Web Dynpro for ABAP: Tutorial 4 - Display Bookings of Selected Flight

SAP NetWeaver 2004s

Copyright

© Copyright 2005 SAP AG. 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 AG. The information contained herein may be changed without prior notice.

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

Microsoft, Windows, Outlook, and PowerPoint are registered trademarks of Microsoft Corporation.

IBM, DB2, DB2 Universal Database, OS/2, Parallel Sysplex, MVS/ESA, AIX, S/390, AS/400, OS/390, OS/400, iSeries, pSeries, xSeries, zSeries, z/OS, AFP, Intelligent Miner, WebSphere, Netfinity, Tivoli, and Informix are trademarks or registered trademarks of IBM Corporation in the United States and/or other countries.

Oracle is a registered trademark of Oracle Corporation.

UNIX, X/Open, OSF/1, and Motif are registered trademarks of the Open Group.

Citrix, ICA, Program Neighborhood, MetaFrame, WinFrame, VideoFrame, and MultiWin are trademarks or registered trademarks of Citrix Systems, Inc.

HTML, XML, XHTML and W3C are trademarks or registered trademarks of W3C®, World Wide Web Consortium, Massachusetts Institute of Technology.

Java is a registered trademark of Sun Microsystems, Inc.

JavaScript is a registered trademark of Sun Microsystems, Inc., used under license for technology invented and implemented by Netscape.

MaxDB is a trademark of MySQL AB, Sweden.

SAP, R/3, mySAP, mySAP.com, xApps, xApp, SAP NetWeaver, and other SAP products and services mentioned herein as well as their respective logos are trademarks or registered trademarks of SAP AG in Germany and in several other countries all over the world. All other product and service names mentioned are the trademarks of their respective companies. Data contained in this document serves informational purposes only. National product specifications may vary.

These materials are subject to change without notice. These materials are provided by SAP AG and its affiliated companies ("SAP Group") for informational purposes only, without representation or warranty of any kind, and SAP Group shall not be liable for errors or omissions with respect to the materials. The only warranties for SAP Group 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.

Icons in Body Text

Icon	Meaning
	Caution
	Example
	Note
	Recommendation
	Syntax

Additional icons are used in SAP Library documentation to help you identify different types of information at a glance. For more information, see *Help on Help → General Information Classes and Information Classes for Business Information Warehouse* on the first page of any version of *SAP Library*.

Typographic Conventions

Type Style	Description
Example text	Words or characters quoted from the screen. These include field names, screen titles, pushbuttons labels, menu names, menu paths, and menu options.
	Cross-references to other documentation.
Example text	Emphasized words or phrases in body text, graphic titles, and table titles.
EXAMPLE TEXT	Technical names of system objects. These include report names, program names, transaction codes, table names, and key concepts of a programming language when they are surrounded by body text, for example, SELECT and INCLUDE.
`Example text`	Output on the screen. This includes file and directory names and their paths, messages, names of variables and parameters, source text, and names of installation, upgrade and database tools.
`Example text`	Exact user entry. These are words or characters that you enter in the system exactly as they appear in the documentation.
`<Example text>`	Variable user entry. Angle brackets indicate that you replace these words and characters with appropriate entries to make entries in the system.
`EXAMPLE TEXT`	Keys on the keyboard, for example, `F2` or `ENTER`.

Web Dynpro for ABAP: Tutorial 4 – Display Bookings of Selected Flight 5
Development Objectives 5
Procedure 5
Create the Web Dynpro component ZZ_00_BOOKINGS 6
Create sub node BOOKINGS and the corresponding Supply Function GET_BOOKINGS 6
Implement the Supply method GET_BOOKINGS 9
Update the Mapping between View Context and Component Controller Context 10
Add a new table UI element in View Layout and Bind it 11
Activation, Creation of a Web Dynpro Application and Execution 13
Result 13
SAP Online Help 13

Web Dynpro for ABAP: Tutorial 4 – Display Bookings of Selected Flight

Development Objectives

In this tutorial, the BAPINAV component from Web Dynpro for ABAP - BAPI Navigation tutorial will be further enhanced to display a list of bookings for a selected flight. The context which contains the flights information will be enhanced with a sub node for the corresponding bookings. We will also show how the booking information is retrieved in a so called supply function and the context node bound to a new table UI element in the Main View.

