

# Database Access

Adrian Streitz, Johannes Rank, Borys Levkovskiy



# Prerequisites

---

- Experience with ABAP Workbench and Eclipse
- Basic ABAP programming skills

# Agenda

- I. Data dictionary**
- II. SAP flight example**
- III. SAP OpenSQL**

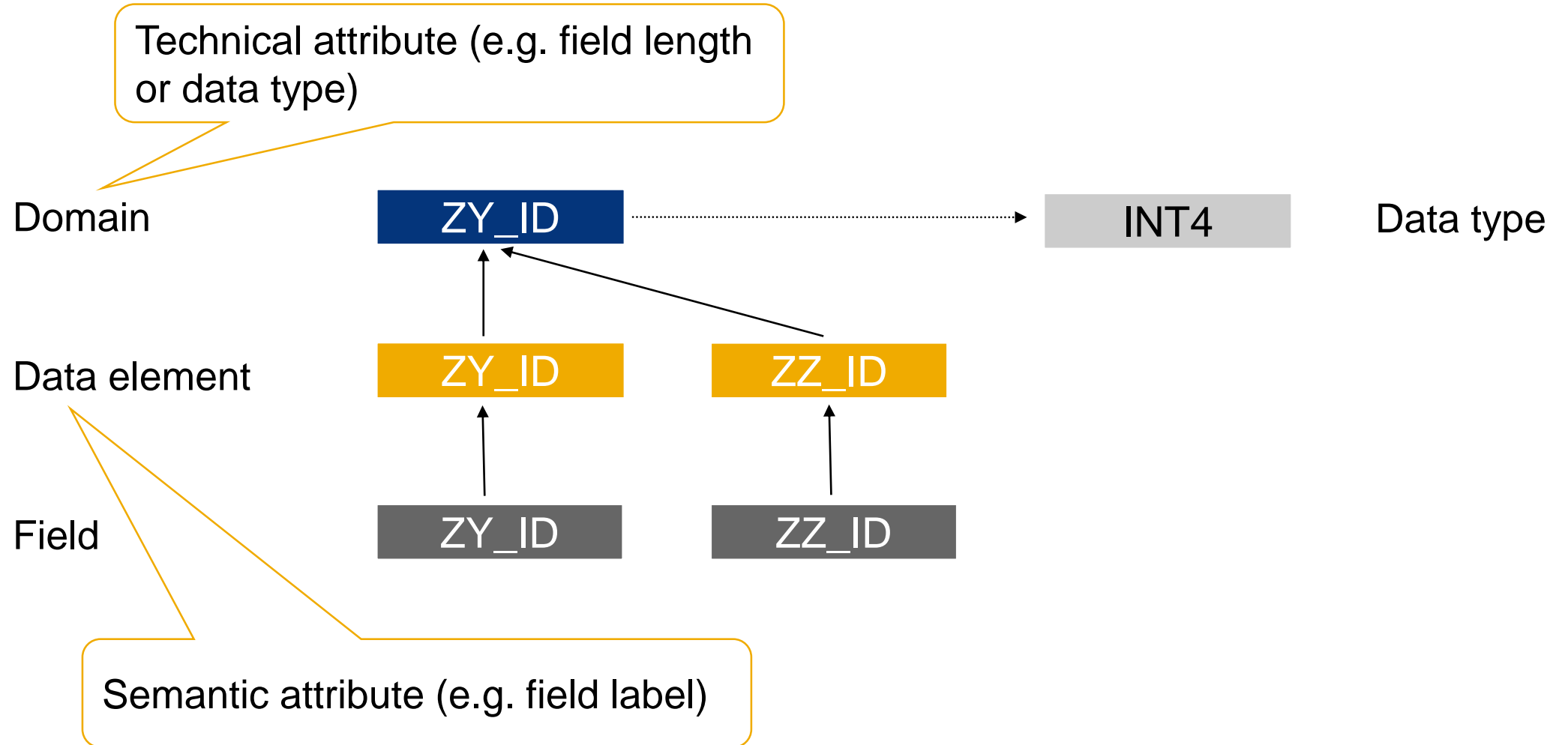
# What is the data dictionary?

