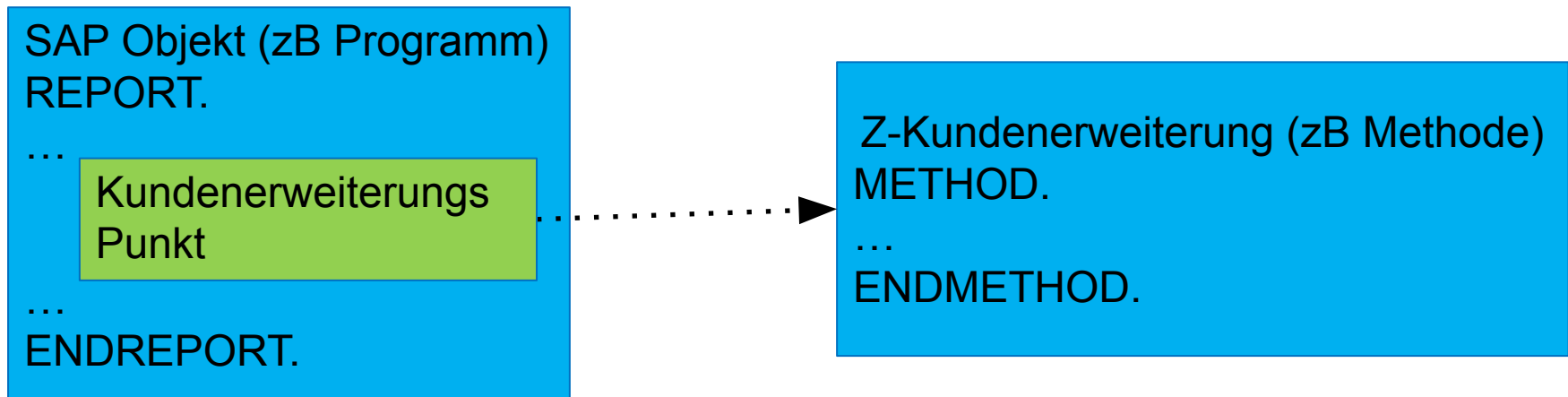
Korrektur



Begriffe und Handgriffe

SAP Erweiterbarkeit

- **Prinzip** (SAP sieht einen Absprung für kundeneigene Implementierung vor = Entwicklungsobjekt im Z-Namensraum)
- **Enhancement Framework**



Begriffe und Handgriffe



SAP Standard- vs. Kundenentwicklung

- SAP liefert **Standardfunktionalität**
- Kunden realisieren eigene Prozesse, um einen Wettbewerbsvorteil zu realisieren, d.h. **Kundenentwicklung (Custom Development)**

Begriffe und Handgriffe

SAP Anmeldung



[SAP Development Tools \(ondemand.com\)](https://www.ondemand.com)



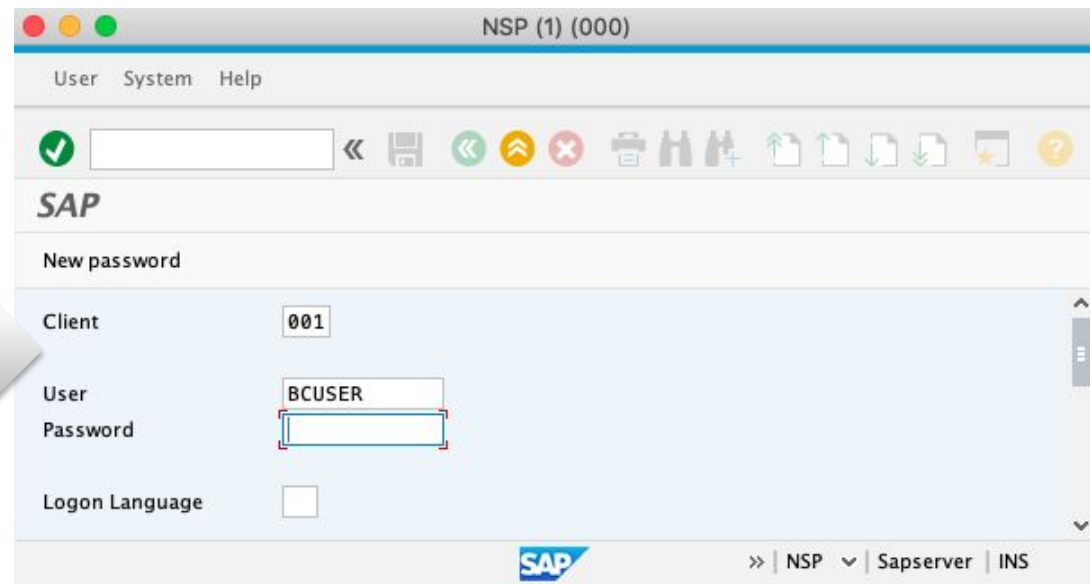
SAPLogon



/nex = Beenden aller Modi

Begriffe und Handgriffe

SAP Anmeldung



Begriffe und Handgriffe



- Anmelden

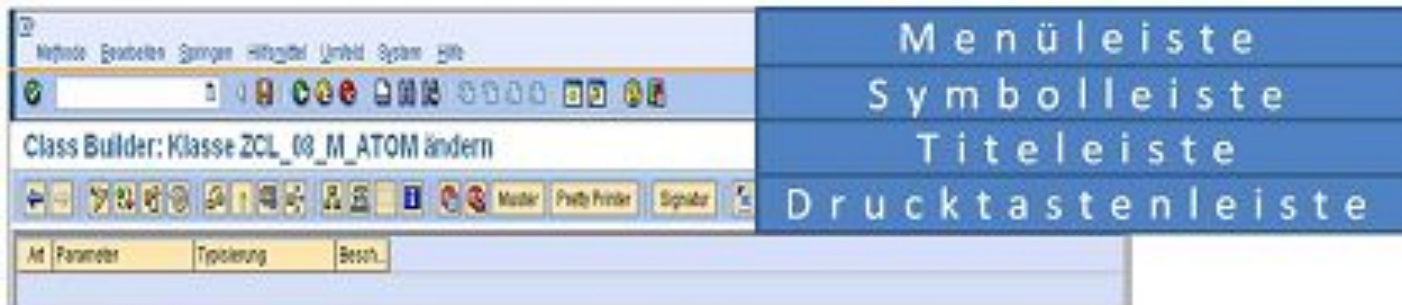
Begriffe und Handgriffe

SAP GUI Aufbau Die GUI Oberfläche eines SAP

Dynpros besteht aus einer **Menüleiste** (mit variablen und den fixen Menüs System und Hilfe), einer **Symbolleiste** (immer gleich aufgebaut), der **Titelleiste** und der **Drucktastenleiste**. Über

Funktionstasten-zuordnungen können Funktionen direkt

au:



Begriffe und Handgriffe



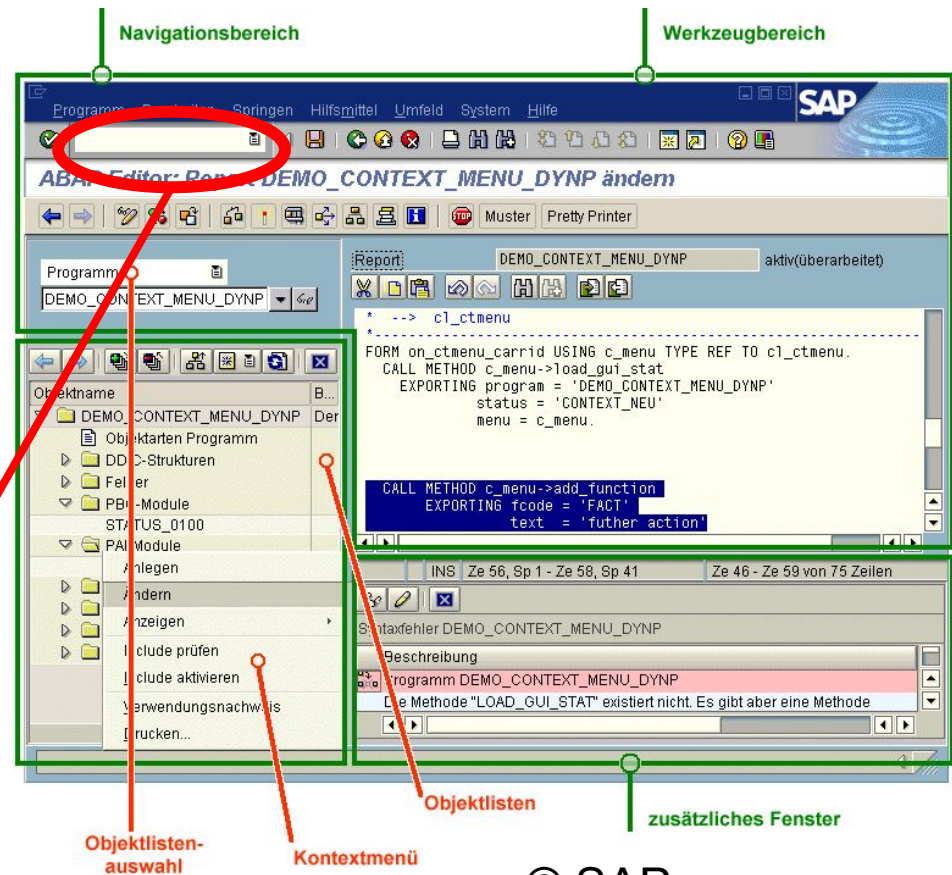
DER Transaktionscode:

SE80

Object Navigator

/n = neuer Start
/o = neues Fenster
/nex = Aussteigen

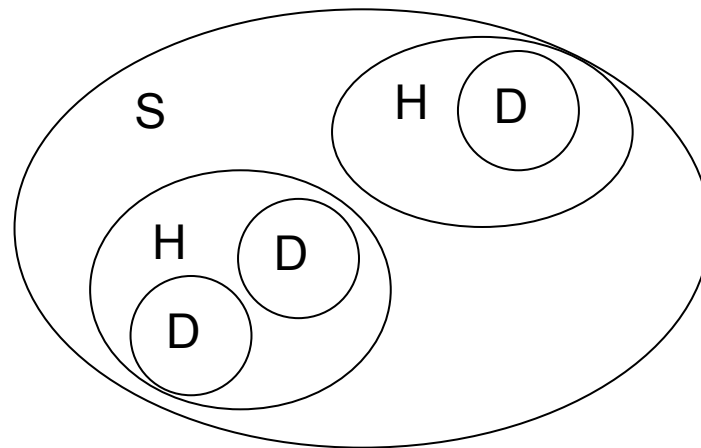
/n + Transaktion,
zB /nse80 = Object Navigator



Begriffe und Handgriffe

Entwicklungsorganisation Pakete (~Java Package)

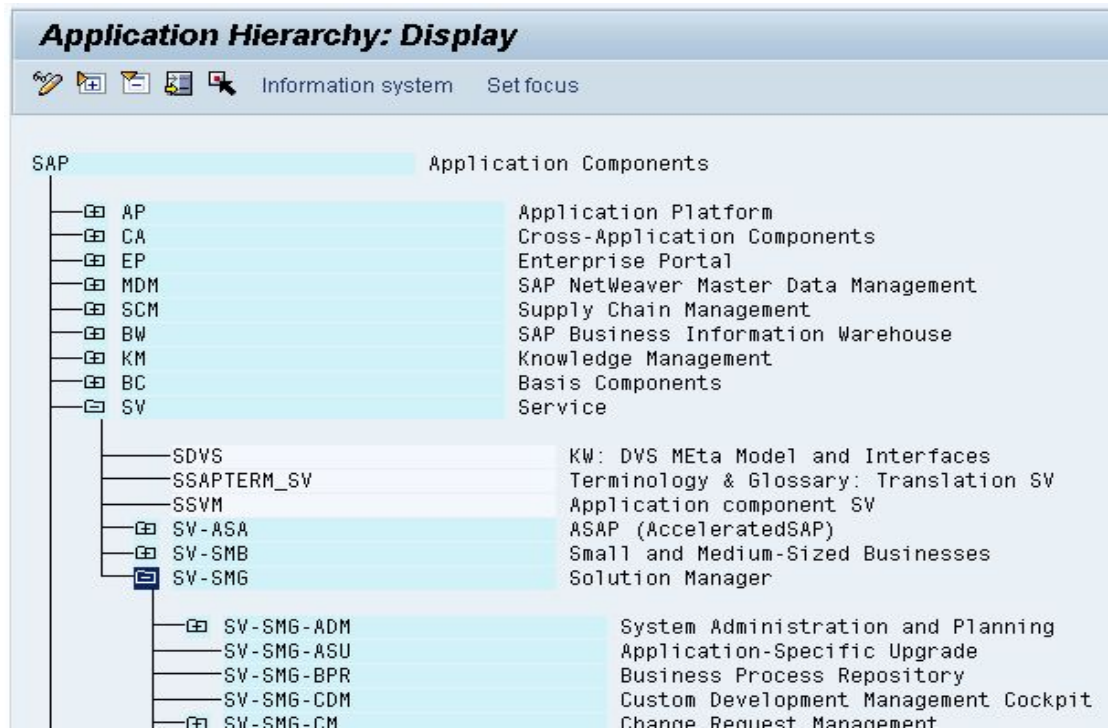
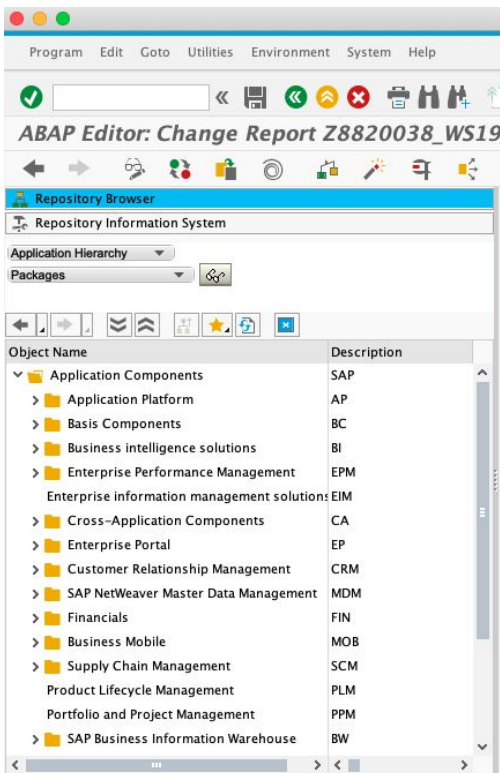
- Strukturpaket (S)
- Hauptpaket (H)
- Standard Paket (“Kein Hauptpaket”, Entwicklungspaket) (D)



Begriffe und Handgriffe



Entwicklungsorganisation Anwendungskomponente



Begriffe und Handgriffe



Entwicklungsorganisation Softwarekomponente

Diese beschreibt eine Menge von **Entwicklungsobjekten**, die nur **gemeinsam auslieferbar** sind.

Ausnahme: Pakete, die nicht an Kunden ausgeliefert werden sollen, müssen abweichend davon der Softwarekomponente **HOME** zugeordnet werden.

Begriffe und Handgriffe

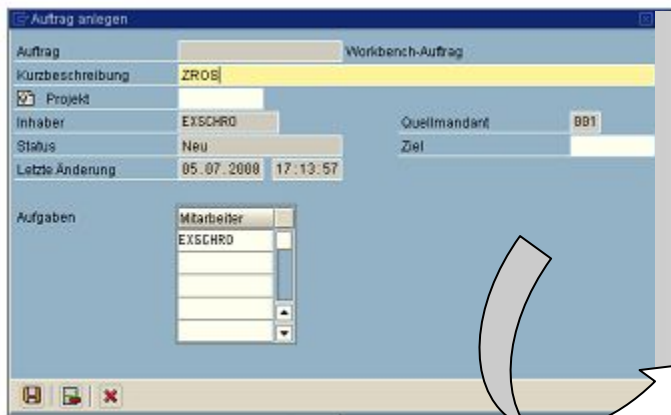


Entwicklungsorganisation Transporterschicht

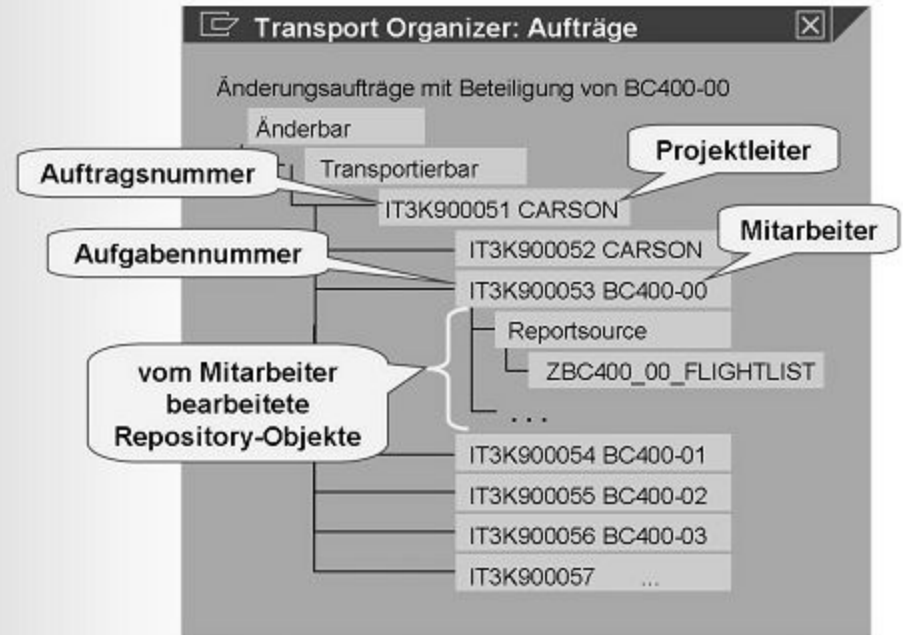
Alle Entwicklungsprojekte, die in einem SAP-System durchgeführt und auf denselben **Transportwegen** transportiert werden, werden zu einer **Transporterschicht** zusammengefasst.

Begriffe und Handgriffe

Entwicklungsorganisation **Änderungsauftrag**



- Projektleiter:
- CARSON
- Team:
- CARSON
 - BC400-00
 - BC400-01
 - BC400-02
 - BC400-03
 - ...



Begriffe und Handgriffe

Entwicklungsorganisation: Paket anlegen bzw. Lokales Paket \$TMP

The screenshot shows a dialog box titled "Package Builder: Paket anlegen". It contains several input fields and a checkbox:

Paket	ZROS
Kurzbeschreibung	ROS
Anwendungskomponente	ca
Softwarekomponente	HOME
Transportschicht	<input type="radio"/>
Hauptpaket	<input type="checkbox"/>

At the bottom left of the dialog, there are three icons: a document icon, a red 'X' icon, and an information icon.

Begriffe und Handgriffe

Namensräume

SAP liefert eigene Softwareentwicklungen an den Kunden aus, die dieser in seine Systeme übernimmt. Der Kunde kann eigene Softwareentwicklungen durchführen. Prinzipiell kann es zu **Namensüberschneidungen** zwischen **SAP Objekten** und **Kunden Objekten** kommen.

