

# ABAP Course

## Chapter 4 – Database access

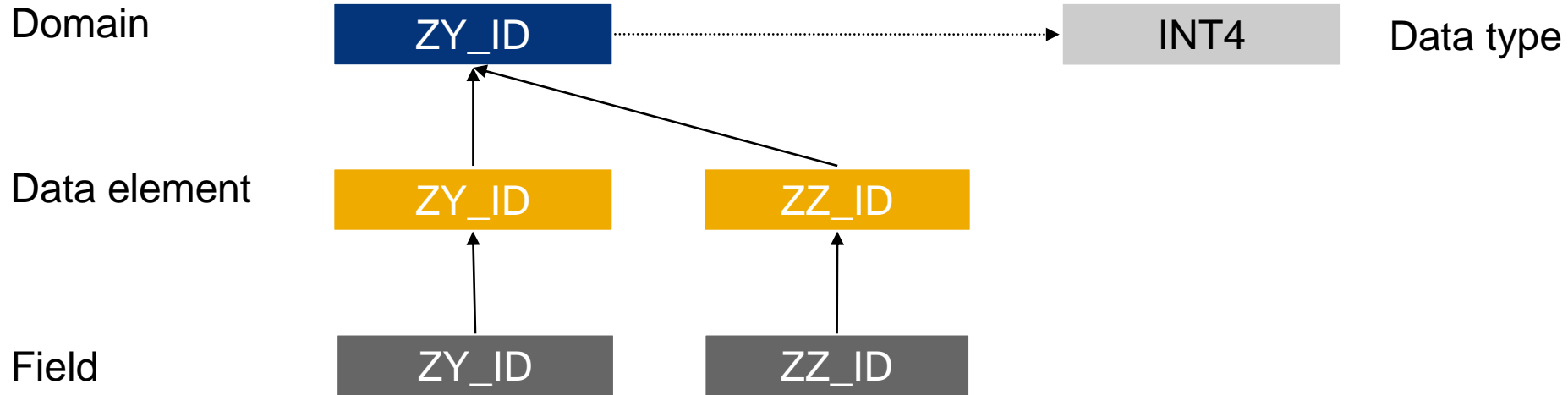
Lecturer: Robert Meyer, UCC Technische Universität München  
Author: Valentin Nicolescu, André Bögelsack

# Agenda

1. Data dictionary
2. SAP flight example
3. Database operations in SAP
4. SAP OpenSQL
5. Types of database changes
6. Types of entry help

- Data dictionary = global directory for data types
- Represent database data type (like INT4, CLNT or CURR); different from ABAP data types (like "i" or "f")
- Assignment of help texts and explanations for data types in different languages
- ERM can be drawn automatically
- Most important objects: structure, table, data element and domain