---

- Data dictionary = global directory for data types
- Assignment of help texts and explanations for data types in different languages
- Entity–relationship model can be shown as a figure automatically
- Most important objects: structure, table, data element and domain

# I. Data dictionary

## Hierarchy



I. Data dictionary

# Hierarchy

---

Structure

Field			
-------	--	--	--

Table

Field			
Field			

# Working with the data dictionary

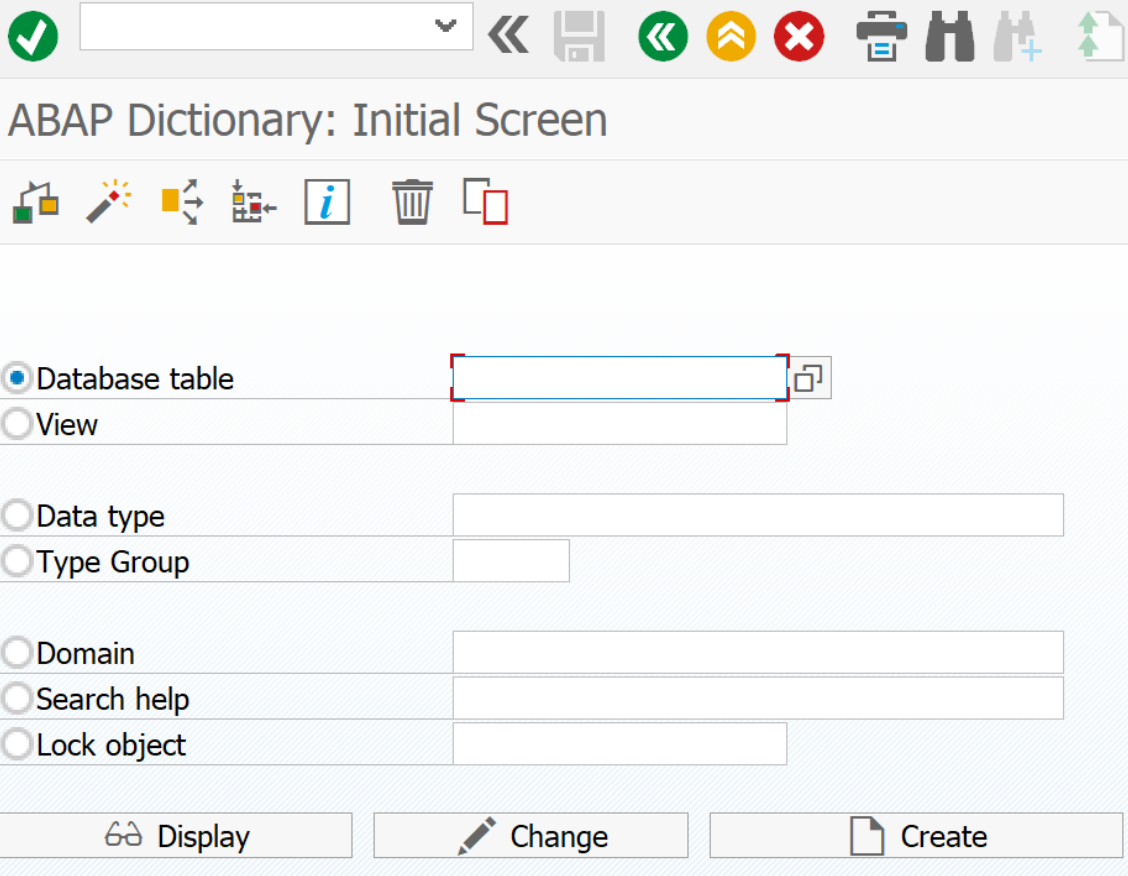
---

- Menu path: **Tools • ABAP Workbench • Development • Dictionary**
- Transaction code: SE11
- View, edit, delete, create tables, data types, domains definitions etc.
- Tables and views from the ABAP dictionary represent tables and views from the database
- User interfaces are generated automatically when changing views, tables etc.
- Table may be changed after they are created without losing data



# Overview about all objects

- Database table
- View
- Data type
  - Data element
  - Structure
  - Database table
  - Table types
  - Views
  - Class / interface
- Type group
- Domain
- Search help
- Lock object



The screenshot shows the 'ABAP Dictionary: Initial Screen' in a SAP environment. At the top, there is a toolbar with various icons for navigation and actions. Below the toolbar, the title 'ABAP Dictionary: Initial Screen' is displayed. Underneath, there is a row of icons representing different object types. The main area of the screen contains a list of object types with radio buttons and input fields. The 'Database table' option is selected. At the bottom, there are three buttons: 'Display', 'Change', and 'Create'.