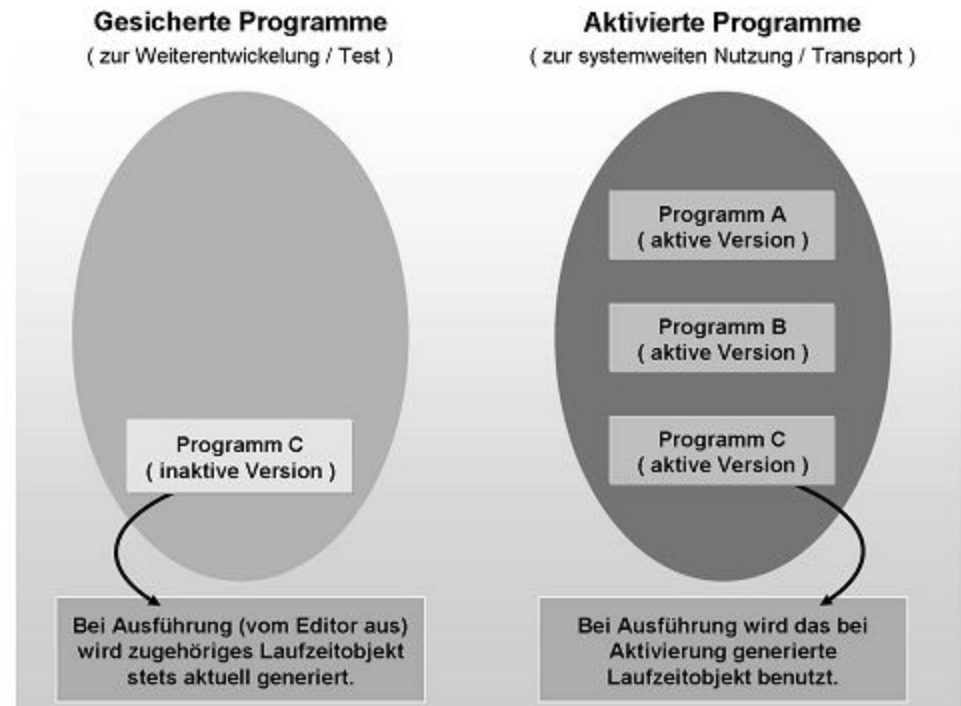
Es gibt **zwei Varianten** (Namensräume), um Kundenobjekte zu schützen.

1. Wenn Entwicklungsobjekte mit **Z oder Y beginnen**, dann sind diese implizit geschützt, da SAP keine Objekte mit Z oder Y beginnend ausliefert (Ausnahme: Kunden-Exits)
2. Der Kunde kann bei SAP einen weltweit eindeutigen **Namensraum** beantragen. Dieser ist 10stellig und beginnt mit / und endet mit /. Dieser Namensraum muss den Entwicklungsobjekten vorangestellt werden. Diese Variante ist speziell für Drittanbieter interessant.



Begriffe und Handgriffe

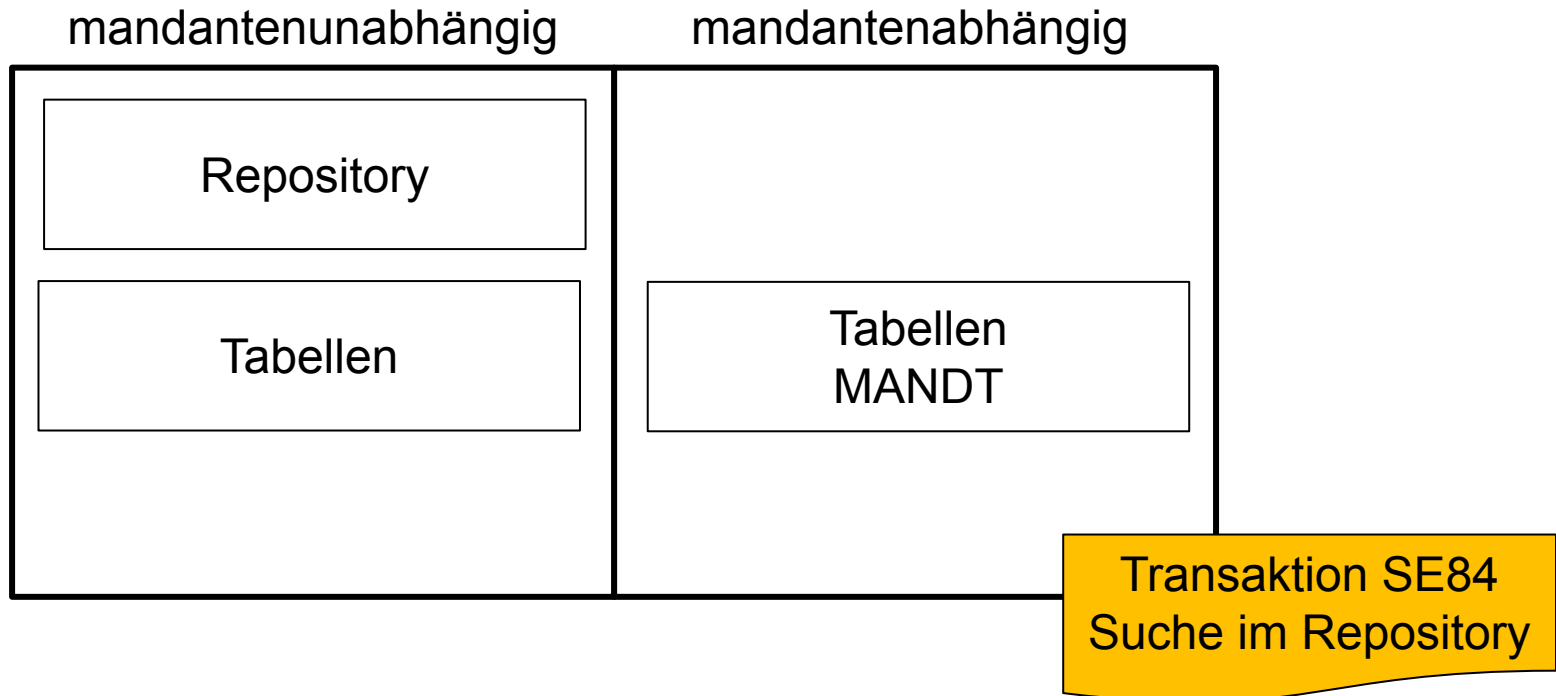
- **Repository Objekte**
 - Aktivitätszustände
 - Programmarten
 - Funktionsbausteine
 - ...
- **Dictionary Objekte**
 - Aktivitätszustände
 - Domäne
 - Datenelement
 - Struktur
 - Transparente Tabelle



Begriffe und Handgriffe

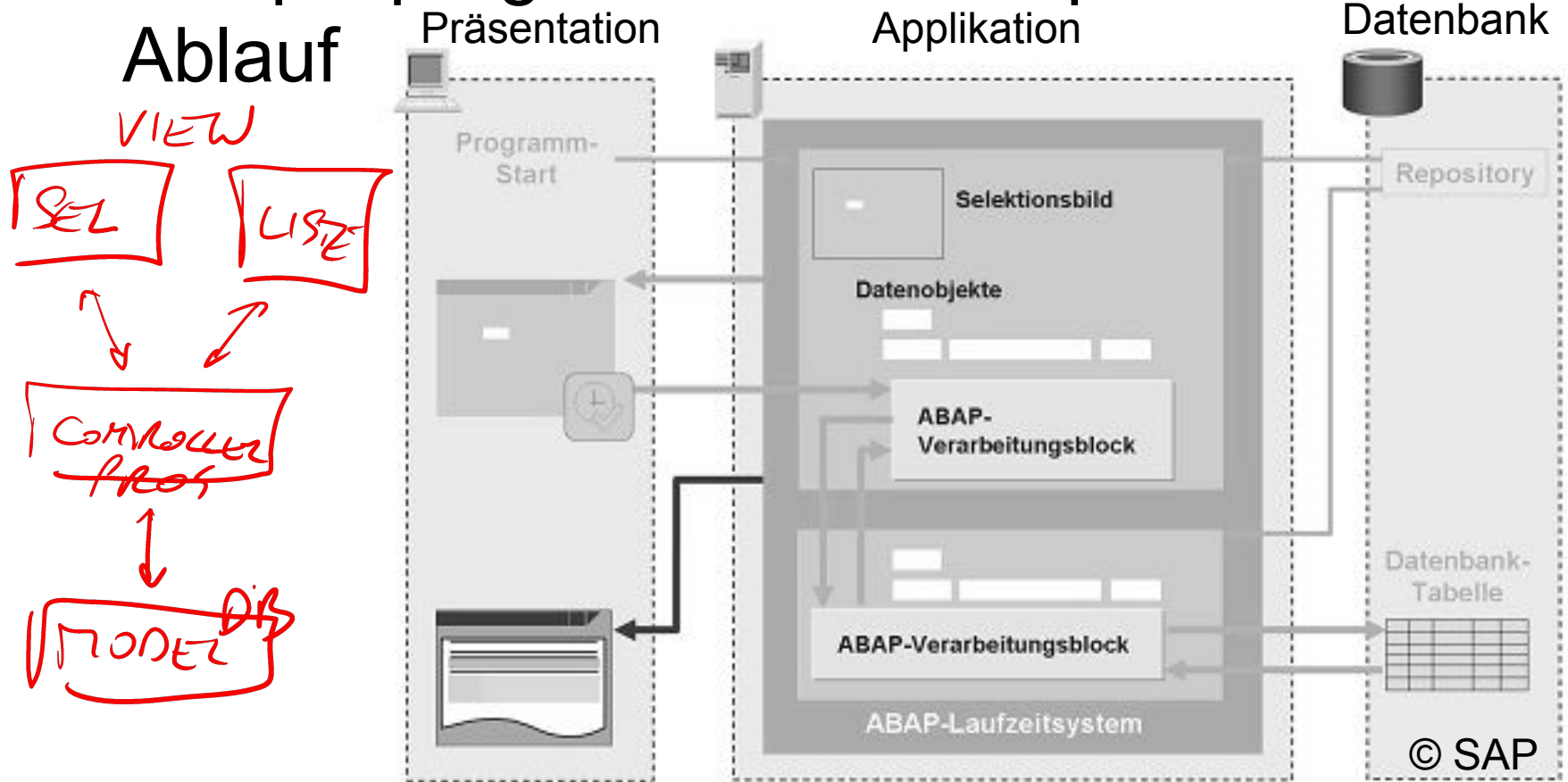
- **Repository Objekte (Aufbau Repository)**

Datenbank



Begriffe und Handgriffe

- Beispielprogramm mit exemplarischem Ablauf



Beispiel



```
*&-----*
*& Report Z8820038_WS19_HELLO_WORLD
*&
*&-----*
*&
*&
*&-----*
```

REPORT z8820038_ws19_hello_world.

* Eingabefeld für Fluggesellschaft
PARAMETERS: pa_car TYPE scarr-carrid.

* Ziel Datenobjekt für Name der FG
DATA gd_carname TYPE scarr-carname.

INITIALIZATION.

* Initialisierung für Eingabefeld
pa_car = 'LH'.

LOAD-OF-PROGRAM.

START-OF-SELECTION.

* Lesen der FG von der DB Tabelle SCARR
SELECT SINGLE carname FROM scarr INTO gd_carname
WHERE carrid = pa_car.

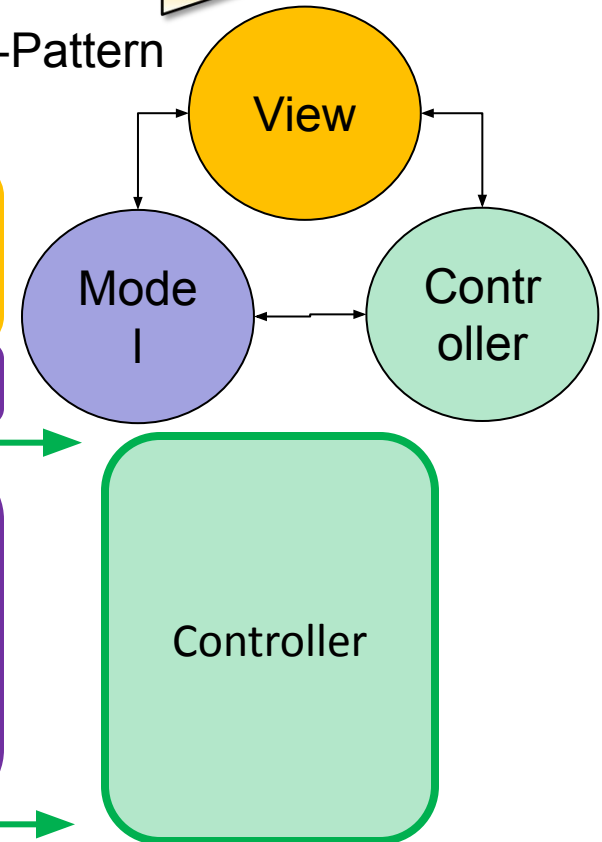
END-OF-SELECTION.

* Ausgabe auf Standardliste
WRITE 'Hello World!'.
* Neue Zeile
new-line.

* Ausgabe des Names der FG
WRITE: / 'Name der FG: ', gd_carname.

Beispielprogramm

MVC-Pattern



REPORT zwatjab

```

PARAMETERS: pa_carr TYPE bapisdeta-carrid default 'LH',
pa_cofro TYPE bapisdeta-countryfr default 'US',
pa_cifro TYPE bapisdeta-cityfrom default 'NEW YORK',
pa_coto TYPE bapisdeta-countryto default 'DE',
pa_cito TYPE bapisdeta-cityto default 'FRANKFURT',
pa_after TYPE bapi_aux-afternoon AS CHECKBOX.
  
```

Selektionsbild = View

```

DATA: it_bapislist TYPE TABLE OF bapislist,
wa_bapislist LIKE LINE OF it_bapislist.
  
```

Geschäftsdaten = Modell

START-OF-SELECTION.

```
CALL FUNCTION 'ZBAPISLIST_FUGHT_GETLIST'
```

```

EXPORTING
fromcountrykey = pa_cofro
fromcity       = pa_cifro
tocountrykey   = pa_coto
tocity         = pa_cito
airlinecarrier = pa_carr
afternoon      = pa_after
* MAXREAD      = 0
* IMPORTING
* RETURN       =
TABLES
flightlist    = it_bapislist
  
```

Geschäftslogik = Modell

END-OF-SELECTION.

```
LOOP AT it_bapislist IN O wa_bapislist.
```

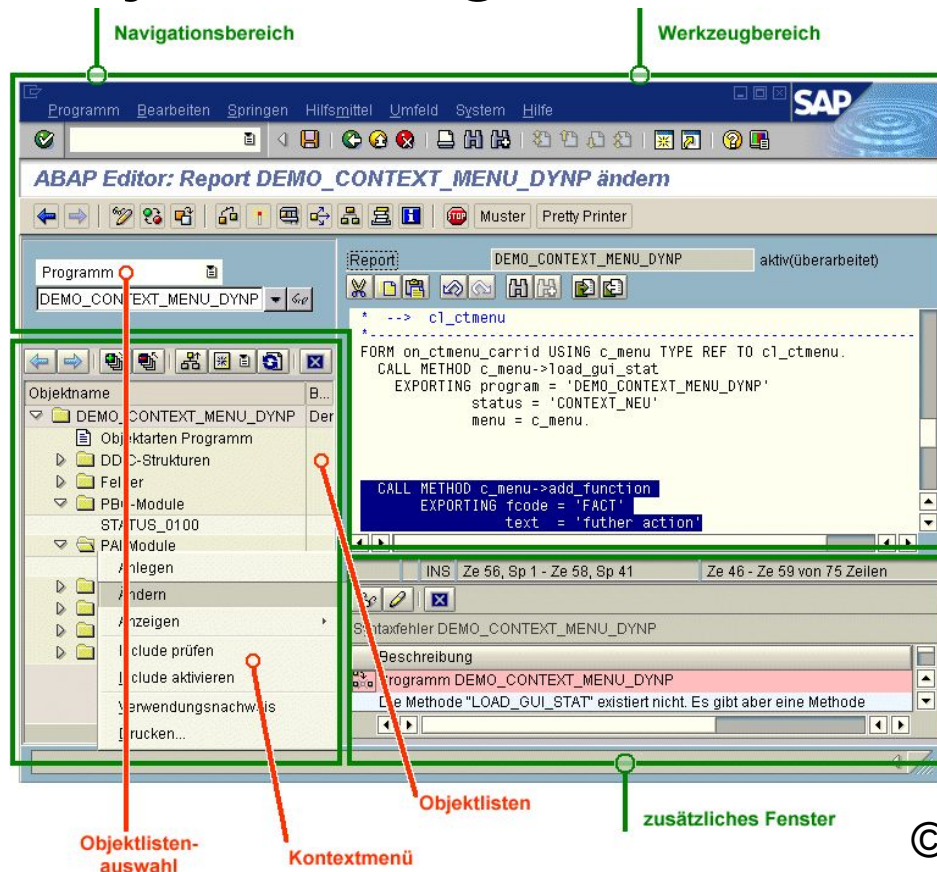
```

WRITE: / wa_bapislist-carrid,
wa_bapislist-connid,
wa_bapislist-fldate,
wa_bapislist-airpfrom,
wa_bapislist-airpto,
wa_bapislist-deptime,
wa_bapislist-seatsmax,
wa_bapislist-seatsocc.
ENDLOOP.
  
```

Ergebnisliste = View

Motivation SAP

- Object Navigator und Werkzeuge



ABAP Editor
Debugger
ABAP Dictionary
Screen Painter
Menu Painter
Function Builder
Class Builder

...

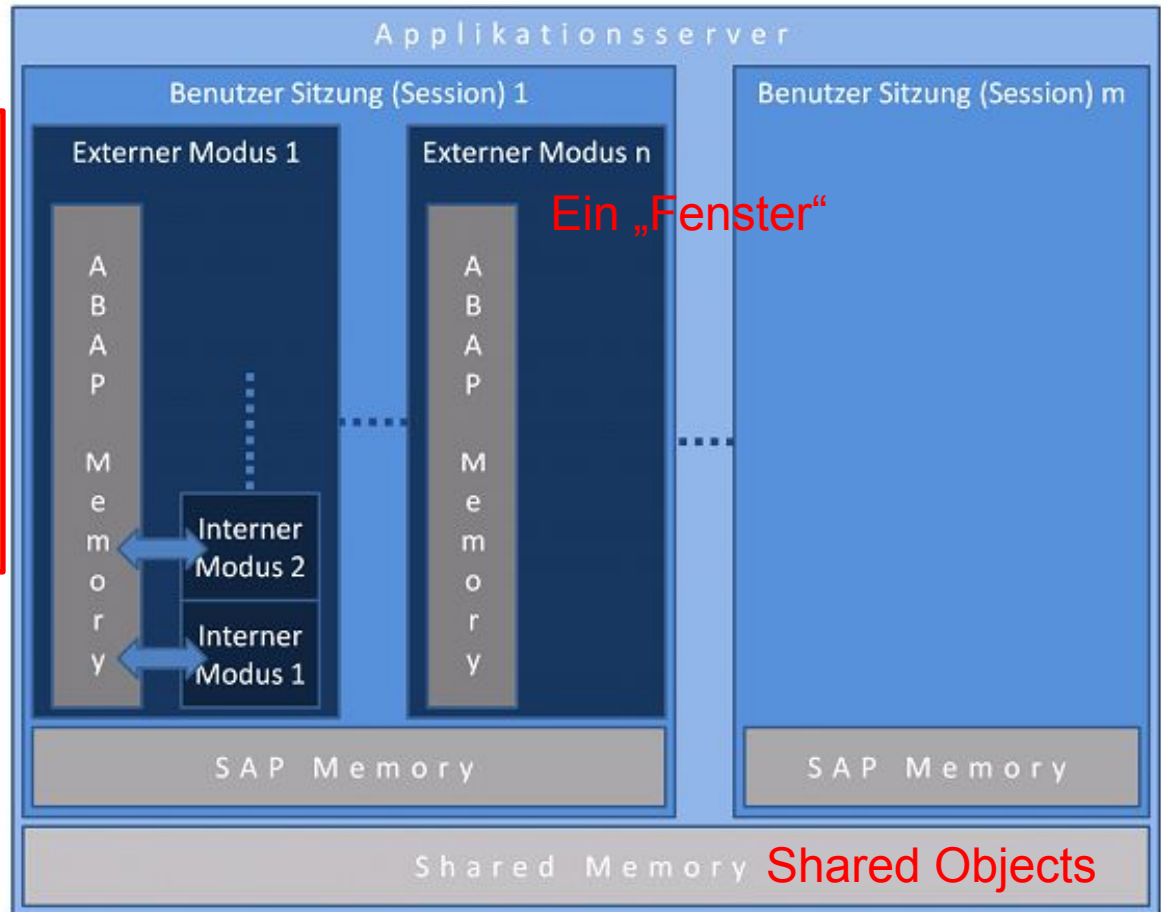
© SAP

Motivation SAP

Speichermodell

/n + Transaktion = zurücksetzen vom aktuellen externen Modus und Programmstart

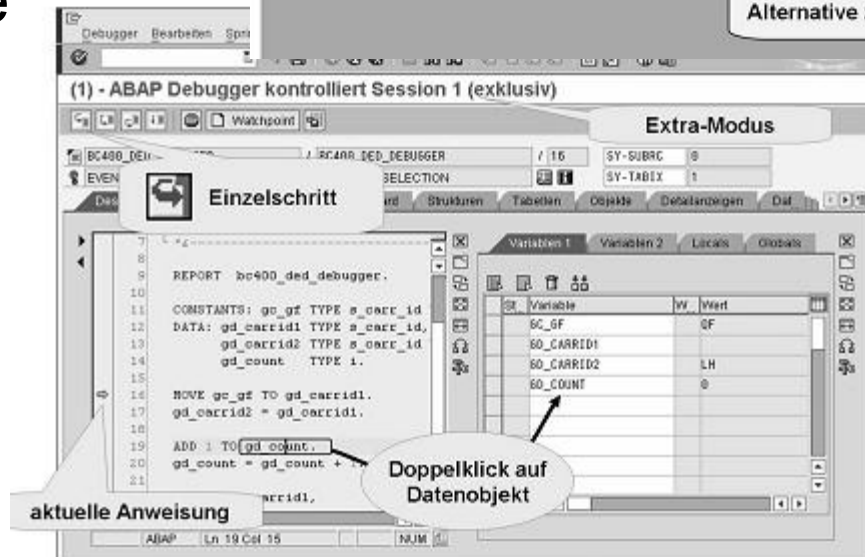
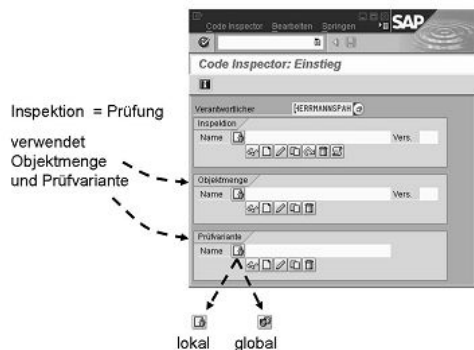
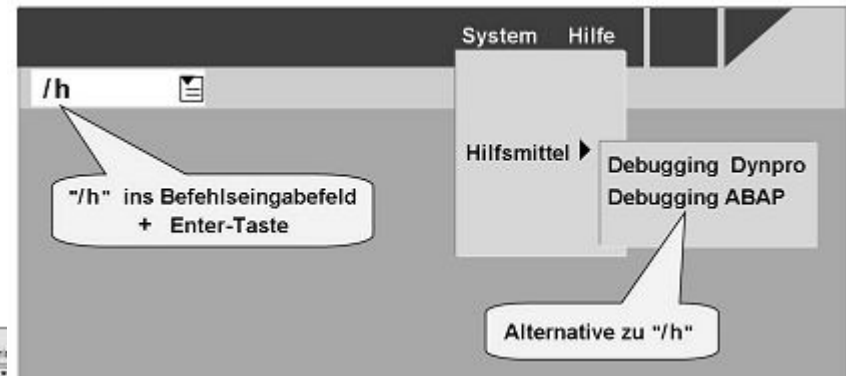
/o + Transaktion = neuer externer Modus und Programmstart