# Data dictionary



Structure

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

Table

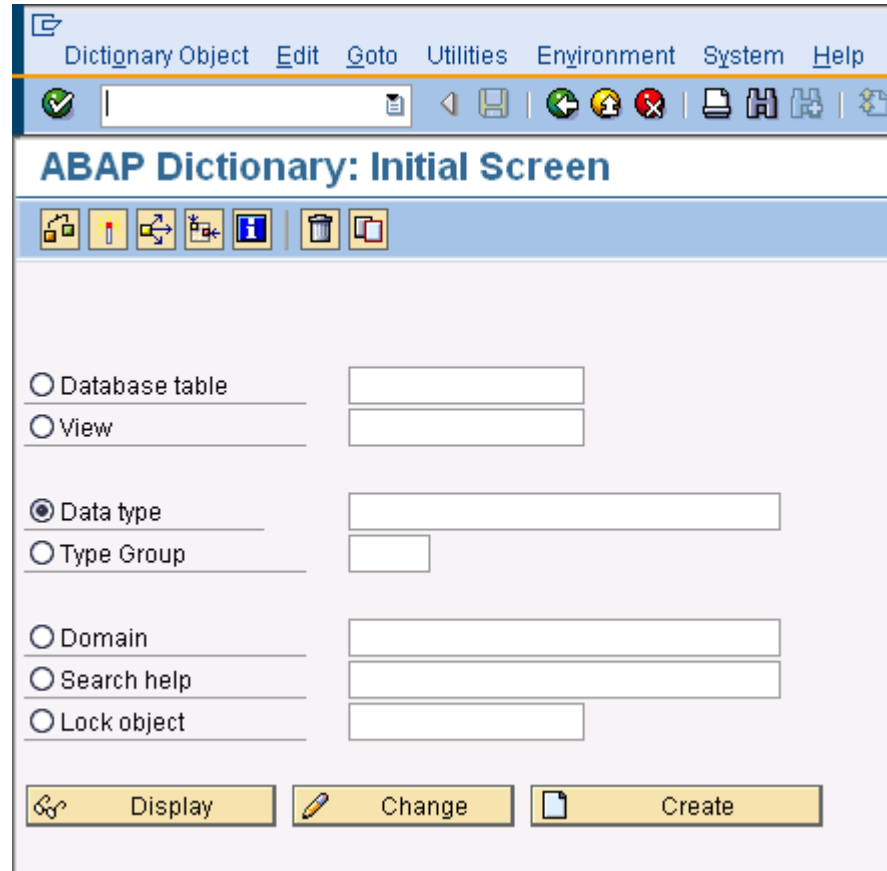
Field			
Field			

- Menu path: **Tools • ABAP Workbench • Development • Dictionary (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 interface are generated automatically when changing views, tables etc.
- Table may be changed after they are created without losing data
- SAP tables may be extended by `APPEND` structures

- Menu path: **Tools • ABAP Workbench • Overview • Data Browser (SE16)**
- View table content
- Add new entries to tables when adding is permitted for the table
- Browse the entry help table content
- Copy table content to transport request
- Download table content