ABAP Dictionary: Initial Screen

☒ Database table

☐ View

☐ Data type

☐ Type Group

☐ Domain

☐ Search help

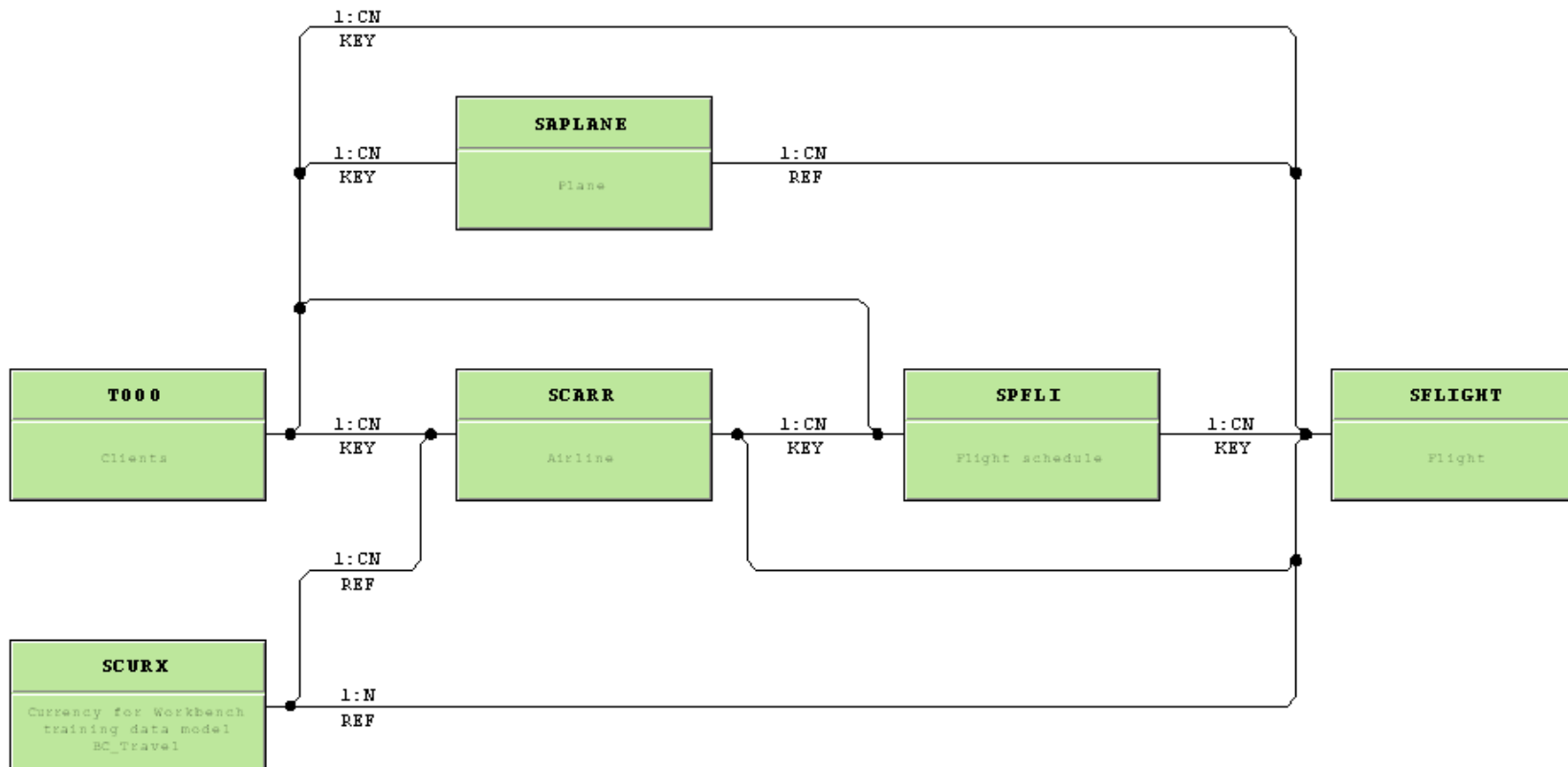
☐ Lock object



# I. Data dictionary

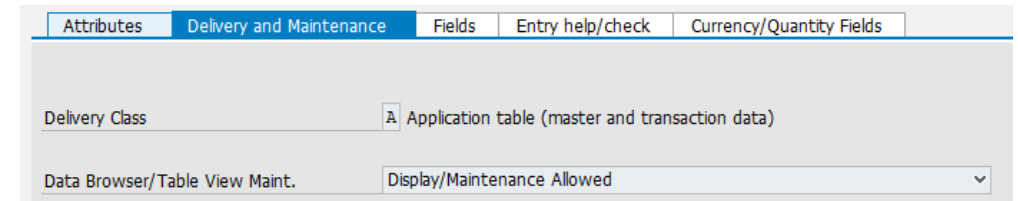
## Visualization

- Foreign key relationships can be visualized in data dictionary
- Table SFLIGHT



# Data browser

- Menu path: **Tools • ABAP Workbench • Overview • Data Browser**
- Transaction code: SE16
- **View table content**
- Add new entries to tables when adding is permitted for the table
  - It is possible to deny adding entries to a table. To verify if it is allowed please look at the delivery and maintenance settings in the data dictionary.
- Browse the entry help table content
- Copy table content to transport request
- Download table content