Statische vs. Dynamische Analyse

- **Debugger**
- Code Inspector
- Prüfvariante
- Objektmenge
- Inspektion

SCI



DB Dictionary

Dictionary

ABAP Dictionary: Transparente Tabelle

- Motivation: Dient zur Definition einer DB-Tabelle
- Anlegen: TA SE11
- Indizes
 - Primär
 - Sekundär
- Technische Einstellungen



Typarten

Elementar



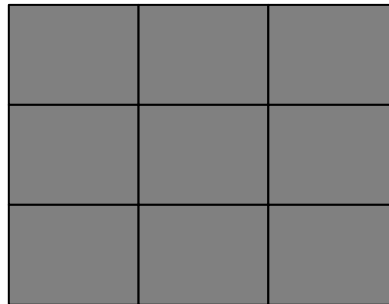
zB SCARR-CARRID -> DE
S_CARR_ID siehe 2schichtiges
Domänenkonzept SAP

Strukturiert



zB SCARR

Tabellenartig
(Tabellentyp)



Dictionary

ABAP Dictionary: Transparente Tabelle

- Spaltendefinition:
 - Name der Spalte
 - Teil des Primärschlüssels?
 - Initialisierung
 - Typisierung

Transp. Table: ZSCARREVAL Active
Short Description: Bewertung der FG

Attributes | Delivery and Maintenance | **Fields** | Entry help/check | Currency/Quantity Fields

Field	Key	Initi..	Data element	Data Type	Length	Decim..	Short Description
MANDT	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	MANDT	CLNT	3	0	Client
CARRID	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	S_CARR_ID	CHAR	3	0	Airline Code
EVAL	<input type="checkbox"/>	<input checked="" type="checkbox"/>	Z8820038_EVAL	INT1	3	0	Rating

Dictionary

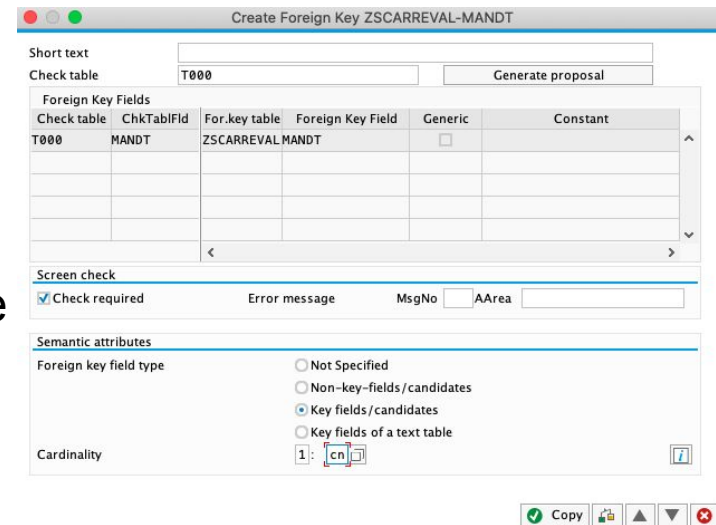
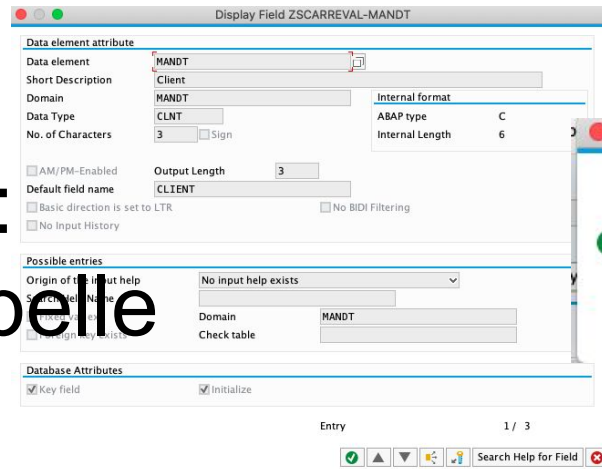
ABAP Dictionary: Transparente Tabelle

Durch die Pflege einer

Fremdschlüsselverprobung wird **nicht**

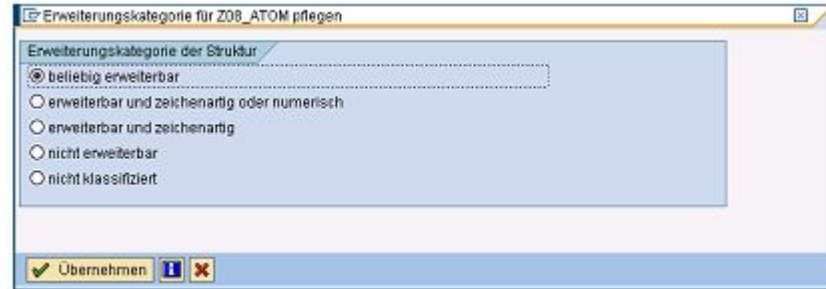
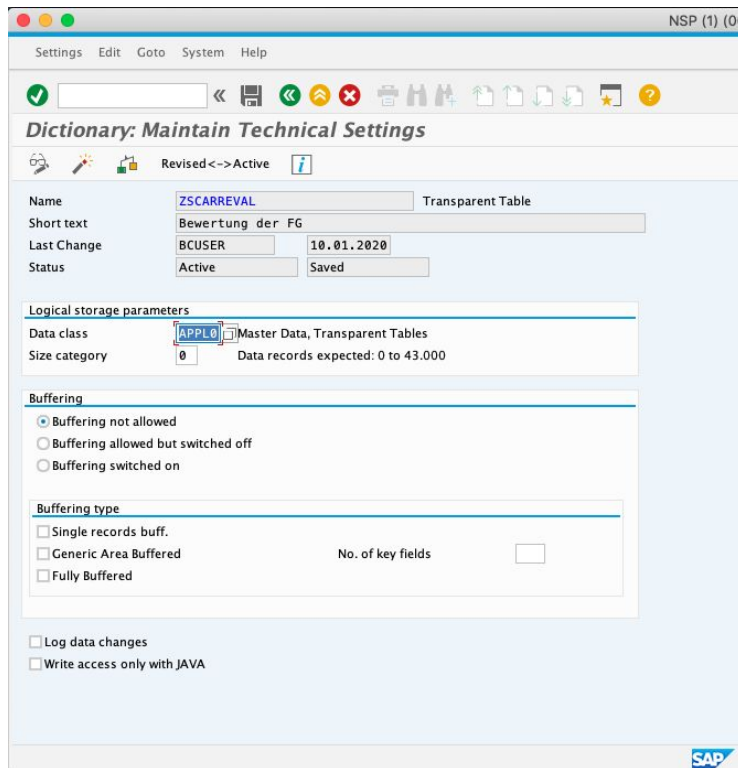
referentielle Integrität auf Datenbankeben
definiert.

Die Definition dieser Verprobung hat nur dann eine
Auswirkung, sofern ein Dynpro- oder ein Web
Dynpro-Feld definiert wird, dass einen Bezug zu
dem Fremdschlüsselfeld herstellt.



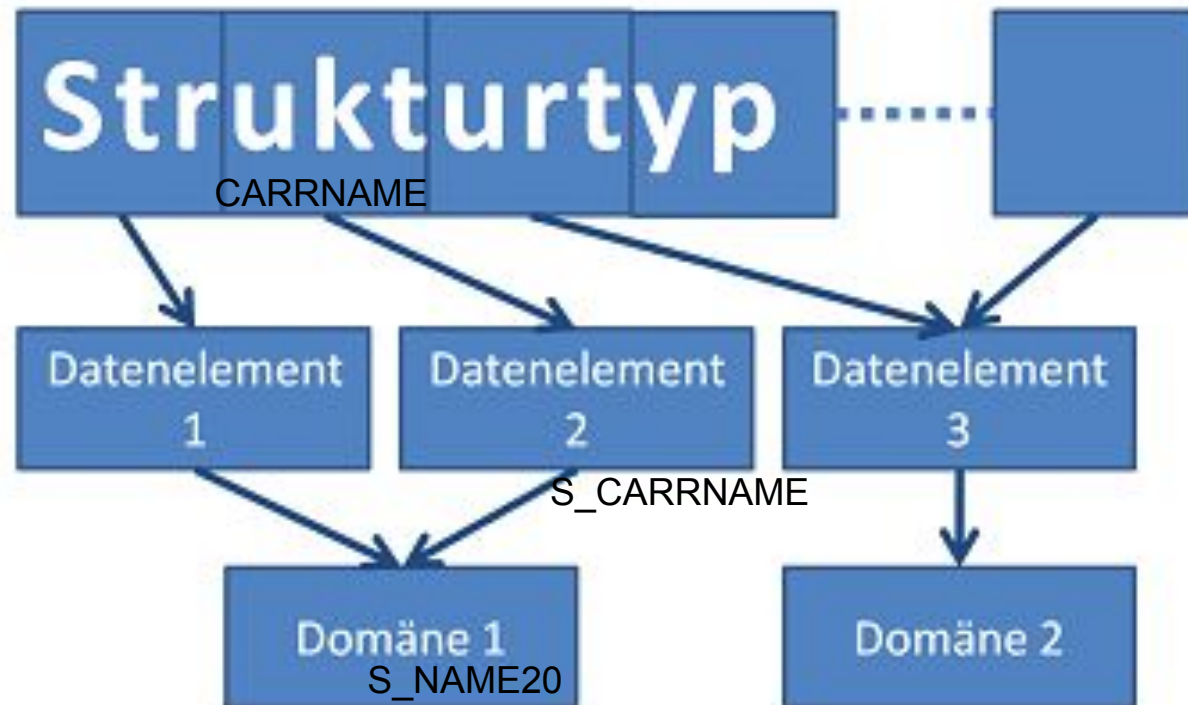
ABAP Dictionary: Transparente Tabelle

Technische Einstellungen & Erweiterbarkeit



Dictionary

Zweischichtiger Domänenansatz im ABAP Dictionary



Sematische
Information

Technische
Information

Dictionary

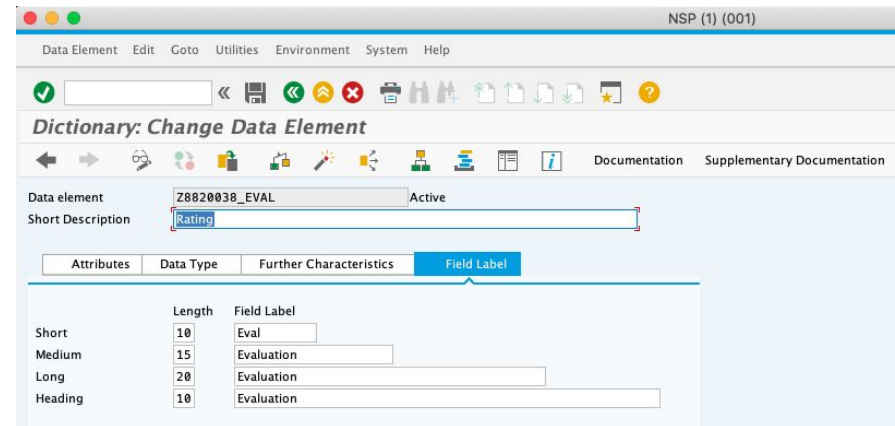
ABAP Dictionary: Datenelement

- Motivation: Definition von globalen skalaren Typen im ABAP Dictionary (betriebswirtschaftliche oder auch semantische Ebene)

- Anlegen: TA SE11

- Verwendung:

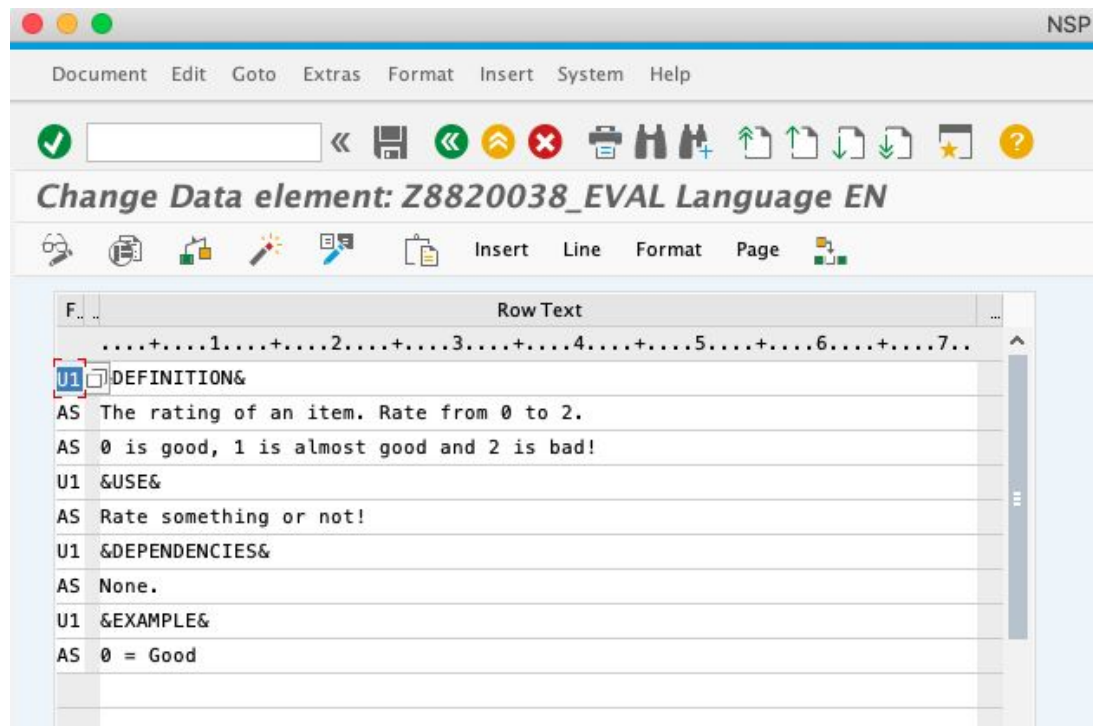
Typisierung von Variablen, Schnittstellen, ...



Dictionary

ABAP Dictionary: Datenelement

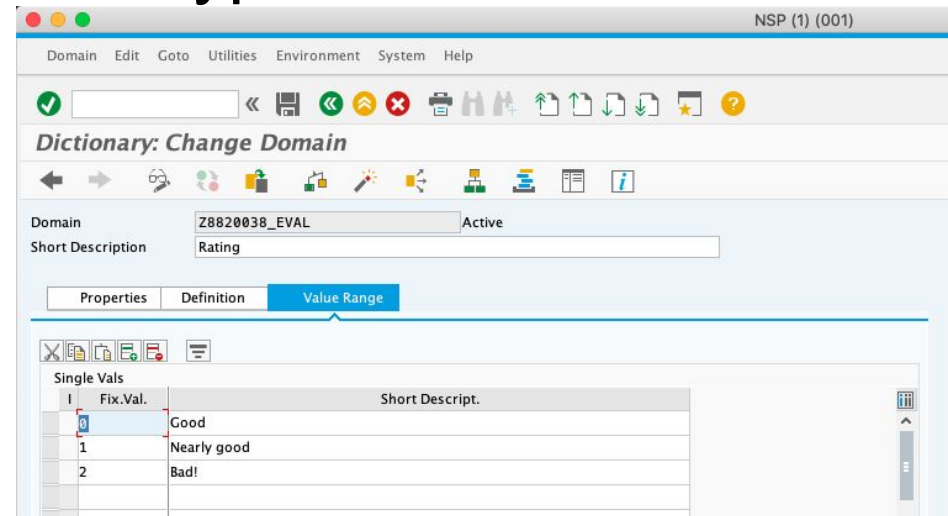
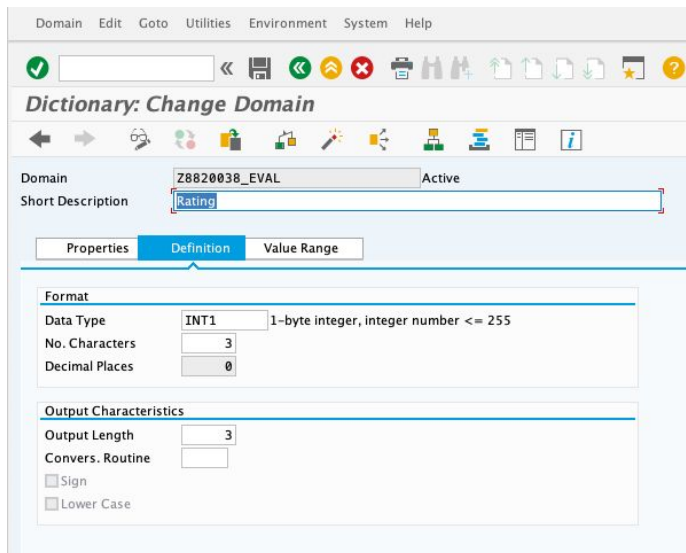
- Dokumentation !



Dictionary

ABAP Dictionary: Domäne

- Motivation: Dient zur Definition von globalen technischen Typen mit Festwerten
- Anlegen: TA SE11
- Verwendung: Technischer Typ für Datenelemente



ABAP Prozedural

“Die Sprache”

- ist proprietär,
- ist typisiert,
- ermöglicht mehrsprachige Anwendungen,
- ermöglicht SQL-Zugriffe, (OpenSQL, NativeSQL)
- ist objektorientiert erweitert worden,
- ist plattformunabhängig,
- ist aufwärtskompatibel.