# 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

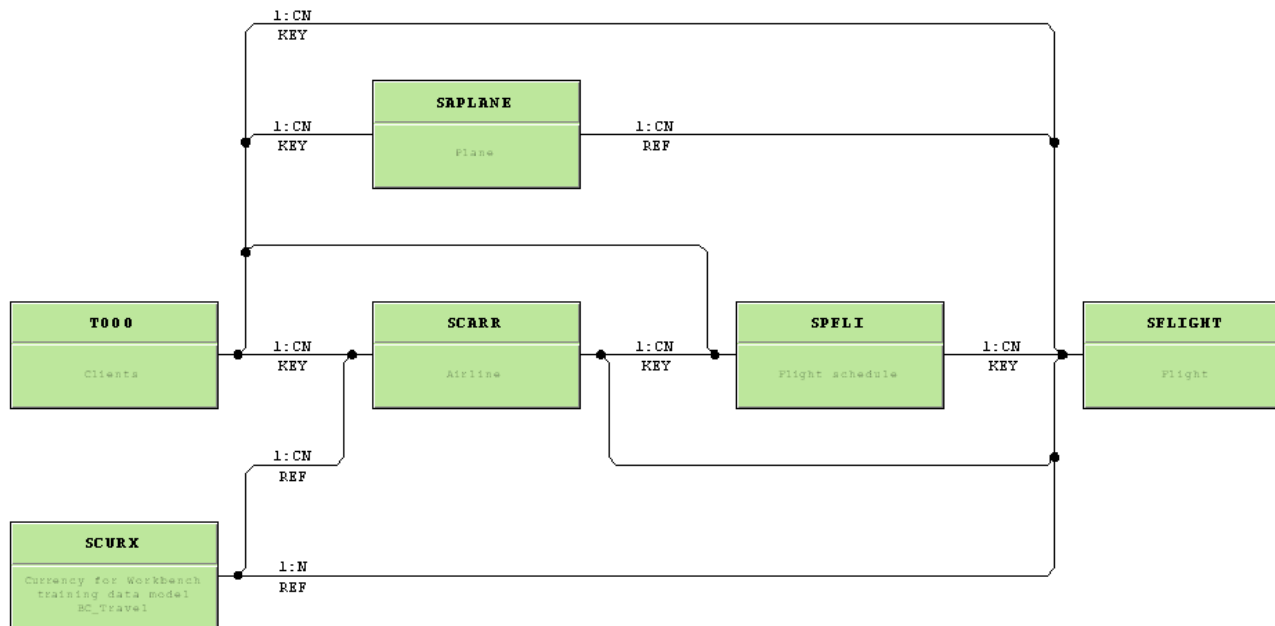


The screenshot shows the 'ABAP Dictionary: Initial Screen' window. It features a menu bar with 'Dictionary Object', 'Edit', 'Goto', 'Utilities', 'Environment', 'System', and 'Help'. Below the menu is a toolbar with various icons. The main area contains a list of object types with radio buttons and corresponding text input fields:

- ☐ Database table
- ☐ View
- ☒ Data type
- ☐ Type Group
- ☐ Domain
- ☐ Search help
- ☐ Lock object

At the bottom, there are three buttons: 'Display' (with a magnifying glass icon), 'Change' (with a pencil icon), and 'Create' (with a document icon).