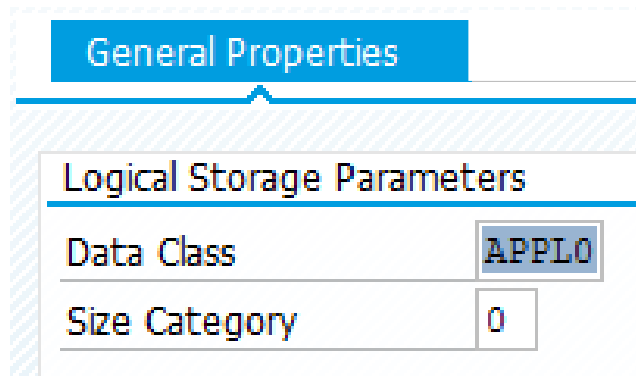


The screenshot shows the 'Delivery and Maintenance' tab in the SAP Data Browser. It contains two main sections: 'Delivery Class' and 'Data Browser/Table View Maint.'. The 'Delivery Class' section has a dropdown menu with the value 'A Application table (master and transaction data)'. The 'Data Browser/Table View Maint.' section has a dropdown menu with the value 'Display/Maintenance Allowed'.

Attributes	Delivery and Maintenance	Fields	Entry help/check	Currency/Quantity Fields
Delivery Class: A Application table (master and transaction data)				
Data Browser/Table View Maint.: Display/Maintenance Allowed				

## Technical settings: Data Class

- SE11 -> Technical Settings



The screenshot shows the SAP SE11 Technical Settings interface. At the top, there is a blue tab labeled 'General Properties'. Below it, a section titled 'Logical Storage Parameters' is visible. This section contains two fields: 'Data Class' with the value 'APPL0' and 'Size Category' with the value '0'.

Logical Storage Parameters	
Data Class	APPL0
Size Category	0

- **APPL0** (*master data*): data that is rarely changed, but frequently read
- **APPL1** (*transaction data*): data that is frequently changed
- **APPL2** (*organizational data*): data that is defined when the system is installed and rarely changed

## Technical settings: Buffering

- SE11 -> Technical Settings

### Buffering

- ☐ Buffering Not Allowed
- ☐ Buffering allowed but switched off
- ☒ Buffering Activated

### Buffering Type

- ☐ Single Records Buffered
- ☐ Generic Area Buffered
- ☒ Fully Buffered

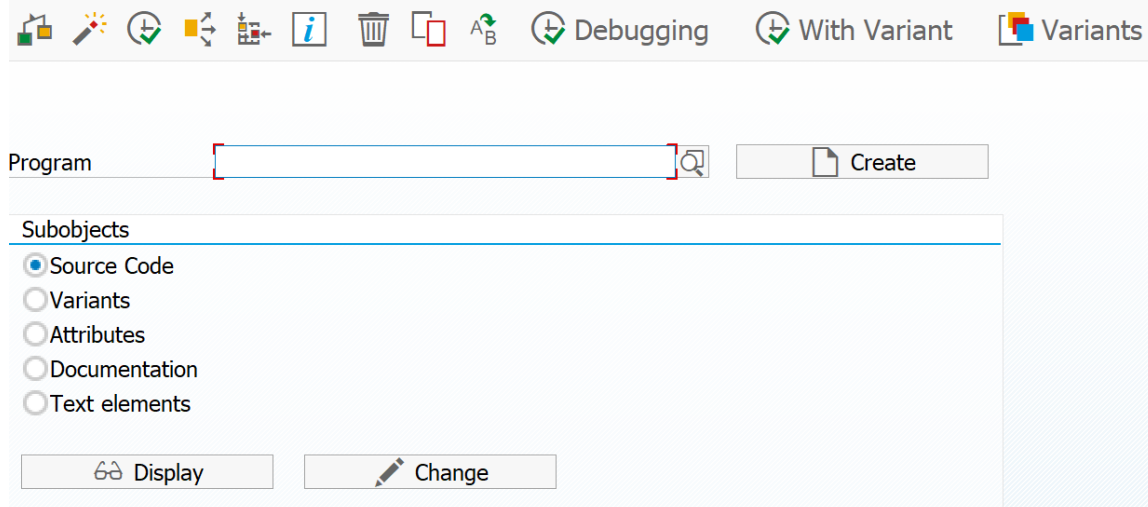
- Buffering:** make data available in memory, so request for the same data can be processed faster
- Full buffering:** Load all records of the table in the buffer, when one record is accessed
- Generic buffering:** When a record of the table is accessed, all the records that have this record in the generic key fields are loaded in the buffer
- Single-record buffering:** Only the records of a table that are really accessed are loaded into the buffer