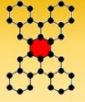
Syntax

Allgemeiner Aufbau einer ABAP-Anweisung

X X X	Y Y Y	.
ABAP-Schlüsselwort	Zusätze und Operanden (schlüsselwort-spezifisch)	Punkt als Abschluss der Anweisung

Programmbeispiel

```
PARAMETERS pa_num TYPE i.  
DATA gv_result TYPE i.  
MOVE pa_num TO gv_result.  
ADD 1 TO gv_result.  
WRITE 'Your input:'  
WRITE pa_num  
NEW-LINE .  
WRITE 'Result: '  
WRITE gv_result.
```



message_class =
Nachrichtenklasse = Transaktion
SE91

Nachrichten (Messages)

```
MESSAGE @nnn (message_class) [ WITH v1 [ v2 ] [ v3 ] [ v4 ] ] .
```

Typ	Bedeutung	Dialogverhalten	Nachricht erscheint in
i	Info-Nachricht	Programm wird nach Unterbrechung fortgesetzt	modalem Dialogfenster
s	Setznachricht	Programm wird ohne Unterbrechung fortgesetzt	Statuszeile *) des Folgebilds
w	Warnung	kontextabhängig	Statuszeile *)
e	Fehler	kontextabhängig	Statuszeile *)
a	Abbruch	Programm wird abgebrochen	modalem Dialogfenster
x	Kurzdump	Laufzeitfehler MESSAGE_TYPE_X wird ausgelöst	Kurzdump

AT SELECTION-SCREEN



AT SELECTION-SCREEN.

IF pa_eval > 2.

MESSAGE e001(zws19) WITH pa_eval.

* The evaluation & is not valid

ENDIF.

ABAP

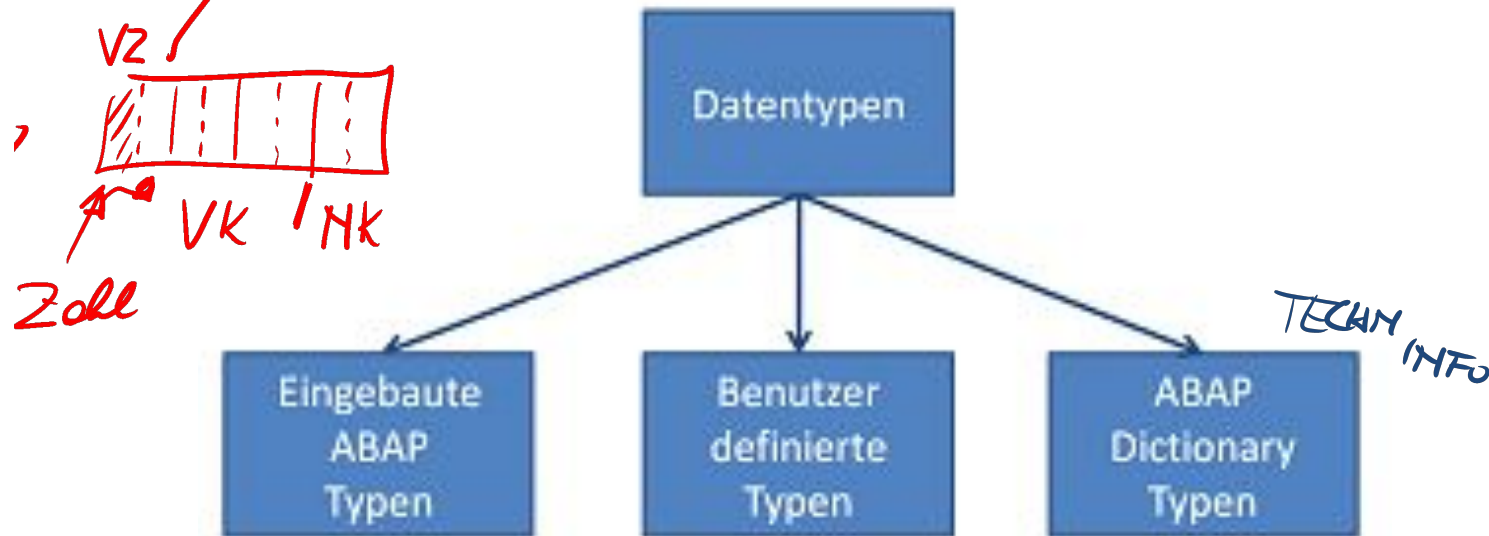


Einige wichtige Anweisungen:

- DATA
- TYPES
- CONSTANTS
- WRITE

Datentypen

Bei der Typisierung von Variablen können **eingebaute** ABAP Typen (d, t, i, c, p, n, f, string, xstring), **benutzerdefinierte** Typen (TYPES) oder **globale Typen** aus dem **ABAP Dictionary** (Datenelemente, Strukturtypen, Tabellentypen) verwendet werden.



Typen Arten:

- Elementare Typen
- Strukturierte Typen

* Zeilentyp für Datenpufferung zu FG und Eval

```
TYPES: BEGIN OF gts_fg_eval,  
        carrid TYPE scarr-carrid,  
        carrname TYPE scarr-carrname,  
        eval TYPE zscarreval-eval,  
END OF gts_fg_eval.
```

- Tabellenartige Typen

* Tabellentyp (Standard Tabelle) mit Zeilentyp
gtt_fg_eval TYPE TABLE OF gts_fg_eval.

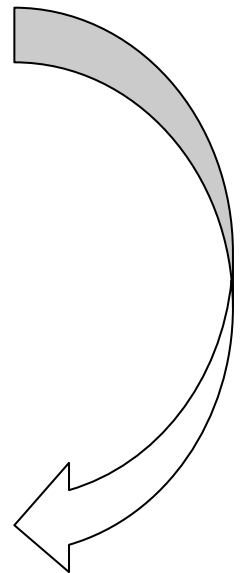
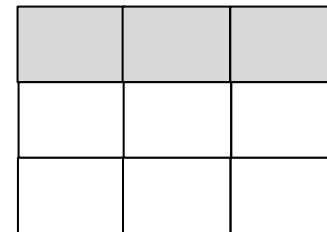


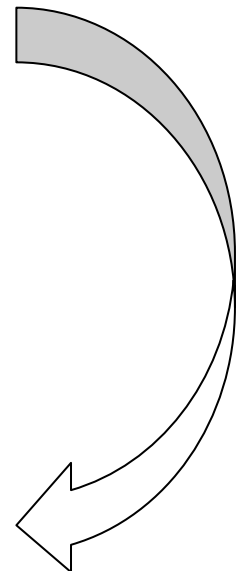
Tabelle befüllen

SELECT * FROM SCARR ...

SELECT eval FROM Z<MATNR>CAREVAL ...

carrid carrname eval

--	--	--



Debugging



The screenshot shows the SAP ABAP Debugger interface. The main window displays the source code of a program, with a red arrow pointing to a loop. The code includes comments in German and a loop that writes data to a table. Below the code, a table shows the field names and their contents. The table has 4 columns and 3 rows. The first row shows 'gt_fg_eval' with content 'Table[9x48]'. The second row shows 'gs_fg_eval-carrid' with content 'SQ'. The third row shows 'gs_fg_eval' with content 'Non-charlike structure'. At the bottom, there are status fields for SY-SUBRC (0), SY-TABIX (9), SY-DBCNT (9), and SY-DYNNR (1000). A green checkmark and the text 'Field copied into the display' are visible at the bottom left.

Debugging Edit Goto Breakpoints Settings Development System Help

ABAP Debugger

Fields Table Breakpoints Watchpoints Calls Overview Settings

Main Program Z8820038_WS19_HELLO_WORLD

Source code of Z8820038_WS19_HELLO_WORLD

EVENT END-OF-SELECTION

- * Ausgeben der FG mit bestimmten Bewertung
- ULINE.
- * Zeile für Zeile werden die Daten aus der Tabelle
- * ausgegeben

```
→ LOOP AT gt_fg_eval INTO gs_fg_eval.  
    WRITE: / gs_fg_eval-carrid,  
           gs_fg_eval-carrname,  
           gs_fg_eval-eval.  
ENDLOOP.
```

Field names	Field contents
gt_fg_eval	Table[9x48]
gs_fg_eval-carrid	SQ
gs_fg_eval	Non-charlike structure

SY-SUBRC 0 SY-TABIX 9 SY-DBCNT 9 SY-DYNNR 1000

Field copied into the display

Strukturierte Typen:

- Struktur (Dictionary oder programmlokal)
- Transparente Tabelle (Dictionary)
- View (Dictionary)

Literale

Fixe Datenobjekte ohne Bezeichner

Literale

Zahlenliterale	Textliterale
positive ganze Zahl : 123	Zeichenketten : 'Hallo'
negative ganze Zahl : -123	Bezimalzahlen : '123.45'
	Gleitpunktzahlen : '123.45E01'

Fixe Datenobjekte mit Bezeichner

Konstanten

```
CONSTANTS gc_myconst TYPE type_name VALUE {literal | IS INITIAL}.
```

ABAP

Beispiele

```
* comments ... }
* comments ... }
* comments ... }

PARAMETERS pa_num TYPE i.
DATA gv_result TYPE i.

MOVE pa_num
  TO gv_result.

ADD 1 TO gv_result.

WRITE : 'Your input:'
      pa_num.

NEW-LINE.

WRITE : 'Result: ' , gv_result.
```

Kommentare
(ganze Zeilen)

Kommentare
(Zeilenrest)

Kettensatz

Kettensatz

ReturnCodes (sy-subrc)

- Wichtige Systemfelder werden in der ABAP Dict Struktur **SYST** definiert.
- Das wichtigste ist **sy-subrc**, welches die Aussage über den Ausgang der vorherigen Operation liefert.

IF sy-subrc = 0. “Alles ok

...

ELSE. “Da lief was falsch

...

ENDIF.

```
* Lesen der FG von der DB Tabelle SCARR
SELECT SINGLE carrname FROM scarr INTO gd_carrname
WHERE carrid = pa_car.
IF sy-subrc <> 0.
  message e002(zws19) with pa_car.
ENDIF.
```

Modularisierung

- Ebene lokal – lokale Wiederverwendung
 - **Unterprogramme**
 - SAP GUI Module
 - Ereignisblöcke
- Ebene global – globale Wiederverwendung
 - **Funktionsbausteine** (Funktionsgruppen) – TA SE37
 - **ABAP Klassen (ABAP Interfaces) – TA SE24**

Unterprogramm

```
*&-----*
*&   Form READ_EVAL
*&-----*
*   Lesen der Bewertung
*-----*
*   -->ID_CARRID  ID der FG
*   <--ED_EVAL   Bewertung
*-----*
FORM read_eval USING   id_carrid TYPE scarr-carrid
                     CHANGING ed_eval TYPE zscarreval-eval.

SELECT SINGLE eval FROM zscarreval INTO ed_eval
WHERE carrid = id_carrid.

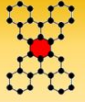
ENDFORM.                " READ_EVAL
```

ABAP Prozedural Struktur



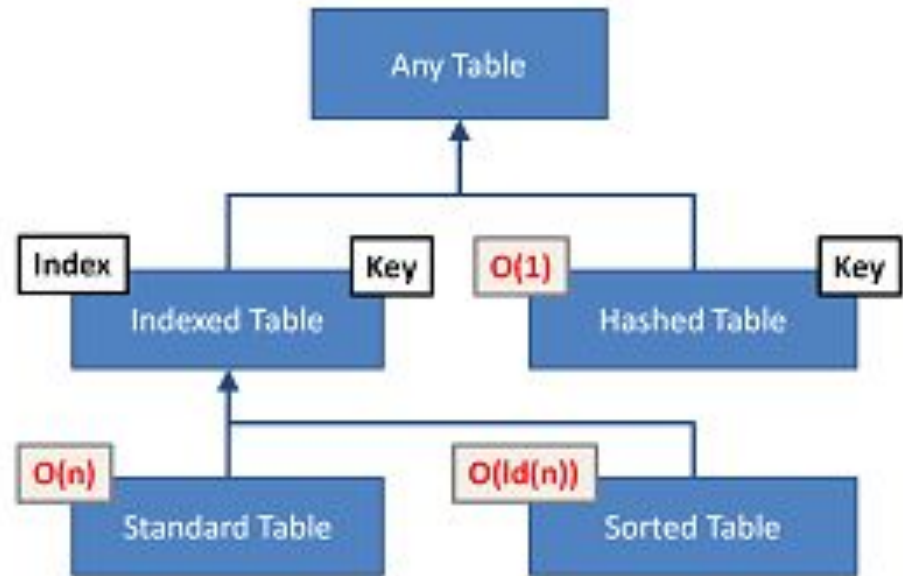
ABAP Dictionary: Struktur

- Motivation: Definition von globalem strukturiertem Typ (vergleichbar: Record, **1-dim Array**)
- Anlegen: TA SE11
- Verwendung: Typisierung von Variablen, Schnittstellen, Tabellentypen, ...
- Zugriff auf Komponenten: Bindestrich Semantik
- Kopieren von Inhalten: **MOVE-CORRESPONDING**



ABAP Dictionary: Tabellentyp (Typ einer internen Tabelle)

- Motivation: Definiert den globalen Typ einer internen Tabelle
- Anlegen: TA SE11
- Definition einer internen Tabelle
- Wichtige Anweisungen:
 - Sort <tabelle> BY <spalte> [ascending|decending]
 - APPEND <zeile> TO <tabelle>.
 - DELETE <zeile> FROM <tabelle>.

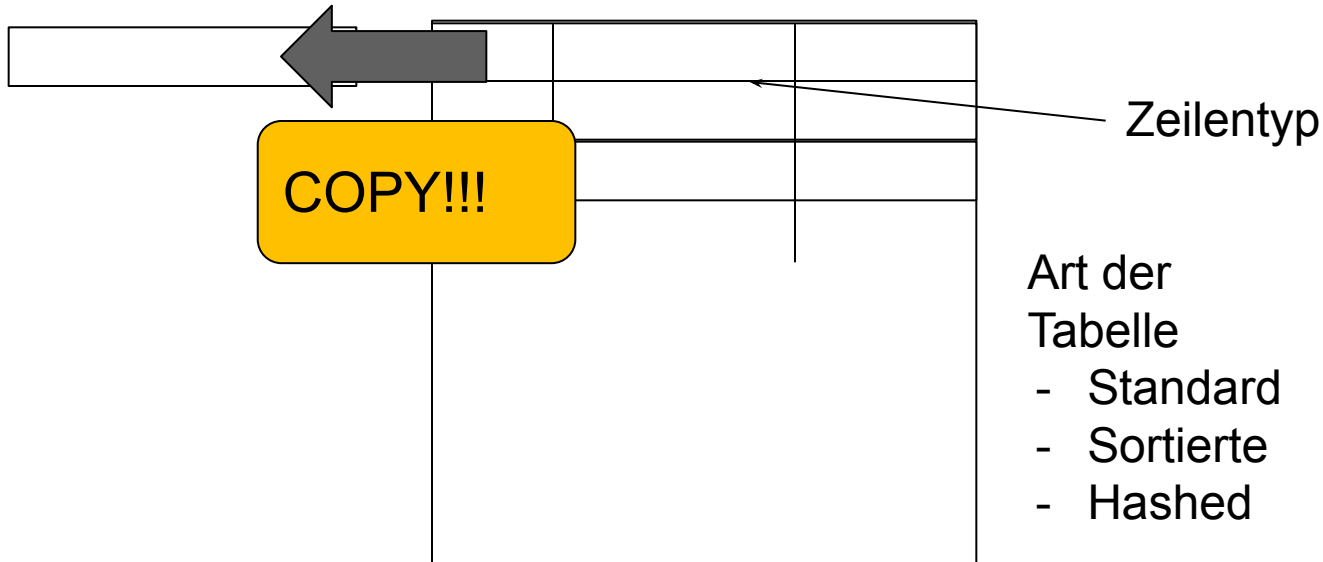


ABAP Procedural Datenhaltung

Interne Tabelle

Arbeitsstruktur (Struktur)

Ist im Hauptspeicher!



Art der
Tabelle

- Standard
- Sortierte
- Hashed

Schlüssel

- Feldern
- Reihenfolge
- Eindeutigkeit

Ausgabe Tabelle

```
*&-----*
*&   Form WRITE_FG_EVAL
*&-----*
*   Ausgabe der Tabelle auf die Liste
*-----*
*   -->IT_FG_EVAL Tabelle der FV und eval
*-----*
FORM write_fg_eval USING it_fg_eval TYPE gtt_fg_eval .

DATA: ls_fg_eval LIKE LINE OF it_fg_eval.

LOOP AT it_fg_eval INTO ls_fg_eval.
  WRITE: / ls_fg_eval-carrid,
         ls_fg_eval-carrname,
         ls_fg_eval-eval.
ENDLOOP.

ENDFORM.          " WRITE_FG_EVAL
```


Lesen in interne Tabelle



```
*-----*
*& Form READ_FG_EVAL
*&-----*
* Ermitteln der FG zu einer eval
*-----*
* -->Id_EVAL Eval
* <--ET_FG_EVAL FG zu eval
*-----*
FORM read_fg_eval USING id_eval TYPE zscarreval-eval
      CHANGING et_fg_eval TYPE gtt_fg_eval.

DATA: Id_carrid TYPE scarr-carrid,
      ls_fg_eval LIKE LINE OF et_fg_eval,
      Id_carrname TYPE scarr-carname.

* Massenoperation mit SELECT
  SELECT carrid FROM zscarreval INTO Id_carrid
  WHERE eval = id_eval.
* Hilfsstruktur befüllen
  ls_fg_eval-eval = id_eval.
  ls_fg_eval-carrid = Id_carrid.
* Lesen der fg von der db tabelle scarr
  PERFORM read_carname
  USING
    Id_carrid
  CHANGING
    Id_carname.

  ls_fg_eval-carname = Id_carname.
* Eintrag in der internen Tabelle erzeugen
  APPEND ls_fg_eval TO et_fg_eval.
  ENDSELECT.

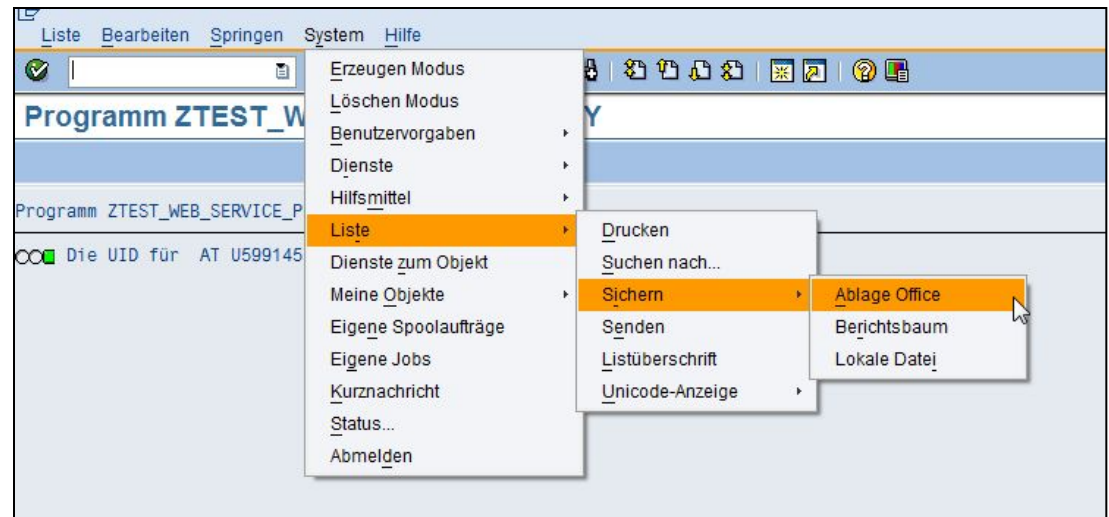
ENDFORM.          " READ_FG_EVAL
```

ABAP Liste

ABAP Liste

Standard Funktionen

- Suche
- Blättern
- Exportieren
- ...