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

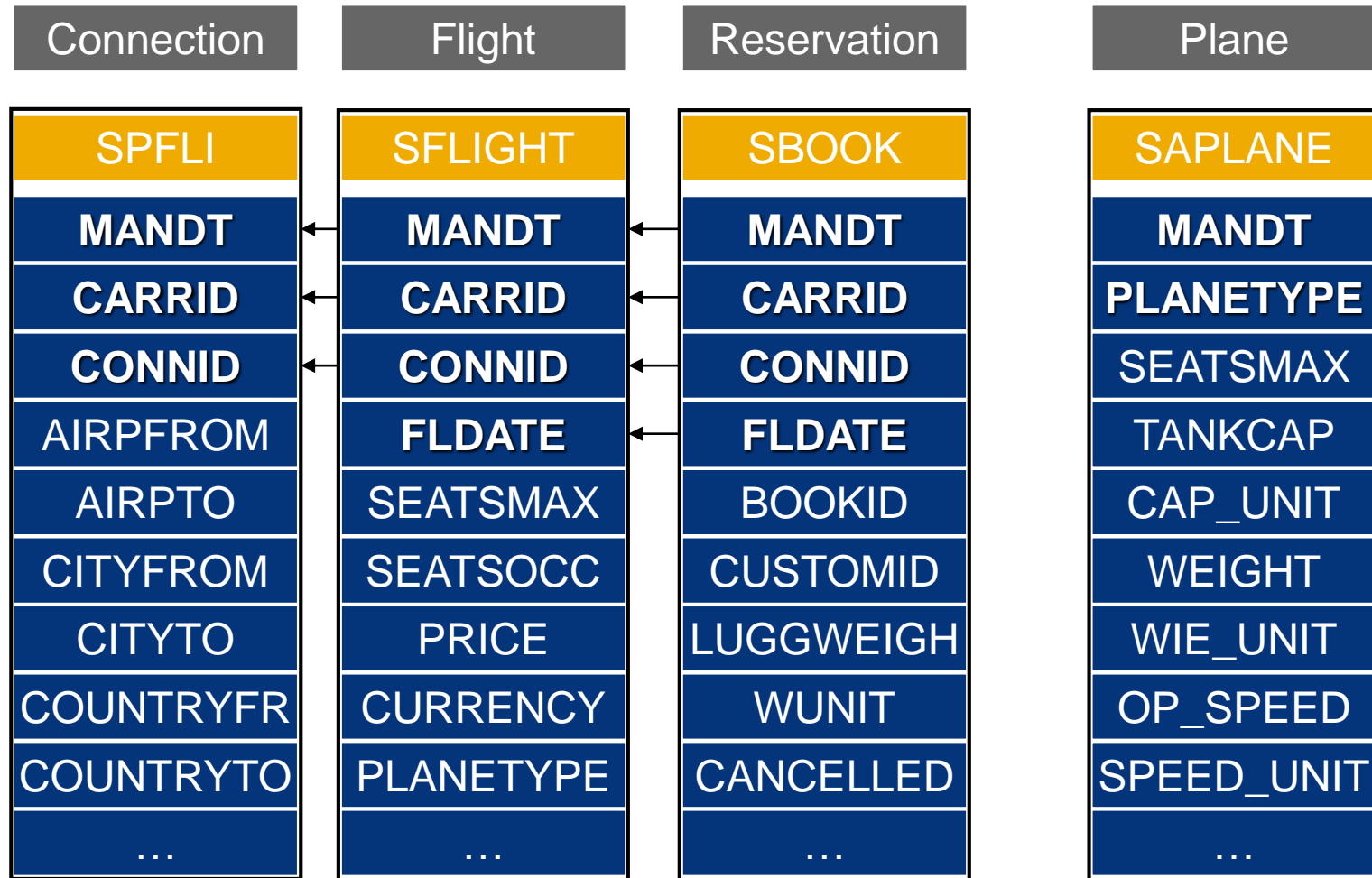




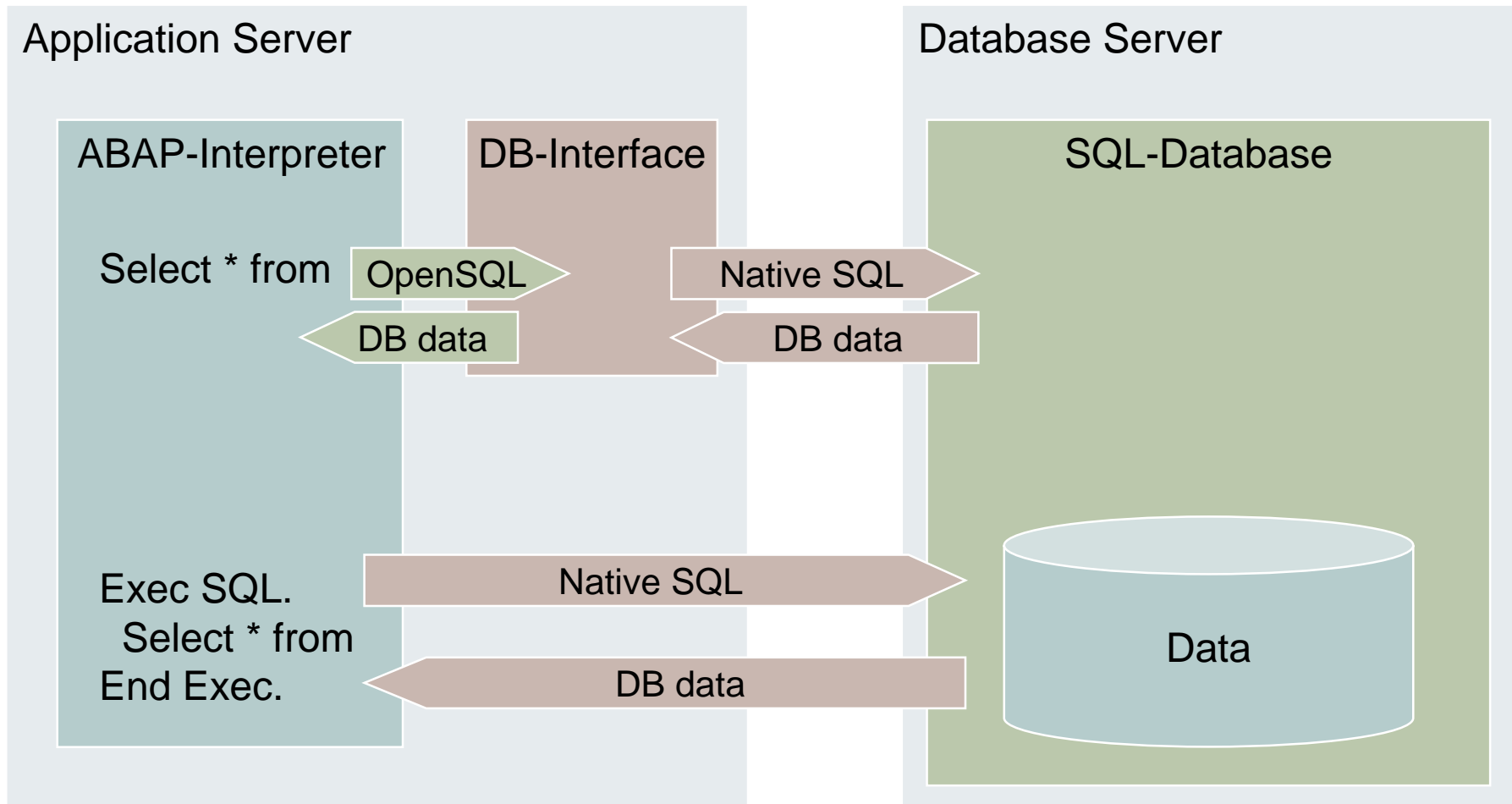
- Created and maintained by SAP to demonstrate database operations
- Contains exercise data for airline, flight connection number, flight date, airfare etc.
- Report for data generation: `SAPBC_DATA_GENERATOR`
- SAP trainings, examples from books build refer to the flight example