## II. SAP flight example

# Short introduction to the SAP flight example

- Created and maintained by SAP to demonstrate database operations
- Contains exercise data for airline, flight connection number, flight date, airfare etc.
- Report for data generation: `SAPBC_DATA_GENERATOR` (use Transaction SE38)
- SAP trainings, examples from books build refer to the flight example

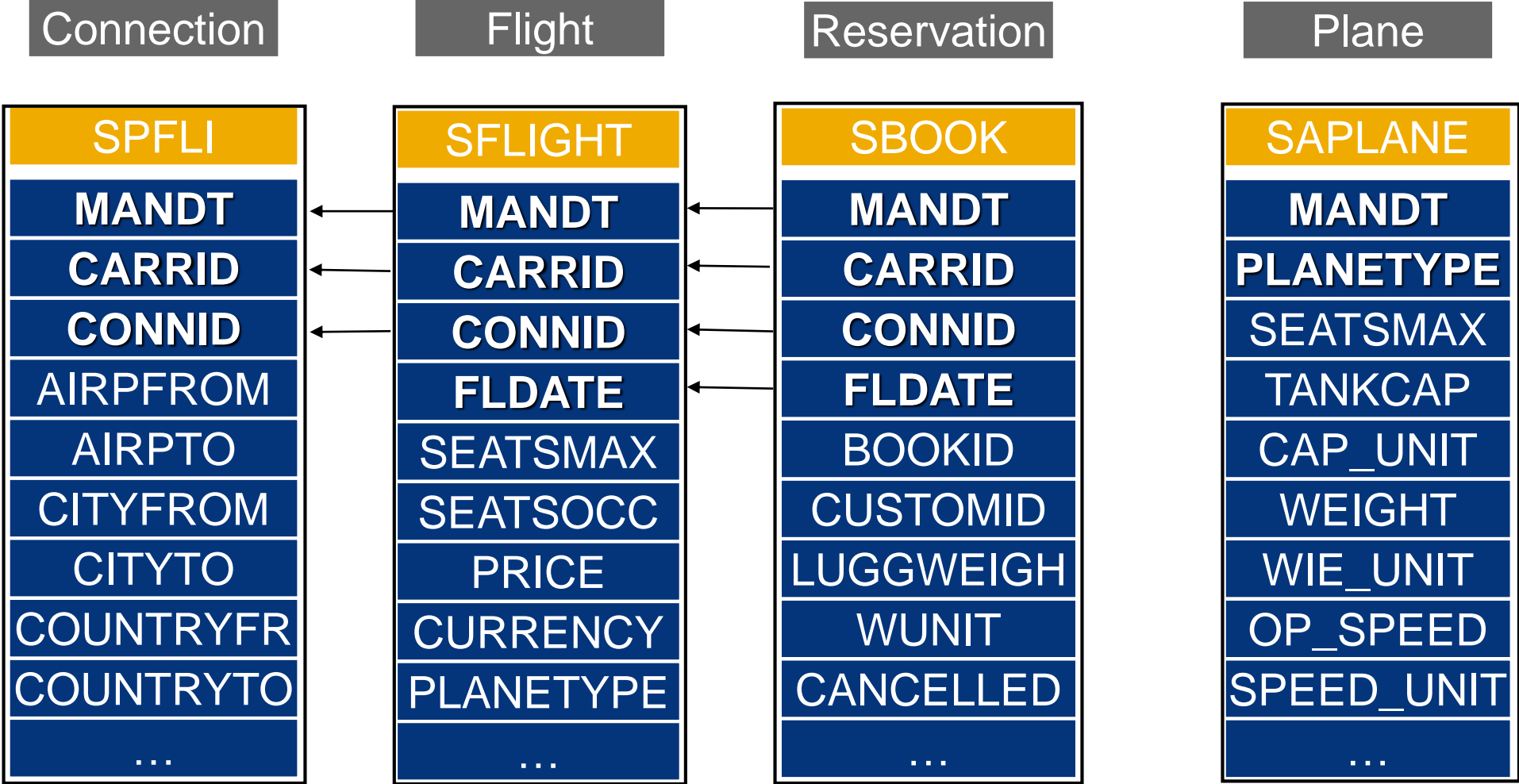
### ABAP Editor: Initial Screen



The screenshot shows the ABAP Editor's initial screen. At the top, there is a toolbar with icons for file operations (new, open, save, delete), development (run, test, debug), and other functions (info, print, undo, redo). Below the toolbar, there is a 'Program' field with a search icon and a 'Create' button. Underneath, a 'Subobjects' list is displayed with radio buttons for 'Source Code' (selected), 'Variants', 'Attributes', 'Documentation', and 'Text elements'. At the bottom, there are 'Display' and 'Change' buttons.