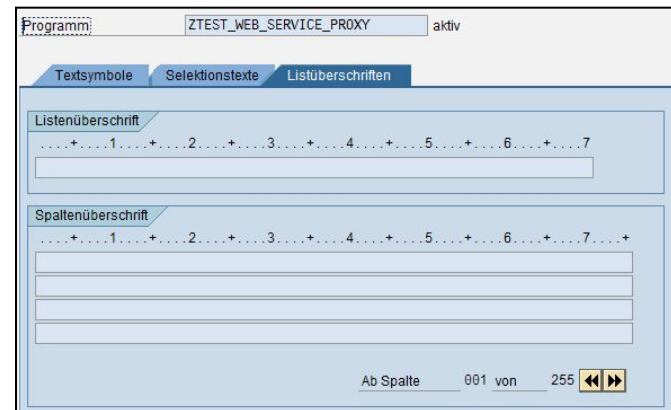
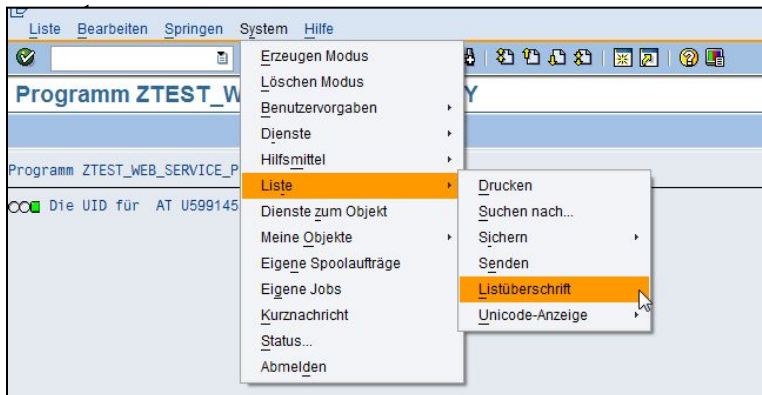


ABAP Liste



Listen und Spaltenüberschriften

- Direkte Pflege
System->Liste->Listenüberschrift
- Über
Springen->Textelemente->Listenüberschrift



ABAP Liste

Ereignisse

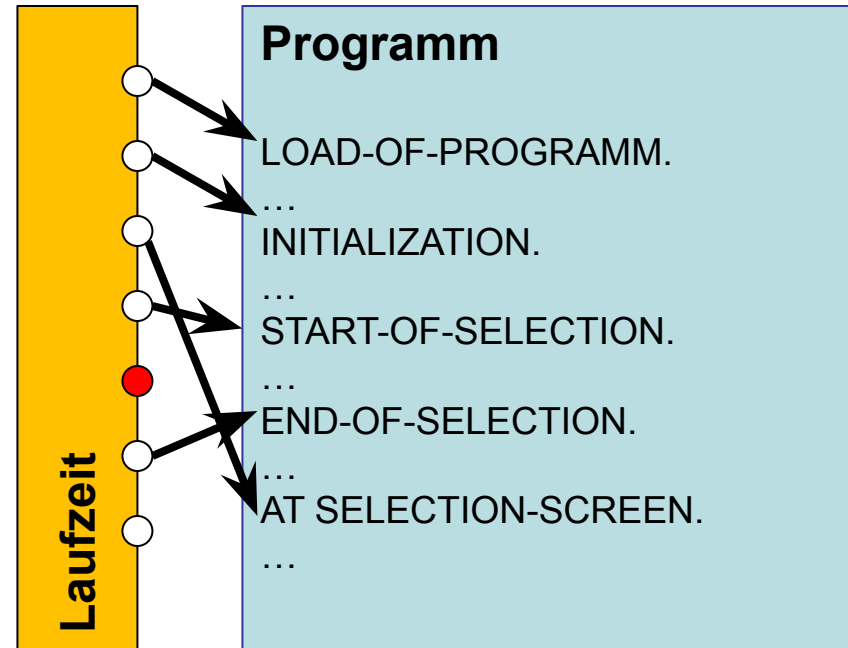
- START-OF-SELECTION
- END-OF-SELECTION
- TOP-OF-PAGE
- ...

```
TYPE-POOLS: abap,  
            icon.
```

```
START-OF-SELECTION.
```

* Gültigkeit ausgeben

```
IF ls_response-valid = abap_true.  
    WRITE: / icon_green_light AS ICON, 'Die UID für ', pa_ms, pa_vat, ' ist gültig!'.  
ELSE.  
    WRITE: / icon_red_light AS ICON, 'Die UID für ', pa_ms, pa_vat, ' ist ungültig!'.  
ENDIF.
```



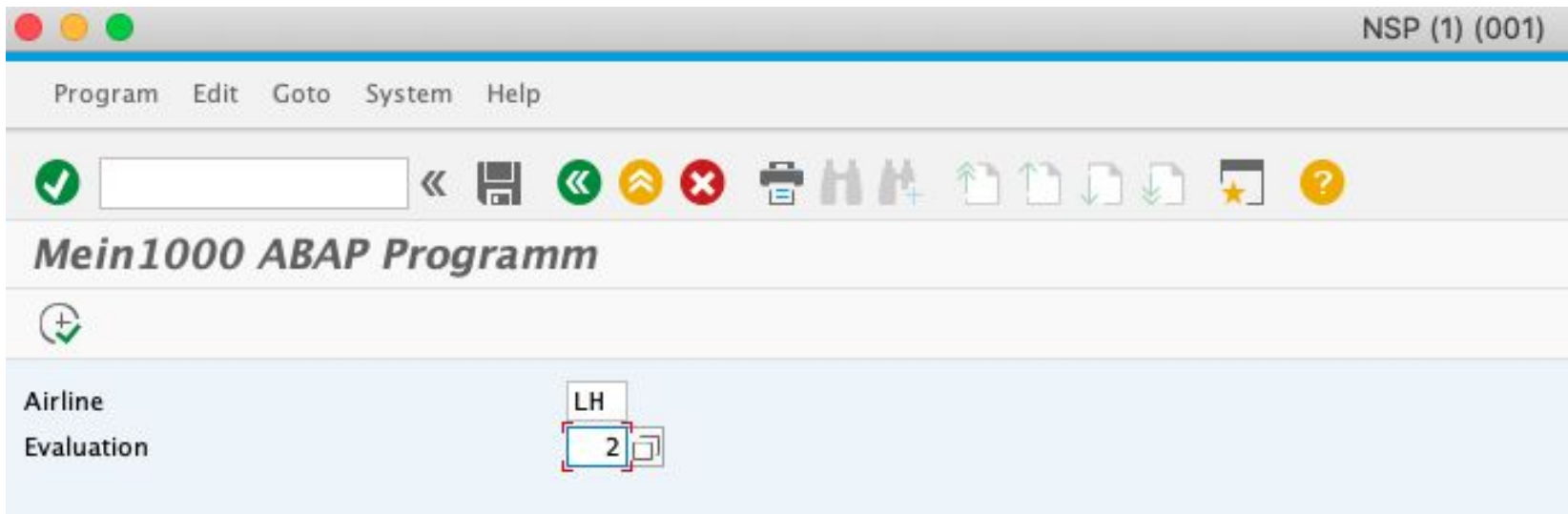
ABAP Selektionsbild

ABAP Selektionsbild



Motivation

- Verwendung

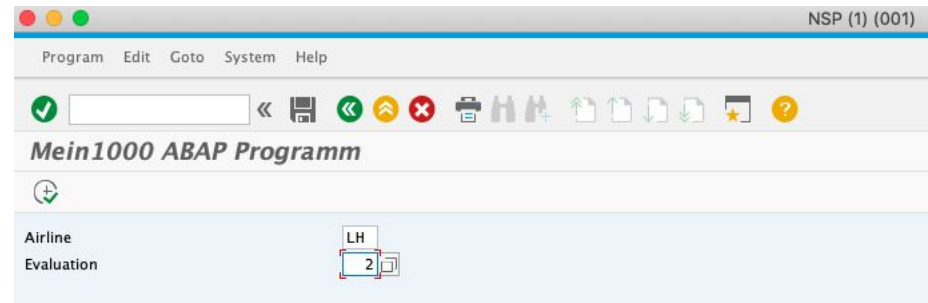


ABAP Selektionsbild



Gestaltung

- Parameters
 - UI Element
 - Datenfeld



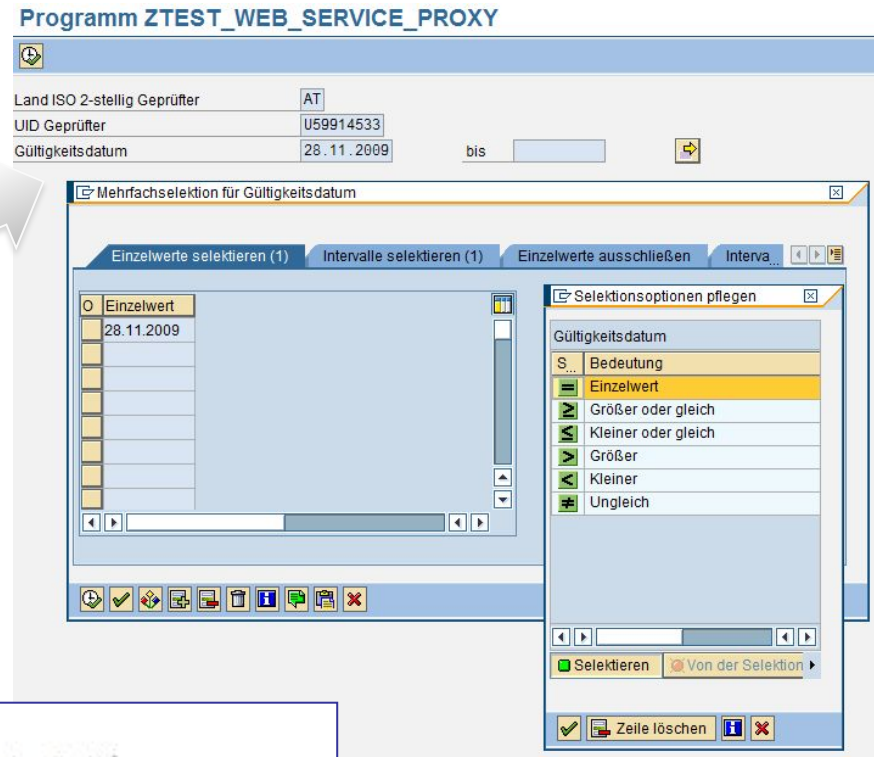
```
* Eingabefeld für Fluggesellschaft  
PARAMETERS: pa_car TYPE scarr-carrid,  
             pa_eval TYPE zscarreval-eval.
```


ABAP Selektionsbild



Gestaltung

- Select-Options
 - UI Element
 - Datenfeld



```
* data declarations
DATA: ld_date TYPE dats. "Datum für Select Option

|SELECT-OPTIONS: so_date FOR ld_date. "Datumsselektion
```

Select Option und Parameter

1. Range deklarieren

* Eingabe für Intervalle

```
SELECT-OPTIONS: so_eval FOR gd_so_eval.
```

* Nur die Daten der S0

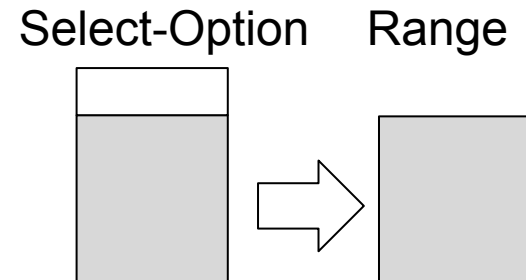
```
data: gt_so_eval like range of gd_so_eval.
```

2. Daten von SO in Range kopieren

* Ermitteln der FG zu einer Eval

```
gt_so_eval = so_eval[[]].
```

```
PERFORM read_fg_eval USING gt_so_eval "pa_eval  
CHANGING gt_fg_eval.
```



3. An Unterprogramm übergeben

```
FORM read_fg_eval USING it_so_eval like gt_so_eval "id_eval" TYPE zscarreval=eval  
CHANGING et_fg_eval TYPE gtt_fg_eval.
```

```
DATA: ld_carrid TYPE scarr-carrid,  
ls_fg_eval LIKE LINE OF et_fg_eval,  
ld_carrname TYPE scarr-carrname.
```

* Hilfsstruktur zum Auslesen der Eval Daten

```
DATA: ls_scarreval type zscarreval.
```

* Massenoperation mit SELECT

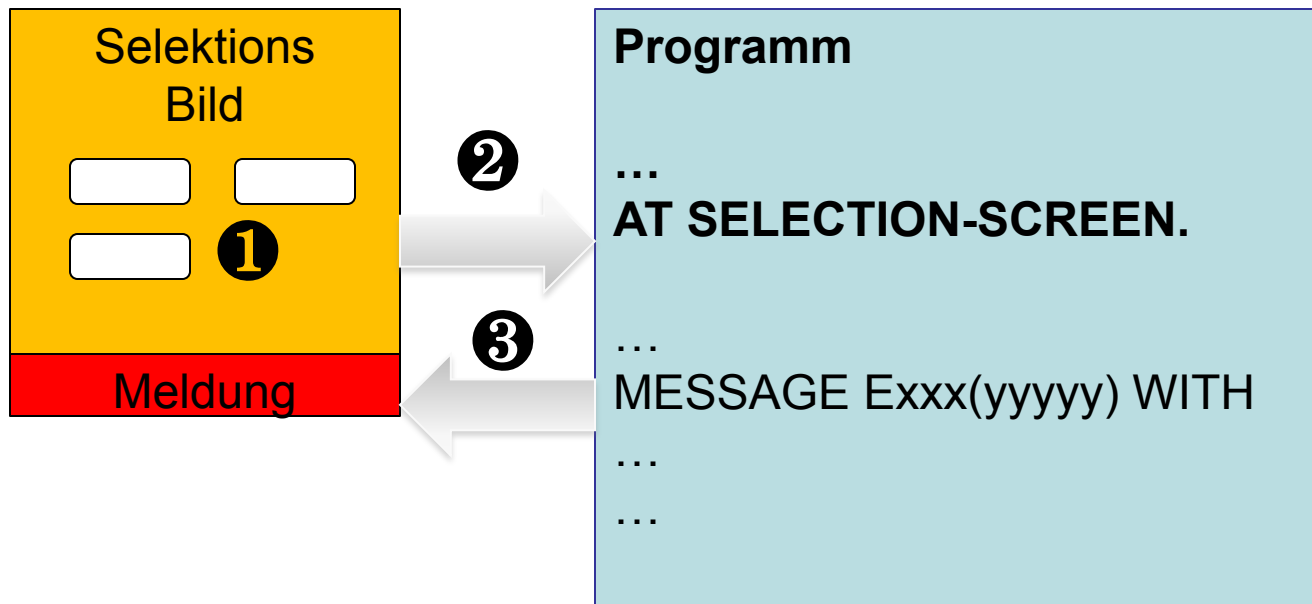
* Auslesen aller Spalten einer Zeile aus eval

```
SELECT * FROM zscarreval INTO ls_scarreval  
WHERE eval in it_so_eval. "=" id_eval.
```

ABAP Selektionsbild

Ereignisse

- AT SELECTION-SCREEN.



Modularisierung

ABAP Prozedural

Interne Modularisierung



Includes

```
13: INCLUDE /      /co_test_import_writf01.  
14:
```

- Objekte können in **Includes** zusammengefasst werden
- Includes werden beim **Übersetzungsvorgang** eingebunden
- Includes sind wiederverwendbare „Textkonserven“
- Eine Wiederverwendung durch Unterprogramme und Deklarationen in Includes ist zu vermeiden!

ABAP Prozedural

Externe Modularisierung



Funktionsgruppen und Funktionsbausteine

- Definition TA SE37 (Function Builder)
 - Schnittstelle
 - IMPORTING
 - EXPORTING
 - CHANGING
 - EXCEPTIONS (prozedural/OO)
- Verwenden (Über Drucktaste Muster in Drucktastenleiste)
 - CALL FUNCTION '<name>'

The screenshot shows the 'Function Builder: BAPI_SFLIGHT_GETLIST anzeigen' window. The 'Quelltext' tab is active, displaying the source code for the function. The code is as follows:

```
1 FUNCTION BAPI_SFLIGHT_GETLIST.  
2 ***  
3 ***"Lokale Schnittstelle:  
4 *** IMPORTING  
5 ***   VALUE (FROMCOUNTRYKEY) LIKE BAPISFDETA-COUNTRYFR  
6 ***   VALUE (FROMCITY) LIKE BAPISFDETA-CITYFROM  
7 ***   VALUE (TOCOUNTRYKEY) LIKE BAPISFDETA-COUNTRYTO  
8 ***   VALUE (TOCITY) LIKE BAPISFDETA-CITYTO  
9 ***   VALUE (AIRLINECARRIER) LIKE BAPISFDETA-CARRID DEFAULT SPACE  
10 ***   VALUE (AFTERNOON) LIKE BAPI_AUX-AFTERNOON DEFAULT SPACE  
11 ***   VALUE (MAXREAD) LIKE BAPI_AUX-MAXREAD DEFAULT 0  
12 *** EXPORTING  
13 ***   VALUE (RETURN) LIKE BAPIRET2 STRUCTURE BAPIRET2  
14 *** TABLES  
15 ***   FLIGHTLIST STRUCTURE BAPISFLIST  
16 ***  
17  
18 CLEAR RETURN
```

FuBa 1



Create Function Module

Function Module: Z8820038_WS19_FG_R_carrname

Function group: Z8820038_WS19_FG

Short text: Read carrname

```
* Lesen der FG von der DB Tabelle SCARR
* SELECT SINGLE carrname FROM scarr INTO gd_carrname
* WHERE carrid = pa_car.
* Aufruf Unterprogramm für die Ermittlung des Carrname
* Verwendungsstelle
* PERFORM read_carrname USING pa_car
*                               CHANGING gd_carrname.

CALL FUNCTION 'Z8820038_WS19_FG_R_CARRNAME'
  EXPORTING
    id_carrid = pa_car
  IMPORTING
    ed_carrname = gd_carrname.
```

Save

Function modules Edit Goto Utilities System Help

Test Function Module: Result Screen

Test for function group: Z8820038_WS19_FG
Function module: Z8820038_WS19_FG_R_CARRNAME
Uppercase/Lowercase:

Runtime: 954 Microseconds

Import parameters	Value
ID_CARRID	LH

Export parameters	Value
ED_CARRNAME	Lufthansa

FuBa 2



```
FUNCTION z8820038_ws19_fg_r_eval.
```

```
*"-----
```

```
***"Local Interface:
```

```
*" IMPORTING
```

```
*" REFERENCE(ID_CARRID) TYPE SCARR-CARRID
```

```
*" EXPORTING
```

```
*" REFERENCE(ED_EVAL) TYPE ZSCARREVAL-EVAL
```

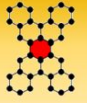
```
*"-----
```

```
SELECT SINGLE eval FROM zscarreval INTO ed_eval  
WHERE carrid = id_carrid.
```

```
ENDFUNCTION.
```


ABAP Objektorientiert

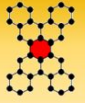
ABAP Objektorientiert



Motivation

- Wofür?
- Prozedural vs. Objektorientiert
- Eigenschaften der OO
- Abdeckungsgrad in ABAP OO
 - Vererbung
 - Polymorphie
 - Kapselung
- Separate Definition und Implementierung

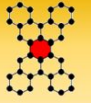
ABAP Objektorientiert



Modellierung mit der Unified Modeling Language (UML), im Speziellen

- Klassendiagramm
 - Klasse
 - Attribut
 - Methode
 - Assoziation
 - Kardinalität
- Sequenzdiagramm
- Deploymentdiagramme

ABAP Objektorientiert



Klasse und Objekt

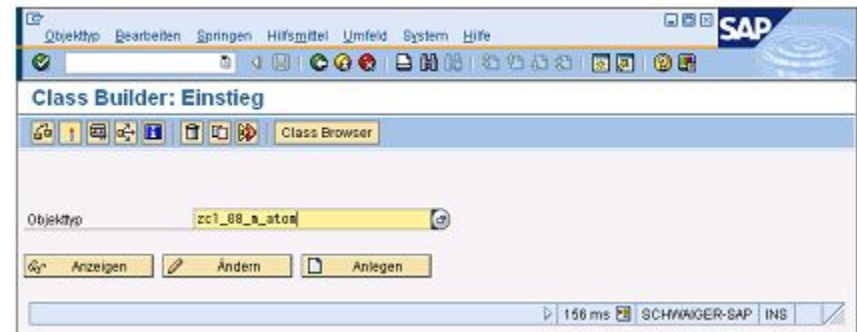
- Dynamisch vs. Statisch
- Sichtbarkeiten
- Attribut
- Methode
- Ereignis
- Bauplan/Blaupause

ABAP Klasse



Arten von ABAP Klassen

- **“Normale” Klasse**
- **Ausnahmeklasse**
- Testklasse
- Persistentes Objekt



Eigenschaften von ABAP Klassen

- Abstrakt
- Final



Klasse SE24

The screenshot shows the 'Create Class' dialog box in SAP. The title bar reads 'Create Class'. There are three colored window control buttons (red, yellow, green) on the left. The main area contains the following fields:

- Class:** A text field containing 'ZCL_8820038_WS19_FG' and a small icon to its right.
- Description:** A text field containing 'Carrier'.
- Instantiation:** A dropdown menu showing 'Public'.

