

Einführung in Programmierung mit ABAP:

Teil 7: Interaktive Listen

Prof. Dr. Peter Hohmann

Technische Hochschule Mittelhessen

FB MNI

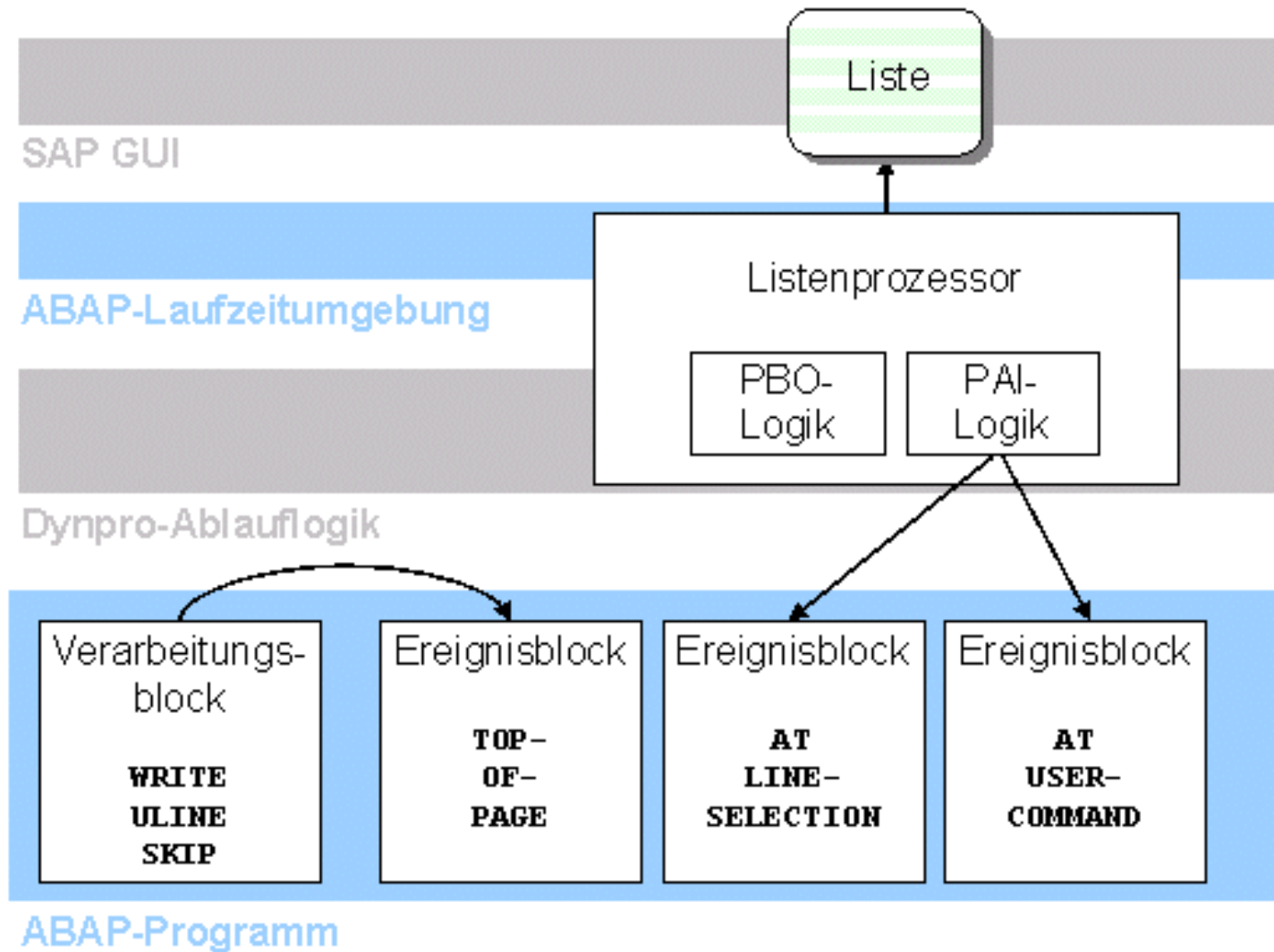
www.prof-dr-hohmann.de

SS 2017

Listen-Ereignisse

- TOP-OF-PAGE .
- TOP-OF-PAGE DURING LINE-SELECTION .
- END-OF-PAGE .
- AT LINE-SELECTION .
 - verdichtete Grundliste, Verzweigungsliste
 - Doppelklick auf der Liste (F2, Funktionscode PICK)
- AT USER-COMMAND .
 - Zuordnung von Funktionscodes zu Funktionstasten (GUI-Status)
 - Abfragen der Funktionscodes

Ablauflogik Listen



Systemfelder Listenereignisse

- Listenindex (Zählung beginnt bei 0):
 - aktueller Listenindex: SY-LSIND
 - Liststufe auslösendes Ereignis: SY-LISTI
- Information aus der angewählten Liste:
 - Zeilennummer : SY-LILLI
 - Inhalt der markierten Zeile: SY-LISEL

Hierarchische Liste

Grundliste

AT LINE SELECTION

1

Verzweigung

2

Verzweigung

....