II. SAP flight example

# Table structure



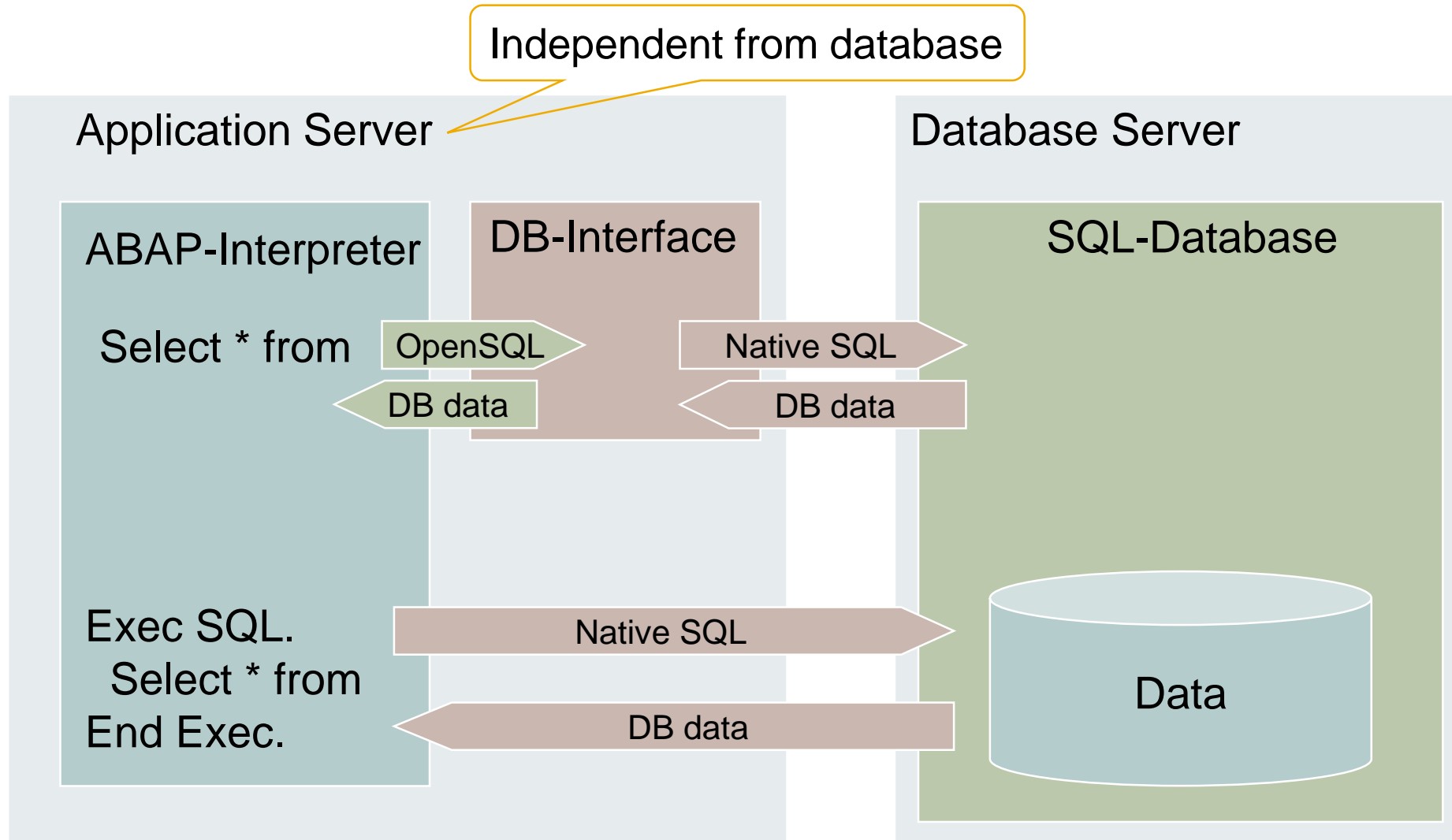
Now you know about the data dictionary and the SAP flight example.