Below these fields is a section titled 'Class Type' with a blue underline. It contains five radio button options:

- Usual ABAP Class
- Exception Class
 - With Message Class
- Persistent class
- Test Class (ABAP Unit)

At the bottom left of the dialog is a checkbox labeled 'Final' with a red dashed border around it. At the bottom right are two buttons: a green checkmark icon followed by the text 'Save', and a red 'X' icon.

ABAP Klasse



Attribut

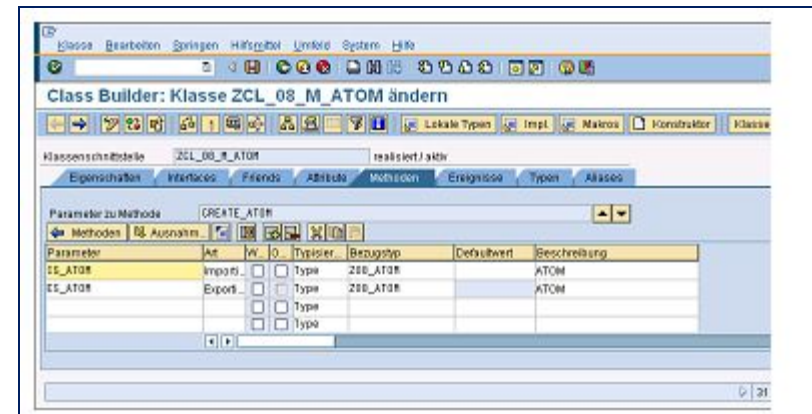
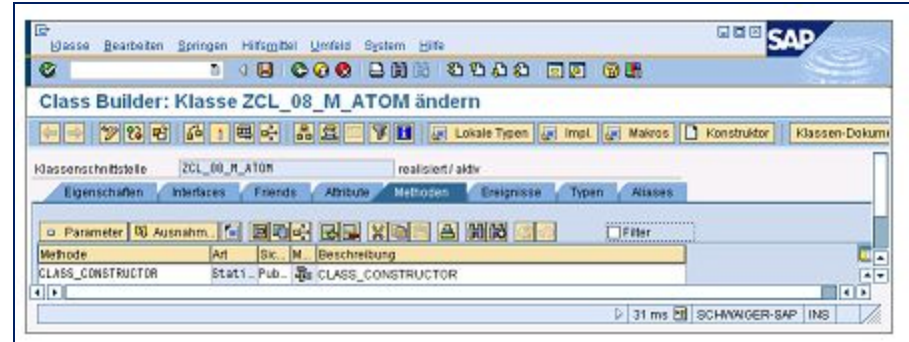
- Statisch
- Instanz
- Sichtbarkeiten



ABAP Klasse

Definition Methode

- Statisch
- Instanz
- Sichtbarkeiten



```

* Daten von der Datenbankstabelle lesen
SELECT * FROM z08_atom INTO TABLE et_atoms
WHERE pk_atom IN it_r_pk_atoms.

IF sy-subrc <> 0.
* Falls keine Daten gefunden wurden, dann Exception
RAISE EXCEPTION TYPE zcx_08_atom
EXPORTING
  textid = zcx_08_atom->atom_no_data
ENDIF.
    
```

Art	Parameter	Typisierung	Beschreibung
IT	IT_R_PK_ATOMS	TYPE Z08_TT_R_PK_ATOM	TT zu Range PK_ATOM
ET	ET_ATOMS	TYPE Z08_TT_ATOM	TT Atom
EX	ZCX_08_ATOM		Exc Atom

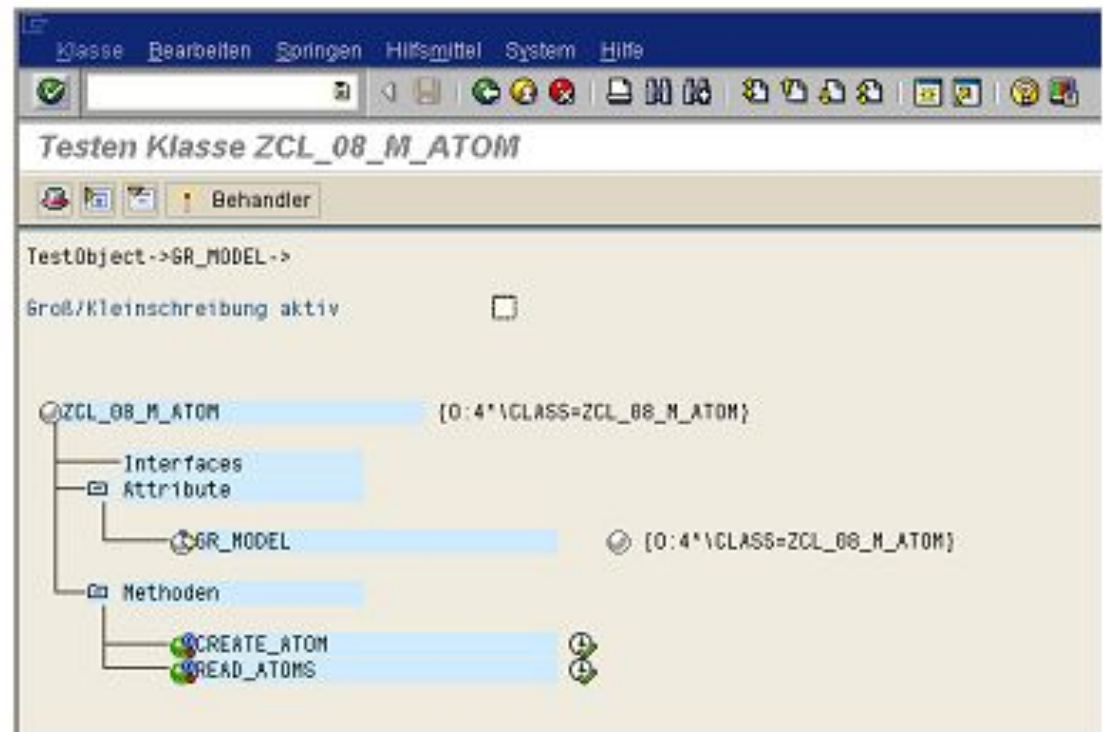
Methode: READ_ATOMS

ABAP Klasse



Testen von Klassen

- Testrahmen
- Testklassen



ABAP Objekt

Zustände von Objekten

- Erzeugen (Constructor)

```
IKT.  
* Objekt instanziiieren  
  CREATE OBJECT go_fg  
  EXPORTING  
    id_carrid = pa_car.  
* Fehlerbehandlung
```

- Terminieren (Garbage Collector)
 - Mittels CLEAR kann die Objektreferenz initialisiert werden und dadurch sammelt der GC das Objekt ein und gibt die Ressourcen frei.
 - Prinzipiell, wenn ein Objekt nicht mehr erreichbar ist, dann gibt der GC die betroffenen Ressourcen frei

ABAP Objekt



Aufruf von Methoden

- Statisch

- Instanz

`go_alv->display().`

- Funktional

`gd_carrname_single = go_fg->get_carrname().`

Verwendung von Attributen

```
cl_salv_table=>factory(  
* EXPORTING  
* list_display = IF_SALV_C_BOOL_SAP=>FALSE  
* r_container =  
* container_name =  
  IMPORTING  
    r_salv_table = go_alv  
  CHANGING  
    t_table = gt_fg_eval  
  ).
```

ABAP OO Ausnahmen

ABAP Ausnahmen (Exceptions)

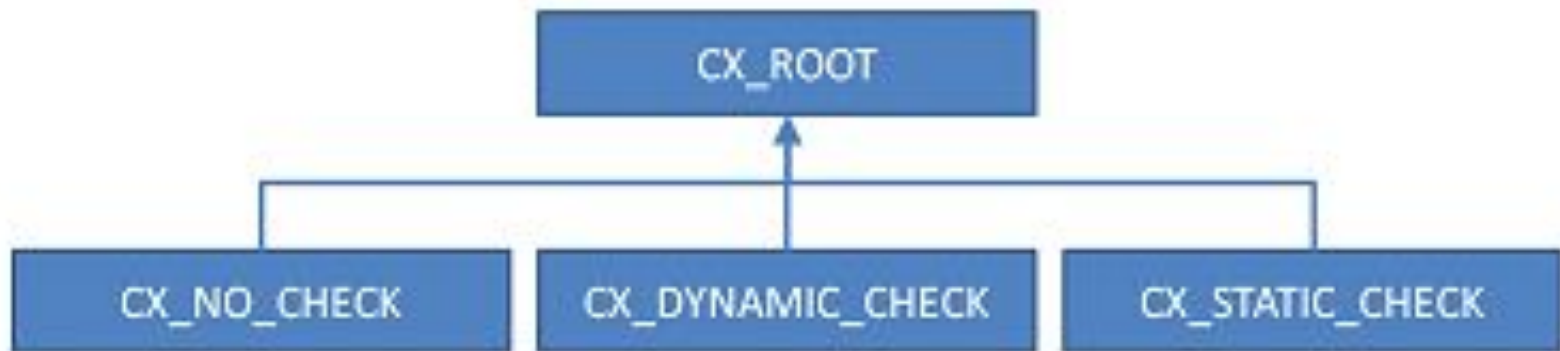


Existierende Ausnahmemechanismen

- Auswertung sy-subrc
- CATCH SYSTEM-EXCEPTION ...
.ENDCATCH. + sy-subrc Auswertung
- TRY. CATCH. ENDTRY.

ABAP Ausnahmen (Exceptions)

Vererbungshierarchie für Exceptions CX_ROOT



ABAP Ausnahmen (Exceptions)

Exception anlegen
Text zur Ausnahme

The screenshot shows the 'Anlegen Klasse' (Create Class) dialog box in SAP. The title bar reads 'Anlegen Klasse ZCX_08_ATOM'. The dialog has several fields: 'Klasse' (Class) is 'ZCX_08_ATOM', 'Erbt von' (Inherits from) is 'CX_STATIC_CHECK', 'Beschreibung' (Description) is 'Exc Atom', and 'Inst-Erzeugung' (Instance Generation) is 'Public'. Below these fields is a 'Klassentyp' (Class Type) section with radio buttons for: 'Gewöhnliche ABAP-Klasse' (unselected), 'Ausnahmeklasse' (selected), 'mit Nachrichtenklasse' (checkbox), 'Persistente Klasse' (unselected), and 'Testklasse (ABAP Unit)' (unselected). At the bottom, there are checkboxes for 'Final' (checked) and 'Nur Modelliert' (unchecked). The bottom bar contains 'Sichern' (Save) and a red 'X' button.

ABAP Ausnahmen (Exceptions)

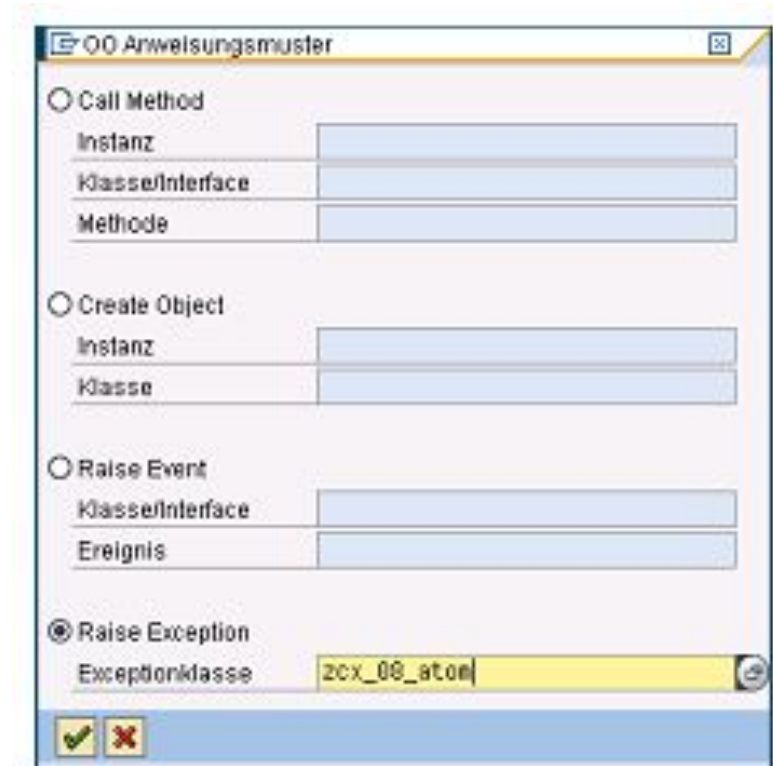


Exception auslösen

RAISE EXCEPTION

Ty.	Parameter	Type spec.	Description
	ID_CARRID	TYPE SCARR-CARRID	Airline Code
	ZCX_8820038_WS19_FG		Exception FG


```
METHOD constructor.  
  
me->gd_carrid = id_carrid.  
  
CALL FUNCTION 'Z8820038_WS19_FG_R_CARRNAME'  
  EXPORTING  
    id_carrid = me->gd_carrid  
  IMPORTING  
    ed_carrname = me->gd_carrname.  
  
IF me->gd_carrname IS INITIAL.  
  RAISE EXCEPTION TYPE zcx_8820038_ws19_fg  
* EXPORTING  
*   textid =  
*   previous =  
*  
ENDIF.
```



ABAP Ausnahmen (Exceptions)

Ausnahmetexte



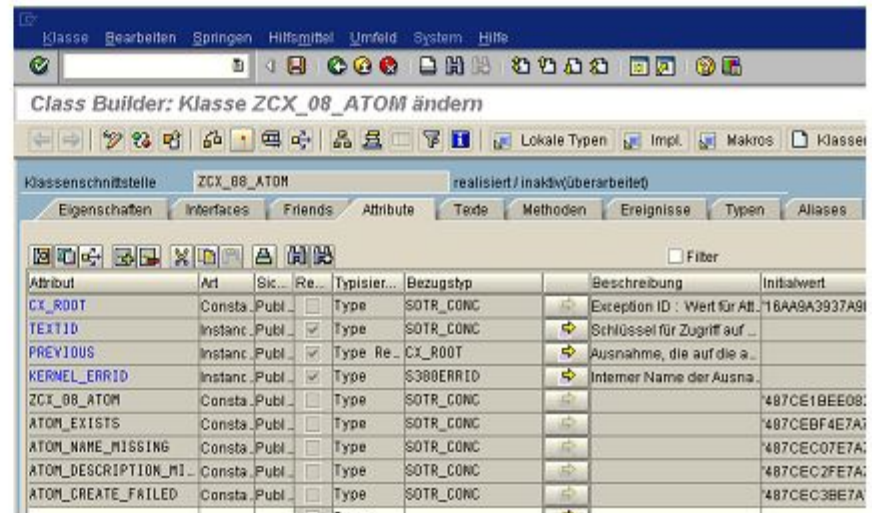
Class Builder: Klasse ZCX_08_ATOM ändern

Klassenschnittstelle ZCX_08_ATOM realisiert / inaktiv(überarbeitet)

Eigenschaften Interfaces Friends Attribute Texte Methoden Ereignisse

Langtext

Exception ID	Text
CX_ROOT	Es ist eine Ausnahme aufgetreten.
ZCX_08_ATOM	Es ist ein Problem im Zusammenhang mit dem Atom aufgetreten
ATOM_EXISTS	Das Atom wurde bereits angelegt
ATOM_NAME_MISSING	Der Name des Atoms ist nicht vorhanden
ATOM_DESCRIPTION_MISSING	Die Beschreibung des Atoms ist nicht vorhanden
ATOM_CREATE_FAILED	Die Anlage des Atoms ist fehlgeschlagen



Class Builder: Klasse ZCX_08_ATOM ändern

Klassenschnittstelle ZCX_08_ATOM realisiert / inaktiv(überarbeitet)

Eigenschaften Interfaces Friends Attribute Texte Methoden Ereignisse Typen Aliases

Filter

Attribut	Art	Stc...	Re...	Typisier...	Bezugsstyp	Beschreibung	Initialwert
CX_ROOT	Consta	Publ	<input type="checkbox"/>	Type	SOTR_CONC	Exception ID : Wert für Ab...	16A9A3937A9
TEXTID	Instanc	Publ	<input checked="" type="checkbox"/>	Type	SOTR_CONC	Schlüssel für Zugriff auf ...	
PREVIOUS	Instanc	Publ	<input checked="" type="checkbox"/>	Type Re...	CX_ROOT	Ausnahme, die auf die a...	
KERNEL_ERRID	Instanc	Publ	<input checked="" type="checkbox"/>	Type	S380ERRID	Interner Name der Ausna...	
ZCX_08_ATOM	Consta	Publ	<input type="checkbox"/>	Type	SOTR_CONC		487CE1BEE08
ATOM_EXISTS	Consta	Publ	<input type="checkbox"/>	Type	SOTR_CONC		487CEBF4E7A
ATOM_NAME_MISSING	Consta	Publ	<input type="checkbox"/>	Type	SOTR_CONC		487CEC07E7A
ATOM_DESCRIPTION_MI...	Consta	Publ	<input type="checkbox"/>	Type	SOTR_CONC		487CEC2FE7A
ATOM_CREATE_FAILED	Consta	Publ	<input type="checkbox"/>	Type	SOTR_CONC		487CEC3BE7A

V1

ABAP OO Anwendung ALV

Vorbereitung Struktur und Tabellentyp



NSP (2) (001)

Structure Edit Goto Utilities Extras Environment System Help

Dictionary: Change Structure

Structure: ZST_8820038_WS19_FG_EVAL Active
Short Description: Eval for FG

Attributes Components Entry help/check Currency/quantity fields

Component	Key	Initi.	Typing Method	Component Type	Data Type	Length	D
CARRID	<input type="checkbox"/>	<input type="checkbox"/>	Types	S_CARR_ID	CHAR	3	
CARRNAME	<input type="checkbox"/>	<input type="checkbox"/>	Types	S_CARRNAME	CHAR	20	
EVAL	<input type="checkbox"/>	<input type="checkbox"/>	Types	Z8820038_EVAL	INT1	3	

NSP (2) (001)

Table Type Edit Goto Utilities Environment System Help

Dictionary: Change Table Type

Table Type: ZTT_8820038_WS19_FG_EVAL Active
Short text: Eval FG

Attributes Line Type Initialization and Access Primary Key Secondary Key

Line Type: ZST_8820038_WS19_FG_EVAL

Predefined Type
Data Type:
No. of Characters: Decimal Places:

Reference type
 Name of Ref. Type:
 Reference to Predefined Type
Data Type:
Length: Decimal Places:

V1: ABAP Klasse CL_SALV_TABLE



GUI Klassen - Lebenszyklus

1. Deklaration Objektreferenz für ALV
2. Instanziierung mittels FACTORY
3. Anzeige mittels DISPLAY

```
** 00 Ausgabe
DATA: go_alv TYPE REF TO cl_salv_table.

TRY.
    cl_salv_table=>factory(
*   EXPORTING
*     list_display   = IF_SALV_C_BOOL_SAP=>FALSE
*     r_container    =
*     container_name =
    IMPORTING
        r_salv_table = go_alv
    CHANGING
        t_table      = gt_fg_eval
    ).

    go_alv->display( ).

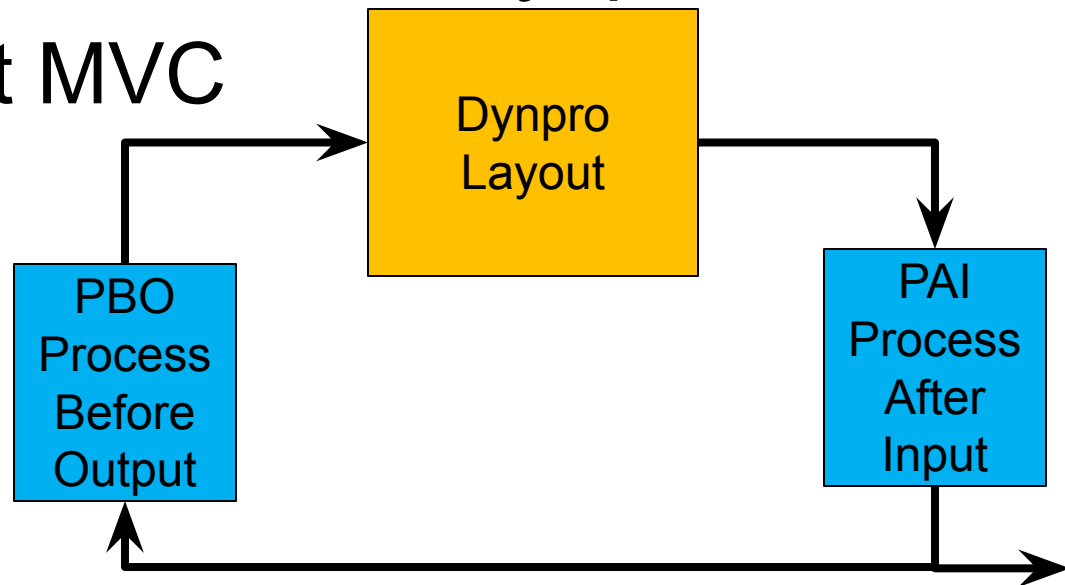
CATCH cx_salv_msg .
ENDTRY.
```

ABAP Dynpro

ABAP Dynpro

Motivation

- Ablauf PBO, PAI, Layout
- Weitere Elemente eines Dynpros
- Vergleich mit MVC



ABAP Dynpro



Dynpro erstellen

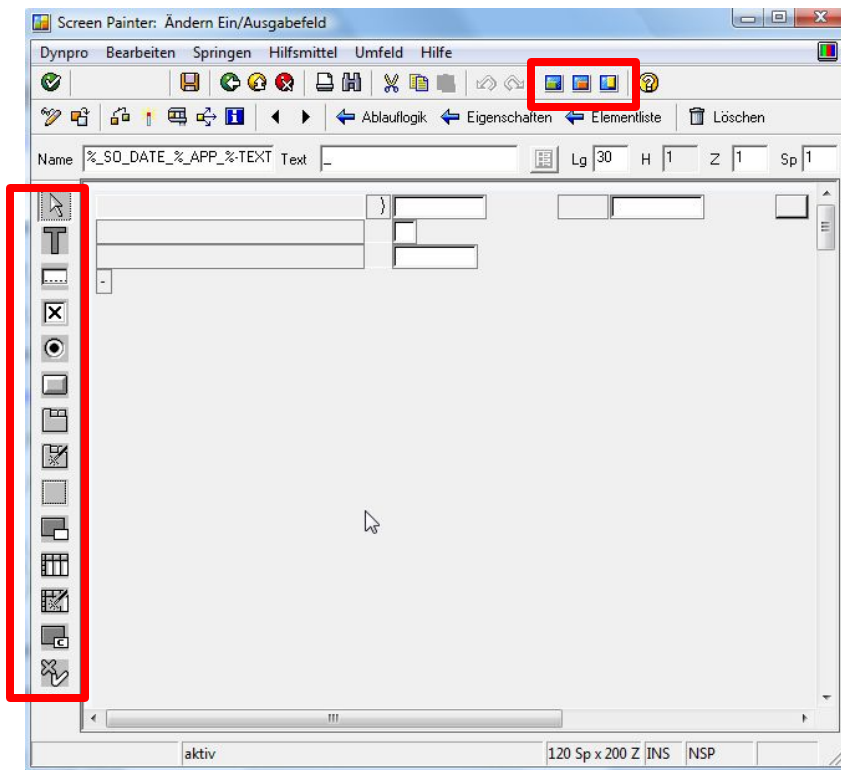
Zum Zeitpunkt END-OF-SELECTION den
Aufruf

```
CALL SCREEN 9000.
```

einfügen und mittels Vorwärtsnavigation
anlegen.

ABAP Dynpro

Grafischen Layout Editor



ABAP Dynpro



- Eine Drucktaste definieren, um wieder zur Aufrufstelle navigieren zu können
 - Funktionscode `BACK` zuordnen
- OK Feld definieren;
 - In den Eigenschaften des Dynpros
 - Im rufenden Programm

```
DATA: ok_code TYPE sy-ucomm.
```

ABAP Dynpro



In der Dynpro Ablauflogik

- Ein Process After Input (PAI) Modul zur Behandlung des Drucktastenfunktionscodes BACK
- Ein Process Before Output (PBO) Modul zum Setzen des Titels einfügen

- Kommunikationsstruktur im ABAP Dictionary definieren bzw. einen vorhandenen `<ABAP Dictionary Strukturtyp>` verwenden
- Layout Elemente basierend auf dem `<ABAP Dictionary Strukturtyp>` im Screen Painter definieren
- Datenstruktur
TABLES: `<ABAP Dict Strukturtyp>`.
Im Programm definieren

ABAP Dynpro



Dynpro erstellen

- Weitere diverse PBO/PAI Module
 - Logik
 - Initialisierung
 - Prüfung Benutzereingaben

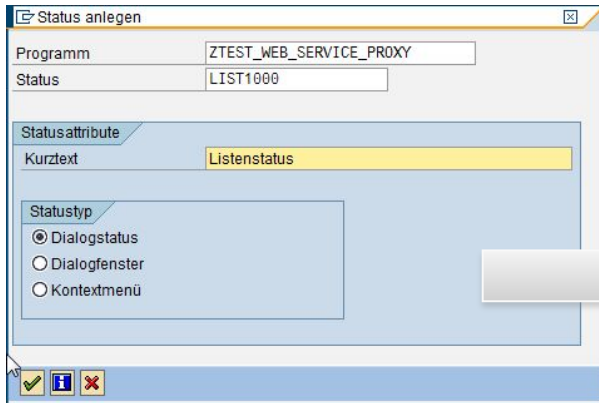
ABAP Dynpro



- Zum Setzen der Oberfläche, d.h. Menü, Drucktastenleiste, Symbolleiste, Funktionstastenzuordnung, wird der Menu Painter verwendet.
- Für unseren Anwendungsfall reicht es, die Drucktaste BACK zu aktivieren

ABAP Dynpro

Menu Painter



Oberfläche ZTEST_WEB_SERVICE_PROXY aktiv(überarbeitet)

Menüleiste Zeilenauswahlliste

Normen einmischen

Liste Bearbeiten Springen

Code Text

PRI	Drucken
	Sichern/Senden
%EX	Beenden
SAVE	Sichern

MYFUNCTION Meine Funktion

Drucktastenleiste

Positionen 1 - 7	PICK	%PRI	%SC	%SC+	RW		
Positionen 8 - 14	MYFUNCTION						

Funktionstasten Zeilenauswahlliste

Symbolleiste

SAVE	BACK	%EX	RW	%PRI	%SC	%SC+	P--	P-	P+
------	------	-----	----	------	-----	------	-----	----	----

Empfohlene Funktionstastenbelegungen

F2	PICK	Auswählen
F9	<.>	Markieren
Umsch-F2	<.>	Löschen
Umsch-F4	<.>	Sich.o.Prüfen
Umsch-F5	<.>	Anderes <Objekt>

Frei belegbare Funktionstasten

F5	MYFUNCTION	Meine Funktion
----	------------	----------------

V2

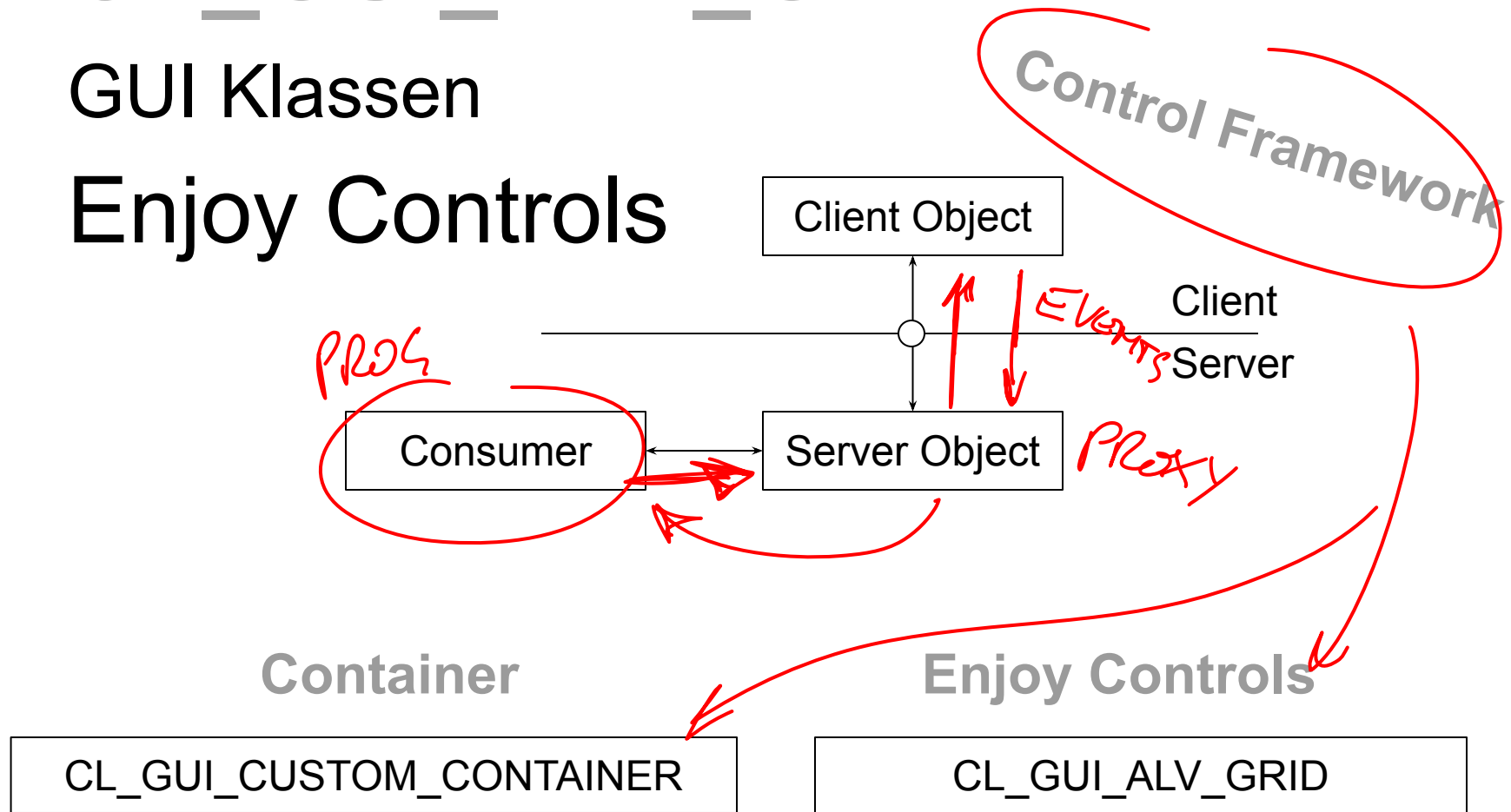
ABAP OO Anwendung ALV

V2: ABAP Klasse CL_GUI_ALV_GRID



GUI Klassen

Enjoy Controls



GUI Klassen - Lebenszyklus

1. Deklaration Objektreferenzen für Container und Enjoy Control (“ALV”)
2. Instanziierung Container und Enjoy Control
3. Verwendung
4. Abbau der Client und Server Objekte

ABAP Klasse

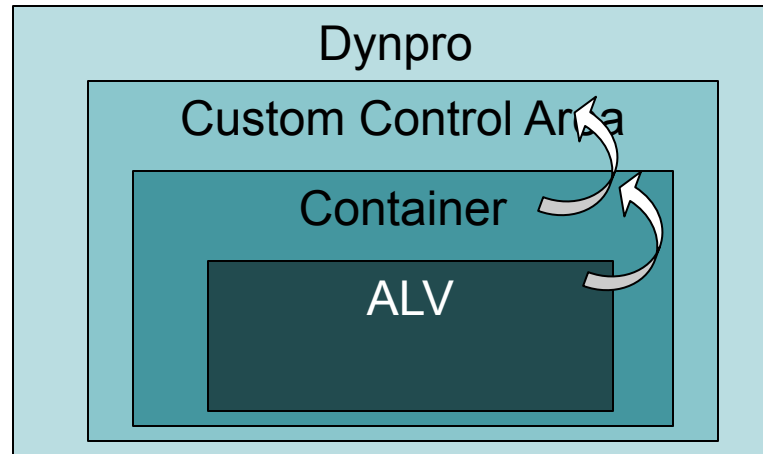
GUI Klassen - Lebenszyklus

1

DATA: gr_cont TYPE REF TO CL_GUI_CUSTOM_CONTAINER,
gr_alv TYPE REF TO CL_GUI_ALV_GRID.

2

CREATE OBJECT: gr_cont ...,
gr_alv ...



3

CALL METHOD gr_alv->set_table_for_first_display ...
SET HANDLER <handler> FOR gr_alv.

4

Gr_alv->free(). "Client
Gr_cont->free(). "Client
Clear gr_alv. "Server
Clear gr_cont. "Server

ABAP OO Events

ABAP Klasse



Ereignisse dienen zur losen Kopplung zwischen Objekten.

- Ein Objekte, der **Sender**, schickt Nachrichten.
- Ein/mehrere Objekte, der/die **Empfänger**, empfängt/empfangen die Nachricht

ABAP Objekt



Sender

1. Event Definition: `EVENTS`
2. Event auslösen: `RAISE EVENT EXPORTING <parameter> = <wert>`

Empfänger

3. Event Handler: `ON_ <event> FOR EVENT <event> OF <sender> IMPORTING sender <parameter>`
4. Behandler registrieren: `SET HANDLER <empfänger>->ON_ <event> FOR <sender>.`

E-Behandler



* Ereignisbehandler (lokale Klasse)

* CLASS lsl_event_handler IMPLEMENTATION

*

CLASS lcl_event_handler DEFINITION.

PUBLIC SECTION.

CLASS-METHODS: on_double_click FOR EVENT double_click
OF cl_gui_alv_grid
IMPORTING e_row.

PROTECTED SECTION.

PRIVATE SECTION.

ENDCLASS. "lcl_event_handler DEFINITION

* CLASS lsl_event_handler IMPLEMENTATION

*

CLASS lcl_event_handler IMPLEMENTATION.

METHOD on_double_click.

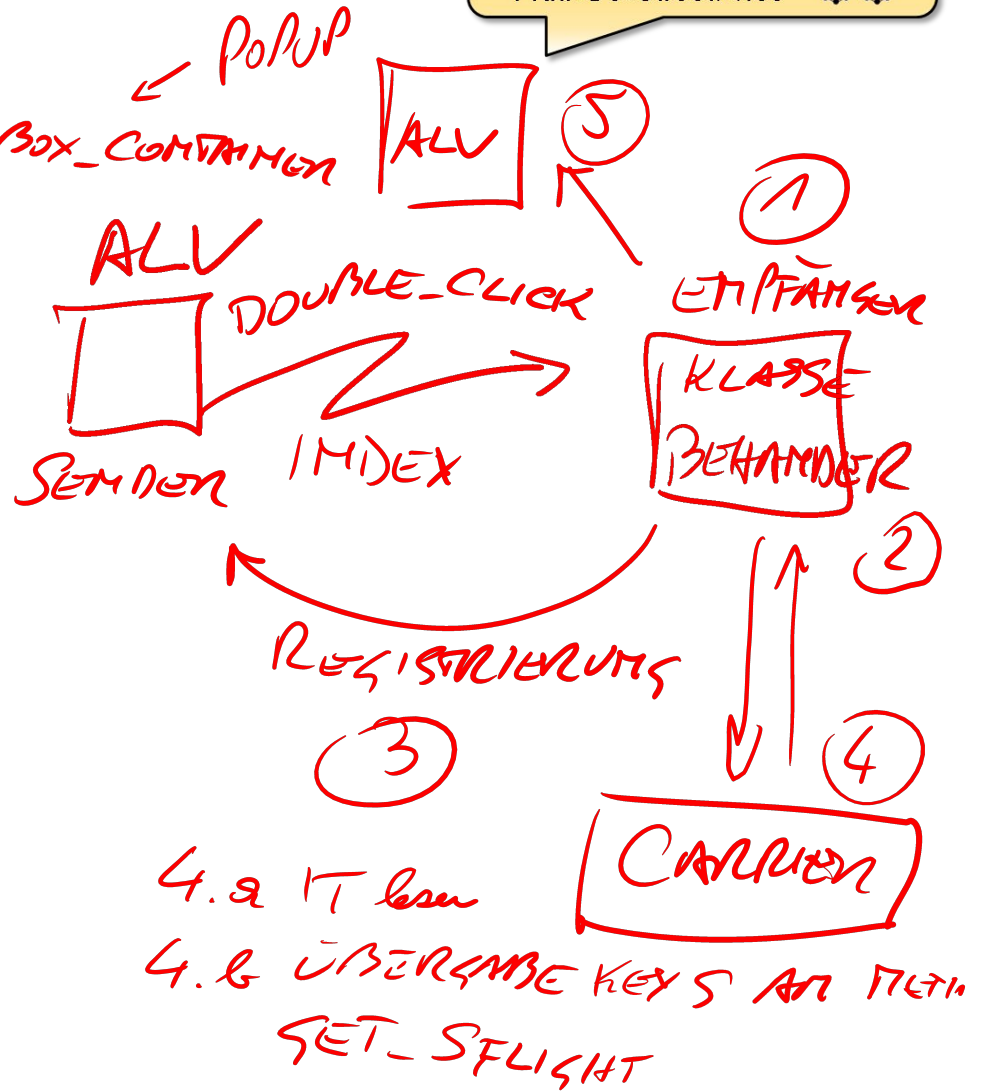
ENDMETHOD. "on_double_click

ENDCLASS. "lsl_event_handler IMPLEMENTATION

ABAP Objekt

Beispiel Doppelklick

- Sender
- Empfänger
- De-/Registrierung
- Behandlung



Ergebnis

NSP (1) (001)

System Help

SAP

Airline: LH
Airline: Lufthansa
Zurück

ID	No.	Flight Date	Airfare	Curr.	Plane Type	Capacity	Occupied	Bookin
LH	400	25.07.2015	666,00	EUR	A310-300	280	271	212.7
LH	400	22.0					271	209.1
LH	400	19.0					271	209.6
LH	400	17.1					271	208.9
LH	400	14.1					271	209.2
LH	400	12.1					265	208.0
LH	400	28.1					271	210.5
LH	400	09.0.2016	666,00	EUR	A310-300	280	265	205.4
LH	400	28.01.2016	666,00	EUR	A310-300	280	271	211.8
LH	400	06.02.2016	666,00	EUR	A310-300	280	271	212.0
LH	400	05.03.2016	666,00	EUR	A310-300	280	261	202.6
LH	400	02.04.2016	666,00	EUR	A310-300	280	65	50.4
LH	400	30.04.2016	666,00	EUR	A310-300	280	70	55.5
LH	400	28.05.2016	666,00	EUR	A310-300	280	16	13.2
LH	400	25.06.2016	666,00	EUR	A310-300	280	89	66.0
LH	400	23.07.2016	666,00	EUR	A310-300	280	51	38.0

Information: [0000000004]

SAP

Oberfläche



NSP (1) (001)