20

Verzweigung

Dialoggesteuerte Listen

Grundliste

AT USER COMMAND

SET PF-STATUS

FKT1

FKT2

Verzweigung

Verzweigung

Werteübergabe mit HIDE I

```
DATA: wa TYPE scarr .
```

```
* Grundliste
```

```
SELECT * FROM scarr INTO wa .
```

```
  WRITE: / wa-carrid,
```

```
         wa-carrname .
```

```
  HIDE wa .
```

```
ENDSELECT .
```

```
* Verzeigungsliste
```

```
AT LINE-SELECTION .
```

```
SELECT * FROM spfli INTO ...
```

```
  WHERE carrid = wa-carrid .
```

```
  <Sekundärliste>
```

```
ENDSELECT .
```

HIDE-Bereich

Zeile Inhalt wa

1	AB Air Berlin
2	AC Air Canada
...	...
...	...
...	...
11	LH Lufthansa
...	...
...	...

Werteübergabe mit HIDE II

```
DATA: wa TYPE scarr .
```

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

```
  * Grundliste
```

```
  SELECT * FROM scarr INTO wa .
```

```
    WRITE: / wa-carrid, wa-carrname .
```

```
    HIDE wa .
```

```
  ENDSELECT .
```

```
  CLEAR wa .
```

```
  * Verzeigungsliste
```

```
  AT LINE-SELECTION .
```

```
  IF NOT wa IS INITIAL AND sy-lsind = 1 .
```

```
    SELECT * FROM spfli INTO wa_spfli
```

```
      WHERE carrid = wa-carrid .
```

```
      <Sekundärliste>
```

```
    ENDSELECT .
```

```
  ENDIF.
```

```
  CLEAR wa.
```

HIDE-Bereich

Zeile Inhalt wa

1	AB Air Berlin
2	AC Air Canada
...	...
...	...
...	...
11	LH Lufthansa
...	...
...	...

Prof. Dr. P. Hohmann

Prof. Dr. M. Scheer, THM

Iterationsmodell while-Schleife

<Erster Fall außerhalb der Schleife behandeln, i.d.R. wird damit die *Schleifenbedingung initialisieren*.>;

```
while (<Bedingung>) {  
    <Bearbeiten des ersten / aktuellen Falls.>;  
    <Nächster Fall behandeln,  
        "Update" auf Schleifenbedingung.>;  
}
```


Listenzeilen lesen I

```
DATA: wa_scarr      TYPE scarr  
      select(1) TYPE c ,  
      index  TYPE i.
```

```
SET PF-STATUS 'LISTE' .
```

*** Grundliste**

```
SELECT * FROM scarr INTO wa_scarr  
      WRITE: / select AS CHECKBOX,  
              wa-carrid, wa-carrname  
ENDSELECT .
```

The screenshot shows the SAP program ZS12_006_DEMO_ with a menu bar (Liste, Bearbeiten, Springen, System) and a toolbar. The main area is titled 'Fluggesellschaften' and contains a table 'Grundliste' with columns for airline codes and names. The table has 16 rows, each with a checkbox in the first column. The checkboxes for AB (Air Berlin), AF (Air France), and LH (Lufthansa) are checked. The other checkboxes are unchecked.

	AA	AB	AC	AF	AZ	BA	CO	DL	FJ	JL	LH	NG	NW	QF	SA	SQ	SR	UA
	American Airlines	Air Berlin	Air Canada	Air France	Alitalia	British Airways	Continental Airlines	Delta Airlines	Air Pacific	Japan Airlines	Lufthansa	Lauda Air	Northwest Airlines	Qantas Airways	South African Air.	Singapore Airlines	Swiss	United Airlines

Prof. Dr. P. Hohmann

Listenzeilen lesen II

* Ereignis wird über eine Funktion (Fkt-Code) ausgelöst
AT USER-COMMAND .

index = 1.

CLEAR: *select, wa_scarr .*

READ LINE *index* **FIELD VALUE**

select wa_scarr-carrid wa_scarr-carrname .

WHILE *sy-subrc = 0 .*

IF *select = 'X' .*

< gelesene Zeile verarbeiten >

ENDIF .

index = index + 1 .

CLEAR: *select, wa_scarr .*

READ LINE *index* **FIELD VALUE**

select wa_scarr-carrid wa_scarr-carrname .

ENDWHILE .

Listenzeile lesen III

- * Ereignis wird über eine Funktion (Fkt-Code) ausgelöst,
- * Funktionscode abfangen in einer CASE-Anweisung umsetzen.
- * In sy-ucomm steht der Funktionscode.

```
AT USER-COMMAND .  
  IF ( sy-lsind = 1 ) .  
    CASE sy-ucomm .  
      WHEN 'FKT5' .
```

*< Liste einlesen und selektierte Zeilen
verarbeiten (siehe vorige Folie)>*

```
      WHEN 'FKT6' .
```

< Liste einlesen Selektionen zurück nehmen>

```
    ENDCASE .  
  ENDIF .
```

Listenzeilen einlesen

* aktuelle Zeile in das Systemfeld *sy-lisel*
* einlesen

```
READ CURRENT LINE .
```

* Zeile *znr* in den Bereich *wa* lesen

```
READ LINE znr LINE VALUE INTO wa .
```

* Listenfeld der Zeile *znr* lesen

```
READ LINE znr FIELD VALUE  
  listenfeld1 [INTO progfeld1]  
  [listenfeld2 [INTO progfeld2] ]... .
```

aktuelle Listenzeilen modifizieren

* Zeilenfarbe ändern

MODIFY CURRENT LINE

LINE FORMAT *option1* [*option2*]

* Feldfarbe ändern

MODIFY CURRENT LINE

FIELD FORMAT *feld1 format1*
[*feld2 format2*]

* Wert eines Felds ändern

MODIFY CURRENT LINE

FIELD VALUE *feld1 FROM wert1*
[*feld2 FROM wert2*]