To consolidate your knowledge, you can do task 1 and 2 of the Database Access exercise.





## OpenSQL vs. Native SQL



## Instructions (examples)

- Select:

SELECT

\*

FROM

table

INTO

structure

WHERE

condition

SELECT

FIELD

view

internal  
table

SINGLE

- Aggregation:

SELECT MAX

SELECT MIN

SELECT COUNT

- Joins:

table1 INNER JOIN table2

table1 LEFT OUTER JOIN table2

## Procedure for database access

---

#### General:

- Be specific to decrease the load on the database
- Access to database takes 10,000 time longer than access to buffers
- Avoid table scans, use indexes

#### Procedure:

- Be specific to decrease the load on the database
1. Read database data into internal tables
    - Internal tables are tables for holding data during runtime
  2. Change data per row
    - Use workareas to hold one row of the table and change data in the workarea
  3. Write changes back to database

## Types of database changes

---

Modify instruction:

```
MODIFY <dbtable> [CLIENT SPECIFIED] FROM <workarea>.
```

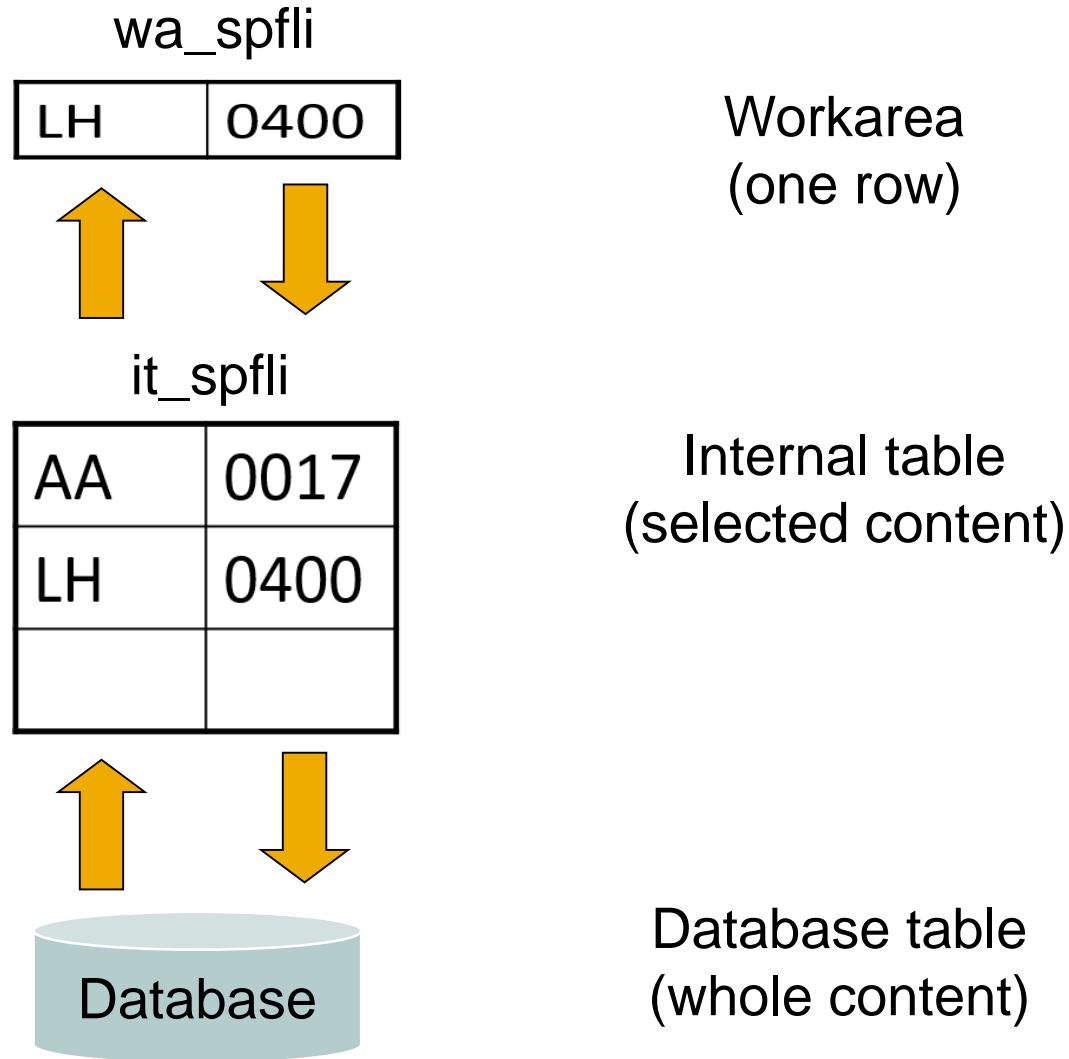
```
MODIFY <dbtable> [CLIENT SPECIFIED] FROM TABLE <internaltable>
```