Enterprise Computing System Help

Liste der Flüge

Detailanzeige

User Interface Z8820038_WS19_DYNPRO Active

Menu Bar Status Screen 9000

+ Display Standards

Enterprise Computing

Code Text

WS19 Details WS19

Application Toolbar Status Screen 9000

Items 1 - 7 WS19 Detail

Items 8 - 14

Function Keys Status Screen 9000

Standard Toolbar

Recommended Function Key Settings

Freely Assigned Function Keys

No.	Flight Date							
00	25.07.2015							
00	22.08.2015							
00	19.09.2015							
00	17.10.2015							
00	14.11.2015							
00	12.12.2015							
00	28.12.2015	666,00	EUR	A310-300	280	271	210.5	
00	09.01.2016	666,00	EUR	A310-300	280	265	205.4	
00	28.01.2016	666,00	EUR	A310-300	280	271	211.8	
00	06.02.2016	666,00	EUR	A310-300	280	271	212.0	
LH	400 05.03.2016	666,00	EUR	A310-300	280	261	202.6	
LH	400 02.04.2016	666,00	EUR	A310-300	280	65	50.4	
LH	400 30.04.2016	666,00	EUR	A310-300	280	70	55.5	
LH	400 28.05.2016	666,00	EUR	A310-300	280	16	13.2	
LH	400 25.06.2016	666,00	EUR	A310-300	280	89	66.0	
LH	400 23.07.2016	666,00	EUR	A310-300	280	51	38.0	

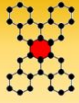
```

*&-----
*&      Module STATUS_9000 OUTPUT
*&-----
*      Oberfläche
*-----
MODULE status_9000 OUTPUT.
  SET PF-STATUS 'STATUS9000'.
  SET TITLEBAR 'TITLE9000'.

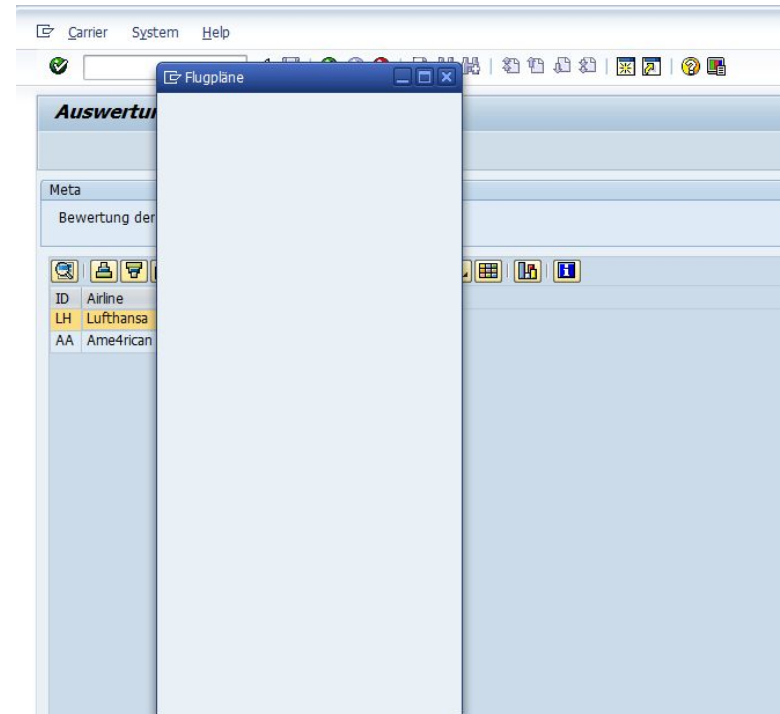
ENDMODULE.                                " STATUS_9000 OUTPUT|
    
```

SAP

PopUp 1



```
METHOD on_double_click.  
    MESSAGE e_row-index TYPE 'I'.  
* Lesen der FG aus der Datentabelle mit Index  
    DATA: ls_fg_eval LIKE LINE OF gt_fg_eval,  
          lo_popup TYPE REF TO cl_gui_dialogbox_container,  
          lo_alv TYPE REF TO cl_salv_table.  
  
    READ TABLE gt_fg_eval INTO ls_fg_eval INDEX e_row-index.  
    IF sy-subrc = 0.  
* TODO: Refactor: Unterprogramm oder Methode anlegen  
        SELECT * FROM spfli INTO TABLE gt_spfli  
            WHERE carrid = ls_fg_eval-carrid.  
* Anzeige der Daten  
* - Popup: CL_GUI_DIALOGBOX_CONTAINER  
* - ALV: CL_SALV_TABLE  
        IF lo_popup IS NOT BOUND.  
            CREATE OBJECT lo_popup  
                EXPORTING  
                    width = 300 " Width of This Container  
                    height = 300 " Height of This Container  
                    caption = 'Flugpläne'. " Dialog Box Caption  
* Registrierung für die Behandlung  
            SET HANDLER lcl_event_handler=>on_close FOR lo_popup.  
  
        ENDIF.  
    ENDIF.  
  
ENDMETHOD. "on_double_click
```



PopUp 2



```
METHOD on_double_click.  
* MESSAGE e_row-index TYPE 'I'.  
* Lesen der FG aus der Datentabelle mit Index  
DATA: ls_fg_eval LIKE LINE OF gt_fg_eval,  
      lo_popup TYPE REF TO cl_gui_dialogbox_container,  
      lo_alv TYPE REF TO cl_salv_table.  
  
READ TABLE gt_fg_eval INTO ls_fg_eval INDEX e_row-index.  
IF sy-subrc = 0.  
* TODO: Refactor: Unterprogramm oder Methode anlegen  
SELECT * FROM spfli INTO TABLE gt_spfli  
WHERE carrid = ls_fg_eval-carrid.  
* Anzeige der Daten  
- Popup: CL_GUI_DIALOGBOX_CONTAINER  
- ALV: CL_SALV_TABLE  
IF lo_popup IS NOT BOUND.  
CREATE OBJECT lo_popup  
EXPORTING  
width = 300 " Width of This Container  
height = 300 " Height of This Container  
caption = 'Flugpläne'. " Dialog Box Caption  
* Registrierung für die Behandlung  
SET HANDLER lcl_event_handler =>on_close FOR lo_popup.  
* ALV Instanziierung  
TRY.  
cl_salv_table =>factory (   
EXPORTING  
list_display = IF_SALV_C_BOOL_GAP=>FALSE " ALV Displayed in List Mode  
r_container = lo_popup " Abstract Container for GUI Controls  
* container_name =  
IMPORTING  
r_salv_table = lo_alv " Basis Class Simple ALV Tables  
CHANGING  
t_table = gt_spfli  
).  
* Daten anzeigen  
lo_alv ->display ( ).  
CATCH cx_salv_msg. " ALV: General Error Class with Message  
  
ENDTRY.  
  
ENDIF.  
ENDIF.  
ENDMETHOD.
```

The screenshot shows the SAP GUI interface. The main window is titled "Auswertung zu den Fluggesellschaften" (Evaluation for Airlines). It contains a table with the following data:

ID	Airline	Evaluation
AA	American Airlines	3
AB	Air Berlin	0
AC	Air Canada	0
AF	Air France	0
AZ	Alitalia	0
BA	British Airways	0
BL	Bruch Landung	0
C..	Communist Air	0
CO	Continental Airlines	0
D...	DAB ON THEM HATERS	0
DL	Delta Airlines	0
FJ	Air Pacific	0
HS	Han Solo	0
JL	Japan Airlines	0
LH	Lufthansa	4
LIT	ON FLEEK AIRLINE	0
NA	Not an Airline	0
NG	Lauda Air	0
NW	Northwest Airlines	0
PA	Panda Airlines	0
QF	Qantas Airways	0
SA	South African Air.	0
SQ	Singapore Airlines	0
SR	Swiss	0

A popup window titled "Flugpläne" is open, displaying a table of flight data:

Cl...	Air...	Flig...	Co...	Depart. city
0...	LH	400	DE	FRANKFURT
0...	LH	401	US	NEW YORK
0...	LH	402	DE	FRANKFURT
0...	LH	24...	DE	FRANKFURT
0...	LH	24...	DE	BERLIN

Suchkontext Selektionsbild



```

43 | pa_eval TYPE z0020030careval-ev.
46 | * Eingabe mit Suchkontext
47 | PARAMETERS: pa_car2 TYPE sflight-carrid,
48 |             pa_con2 type sflight-connid.
49 |
  
```

My first SAP program

Airline: LH
 PA_CON:
 Evaluation:
 PA_CAR2: LH

Flight Connection Number (1) 5 Entries found

Restrictions

Airline: LH

No.	Depart. ci	Arrival city
0400	FRANKFURT	NEW YORK
0401	NEW YORK	FRANKFURT
0402	FRANKFURT	NEW YORK
2402	FRANKFURT	BERLIN
2407	BERLIN	FRANKFURT

Transp. Table: SFLIGHT Active
 Short Description: Flight

Display Field SFLIGHT-CONNID

Attributes

Data element attribute

Data element: S_CONN_ID
 Short Description: Flight Connection Number
 Domain: S_CONN_ID
 Data Type: NUMC
 No. of Characters: 4 Sign
 Internal format: ABAP type, Internal Length

AM/PM-Enabled: Output Length: 4
 Default field name: CONNECTID
 Basic direction is set to LTR: No BIDI Filtering:
 No Input History:

Possible entries

Origin of the input help: Input help implemented with check table
 Search Help Name: H_SPFLI
 Fixed val. ex.: Domain: S_CONN_ID
 Foreign key exists: Check table: SPFLI

Field list: MANDI, CARRID, CONNID, FLDATE, PRICE, CURRENCY, PLANETYPE, SEATSMAX, SEATSOCC, PAYMENTISUM, SEATSMAX_B, SEATSOCC_B, SEATSMAX_F, SEATSOCC_F

Elementary srch hlp: H_SPFLI Active
 Short description: Help view for table SPFLI

Attributes Definition

Data collection: Selection method: SPFLI, Text table:
 Dialog behavior: Dialog type: Display values immediately, Hot key:

Search help exit:

Parameter

Search help parameter	IMP	EXP	LPos	SPos	SDis	Data element	M...	Default
CARRID	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	1	1	<input type="checkbox"/>	S_CARR_ID	<input type="checkbox"/>	
CONNID	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	2	2	<input type="checkbox"/>	S_CONN_ID	<input type="checkbox"/>	
CITYFROM	<input type="checkbox"/>	<input checked="" type="checkbox"/>	3	3	<input type="checkbox"/>	S_FROM_CIT	<input type="checkbox"/>	
CITYTO	<input type="checkbox"/>	<input checked="" type="checkbox"/>	4	4	<input type="checkbox"/>	S_TO_CITY	<input type="checkbox"/>	

Done!



Done!

Transaktionscodes



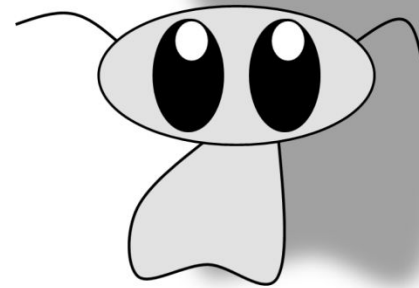
Abkürzung	Erläuterung
SE80	Object Navigator
SE11	ABAP Dictionary
SE24	Class Builder
SE63	Übersetzungswerkzeug für Texte
SU01	Benutzerverwaltung
SICF	Servicebaum
SE84	Repository Infosystem
STMS	Transport und Management System
SE09	Transport Organizer
SE16	Tabellenpflege

Konventionen



*l** Lokal
*g** Global
*i** Input
*r** Returning
*c** Changing
*+d_** elementares Datenobjekt (DO)
*+s_** strukturiertes DO
*+t_** tabellenartiges DO
*pa_** Parameter
*READ_** Lese Op auf DB
*OUTPUT_** Ausgabe Op auf Liste

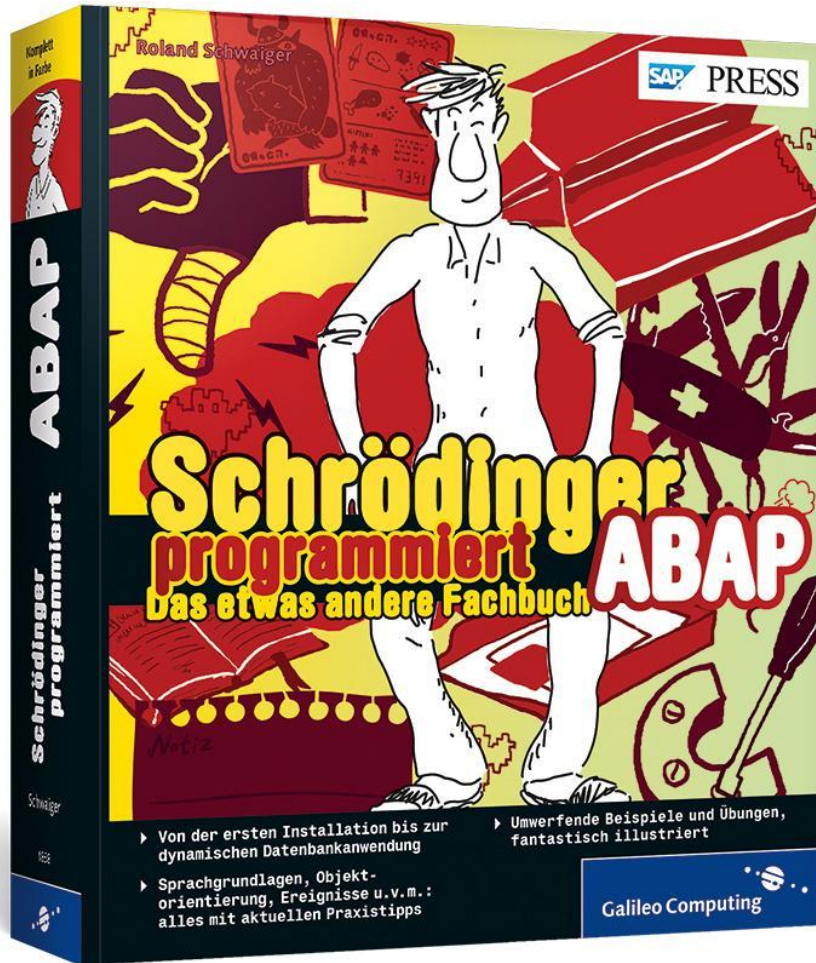
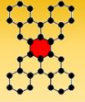
Literaturliste



- www.wikipedia.org
 - sdn.sap.com
 - help.sap.com
 - www.rolandschwaiger.at
-
- (2016) Dominik Ofenloch und Roland Schwaiger (2016). "**Getting Started with Web Dynpro ABAP**" SAP PRESS, End of 11/2016, ISBN(13) 978-1-5922-9311-7
 - (2016) Dominik Ofenloch und Roland Schwaiger (2016). "**Einstieg in Web Dynpro ABAP**" SAP PRESS, Ende 04/2016, ISBN(13) 978-3-8362-1315-8
 - (2008) Roland Schwaiger (2008). "**Sprachen und Standards für IST- und SOLL-Prozessbeschreibungen im betrieblichen Umfeld**", Books on Demand, 2008, ISBN(13) 978-3-8370-6322-6
 - (2016) Roland Schwaiger und Martin Schwaiger (2016). "**Agile Prozesserfassung**", Books on Demand, 09.2016, ISBN(13) 978-3-8391-6919-3

Möge der ABAP mit dir sein!

facet.at
Plan-Do-Check-Act

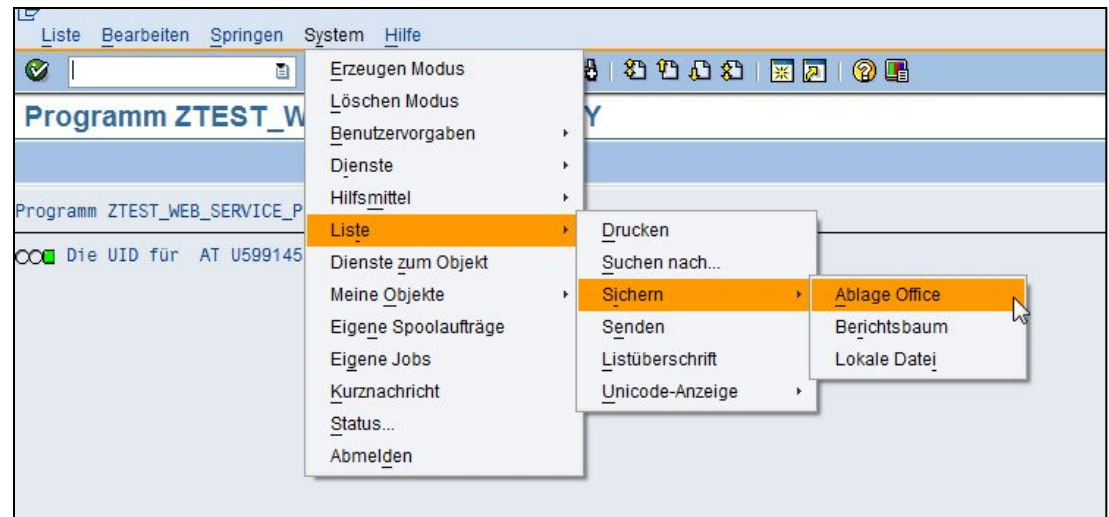
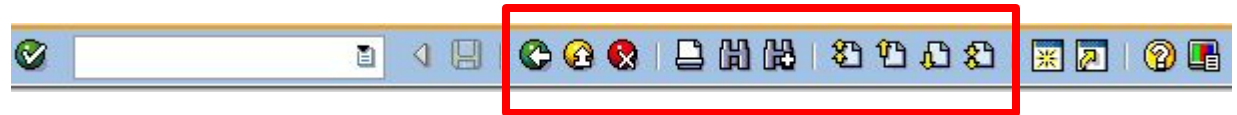


ABAP Liste

ABAP Liste

Standard Funktionen

- Suche
- Blättern
- Exportieren
- ...



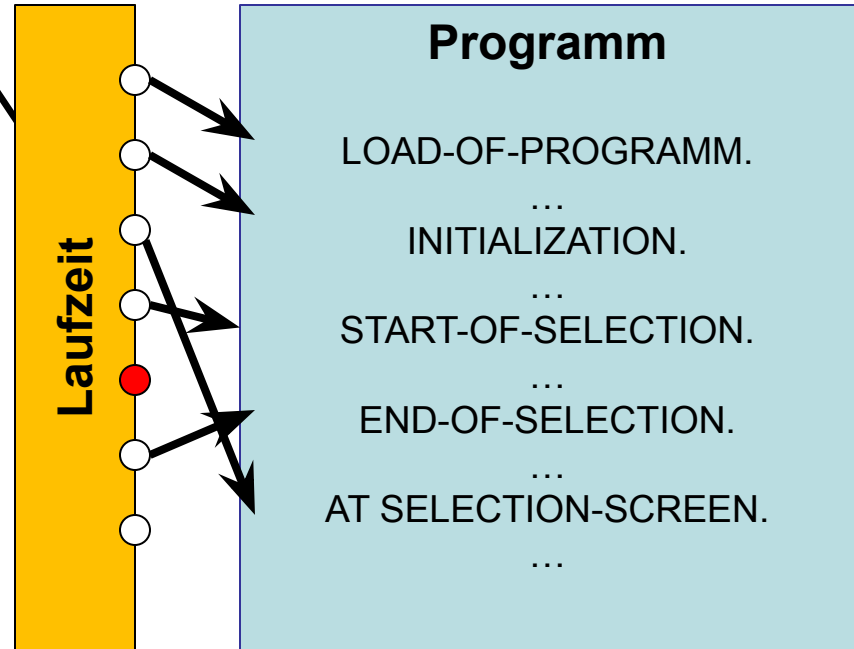
ABAP Liste

Ereignisse

- START-OF-SELECTION
- END-OF-SELECTION
- TOP-OF-PAGE
- ...

```
TYPE-POOLS: abap,  
            icon.
```

```
START-OF-SELECTION.  
  
* Gültigkeit ausgeben  
  IF ls_response-valid = abap_true.  
    WRITE: / icon_green_light AS ICON, 'Die UID für ', pa_ms, pa_vat, ' ist gültig!'.  
  ELSE.  
    WRITE: / icon_red_light AS ICON, 'Die UID für ', pa_ms, pa_vat, ' ist ungültig!'.  
  ENDIF.
```



ABAP OO

ABAP Klasse

Erbung/Vererbung

- Redefinition
- Casting (Up/Down)