Run this command at the Command field:  
`/nSE38`  
Find `SAPBC_DATA_GENERATOR` -> display  
-> execute -> choose standard dataset

# SAP flight example – table structure



- Independent from database



- **Select:**
  - SELECT / SELECT SINGLE
  - \* / field 1 ..field n /
  - FROM table / view
  - INTO structure / internal table
  - WHERE condition
- **Aggregation:**
  - MAX, MIN, AVG
  - SUM, COUNT
- **Joins:**
  - Left outer join
  - Inner join
  - View

## 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:

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

# Procedure for database access

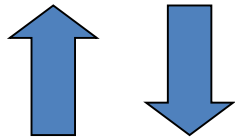
## Internal tables and workareas:

wa\_spfli

LH	0400
----	------

Workarea  
(one row)

- Use the workarea to modify one data set

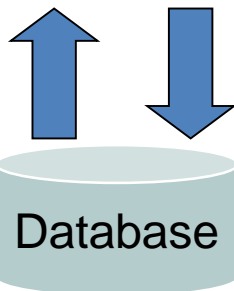


it\_spfli

AA	0017
LH	0400

Internal table  
(selected content)

- Use the internal table to store the selected database content temporarily
- Internal tables are deleted after program is finished



Database table  
(whole content)

- Use the database to read/write data

## Modify instruction:

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

- **Modify = Update** – the command updates the database table if the database table already contains dataset
- **Modify = Insert** – the command inserts new datasets into the database table
- **Warning: Expensive statement!**

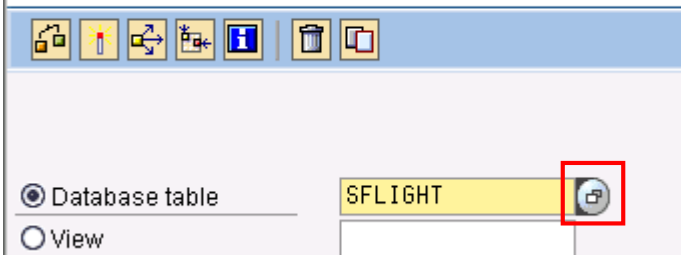
## Delete instruction:

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

# Entry help

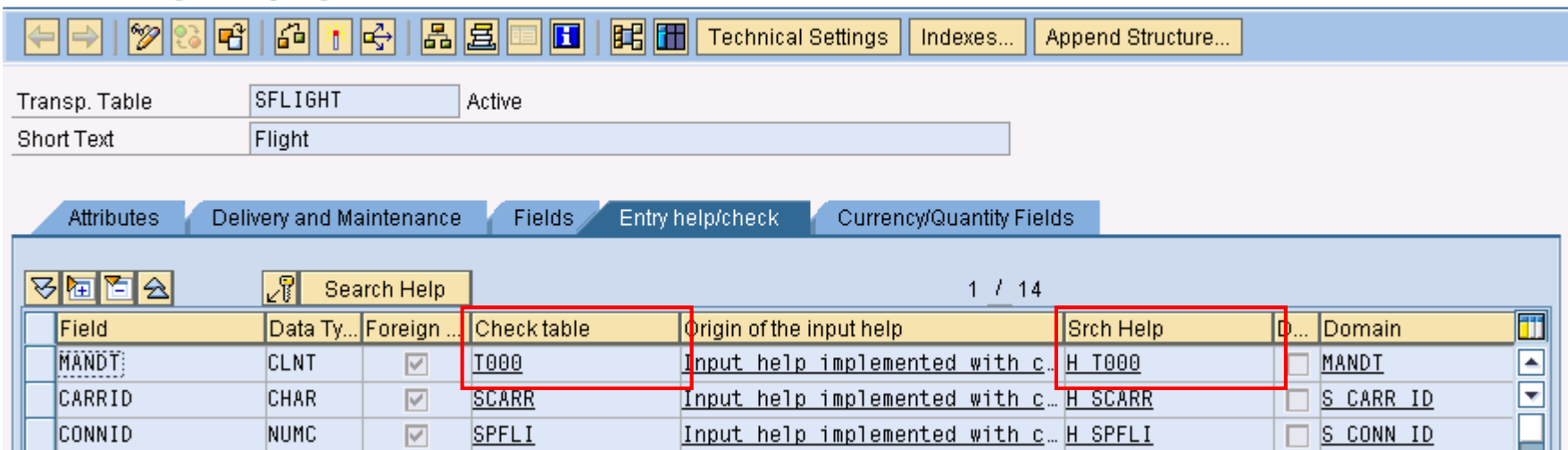
- Entry help uses a table in the background
- Predefined entry help for a table

## ABAP Dictionary: Initial Screen



The screenshot shows the 'ABAP Dictionary: Initial Screen'. At the top is a toolbar with icons for various dictionary objects. Below the toolbar, there are two radio buttons: 'Database table' (selected) and 'View'. To the right of these buttons is a text field containing 'SFLIGHT'. A red rectangle highlights a small icon (a circle with a square inside) located to the right of the 'SFLIGHT' text field.

## Dictionary: Display Table



The screenshot shows the 'Dictionary: Display Table' screen. At the top is a toolbar with icons and buttons for 'Technical Settings', 'Indexes...', and 'Append Structure...'. Below the toolbar, there are two text fields: 'Transp. Table' containing 'SFLIGHT' and 'Active', and 'Short Text' containing 'Flight'. Below these fields are five tabs: 'Attributes', 'Delivery and Maintenance', 'Fields', 'Entry help/check', and 'Currency/Quantity Fields'. The 'Entry help/check' tab is selected. Below the tabs is a table with the following columns: 'Field', 'Data Ty...', 'Foreign ...', 'Check table', 'Origin of the input help', 'Srch Help', 'D...', and 'Domain'. The table contains three rows of data. The first row is highlighted with a red rectangle. The second row is also highlighted with a red rectangle. The third row is highlighted with a red rectangle. The table is titled '1 / 14'.

Field	Data Ty...	Foreign ...	Check table	Origin of the input help	Srch Help	D...	Domain
MANDT	CLNT	✓	T000	Input help implemented with c...	H T000	□	MANDT
CARRID	CHAR	✓	SCARR	Input help implemented with c...	H SCARR	□	S CARR ID
CONNID	NUMC	✓	SPFLI	Input help implemented with c...	H SPFLI	□	S CONN ID