- a) Modify = Update – the command updates the database table if the database table already contains dataset
- b) Modify = Insert – the command inserts new datasets into the database table

Delete instruction:

```
DELETE FROM <dbtable> WHERE [SQL statement].
```

## Procedure for database access



- Use the workarea to modify one data set
- Use the internal table to store the selected database content temporarily
- Internal tables are deleted after program is finished
- Use the database to read/write data

### III. SAP OpenSQL

# Indexes

---

#### General:

- Indexes (secondary keys) can further improve the efficiency of search operations
- Index: database object that can be defined for an individual column or a series of columns in a database table
- Consists of one or more inversion lists (set of non-overlapping numeric ranges)
- Indexes provide access to the table data via non-key columns
- Necessary if the table is frequently used without taking advantage of the primary key

## Create index (secondary key)

### Step-by-Step:

- Go to SE11, enter required Database, press change
- Switch to the tab indexes, click create -> create index

Transparent Table **Z301\_TITLE** Active

Short Description **My first Data Table**

Attributes | Delivery and Maintenance | Fields | Input Help/Check | Currency/Quantity Fields | **Indexes**

**Create Index**

Table Name **Z301\_TITLE**

Index Name

✓ ✗

- Choose an index name
- Select the fields the index should comprise

**Field Selection from Table Z301\_TITLE**

Field Name	Selection	Description
MANDT	<input type="checkbox"/>	Client
TITLE	<input checked="" type="checkbox"/>	My own data taype

☒ Database table **Z###\_TITLE**

☐ View

☐ Data type

☐ Type Group

☐ Domain

☐ Search help

☐ Lock object

Display Change Create



Now you know about SAP OpenSQL.

To consolidate your knowledge, you can do task 3 and 4 of the Database Access exercise.



# Check your knowledge

# Check your knowledge

---

- You can use the data dictionary to add entries to a table manually.
  - ☐ True ☐ False
- Explain a reason not to use Native-SQL in favor of OpenSQL (think about an example scenario where Native-SQL is problematic).
- Why is the database data read into internal tables
  - ☐ To save memory ☐ To accelerate the application ☐ To avoid high disk usage

# Solution

# Check your knowledge

---

- You can use the data dictionary to add entries to a table manually.  
☐ True   ☒ False
- Explain a reason not to use Native-SQL in favor of OpenSQL (think about an example scenario where Native-SQL is problematic).

See slide 13 - OpenSQL vs. Native SQL

- Why is the database data read into internal tables  
☐ To save memory   ☒ To accelerate the application   ☐ To avoid high disk usage

# © 2018 SAP SE or an SAP affiliate company. All rights reserved.

---

No part of this publication may be reproduced or transmitted in any form or for any purpose without the express permission of SAP SE or an SAP affiliate company.

SAP and other SAP products and services mentioned herein as well as their respective logos are trademarks or registered trademarks of SAP SE (or an SAP affiliate company) in Germany and other countries. Please see <http://global12.sap.com/corporate-en/legal/copyright/index.epx> for additional trademark information and notices.

Some software products marketed by SAP SE and its distributors contain proprietary software components of other software vendors.

National product specifications may vary.

These materials are provided by SAP SE or an SAP affiliate company for informational purposes only, without representation or warranty of any kind, and SAP SE or its affiliated companies shall not be liable for errors or omissions with respect to the materials. The only warranties for SAP SE or SAP affiliate company products and services are those that are set forth in the express warranty statements accompanying such products and services, if any. Nothing herein should be construed as constituting an additional warranty.

In particular, SAP SE or its affiliated companies have no obligation to pursue any course of business outlined in this document or any related presentation, or to develop or release any functionality mentioned therein. This document, or any related presentation, and SAP SE's or its affiliated companies' strategy and possible future developments, products, and/or platform directions and functionality are all subject to change and may be changed by SAP SE or its affiliated companies at any time for any reason without notice. The information in this document is not a commitment, promise, or legal obligation to deliver any material, code, or functionality. All forward-looking statements are subject to various risks and uncertainties that could cause actual results to differ materially from expectations. Readers are cautioned not to place undue reliance on these forward-looking statements, which speak only as of their dates, and they should not be relied upon in making purchasing decisions.