Procedure

- Copy the Web Dynpro component from Tutorial 3 or copy the component ZZ_00_BAPINAV to a new component with name *ZZ_00_BOOKINGS.* Adjust the name of the window according to the component name.

- Enhance the FLIGHT_LIST node in the component controller context with a sub node BOOKINGS for the bookings, based on the dictionary structure SBOOK with cardinality 0..n and supply function GET_BOOKINGS.

- Select the following attribute from the structure SBOOK: CARRID, CONNID, FLDATE, BOOKID, CUSTOMID, CUSTTYPE, CLASS, ORDER_DATE, PASSNAME.

- Implement supply method GET_BOOKINGS which fills the node BOOKINGS with booking information, according to the selected flight. Use the static GET_BOOKINGS method of the class CL_WDABAP_FLIGHT_MODEL.

- Update the mapping of the FLIGHT_LIST node between the component controller context and the view context of FLIGHTLISTVIEW.

- Add a new table UI element to the layout of the view FLIGHTLISTVIEW and bind it to the node BOOKINGS of the view context.

- Activation, Creation of a Web Dynpro Application and Execution. Select different flights.

Create the Web Dynpro component ZZ_00_BOOKINGS

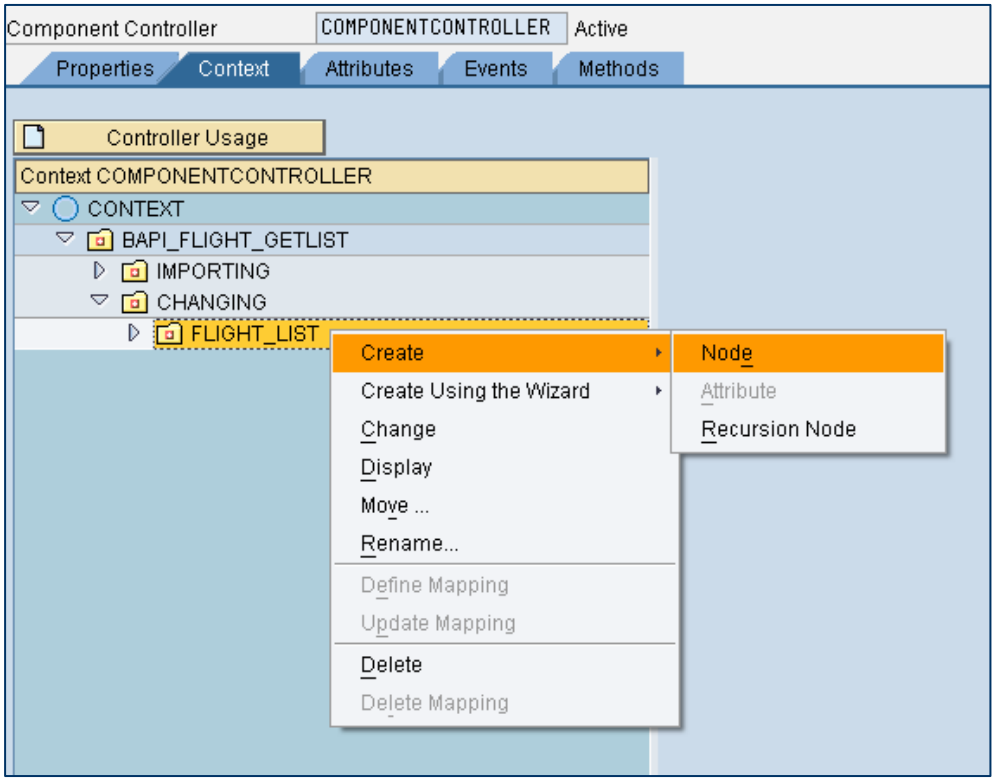
Procedure

1. Copy the Web Dynpro component from Tutorial 3 or copy the component ZZ_00_BAPINAV to a new component with name *ZZ_00_BOOKINGS.*

Create sub node BOOKINGS and the corresponding Supply Function GET_BOOKINGS

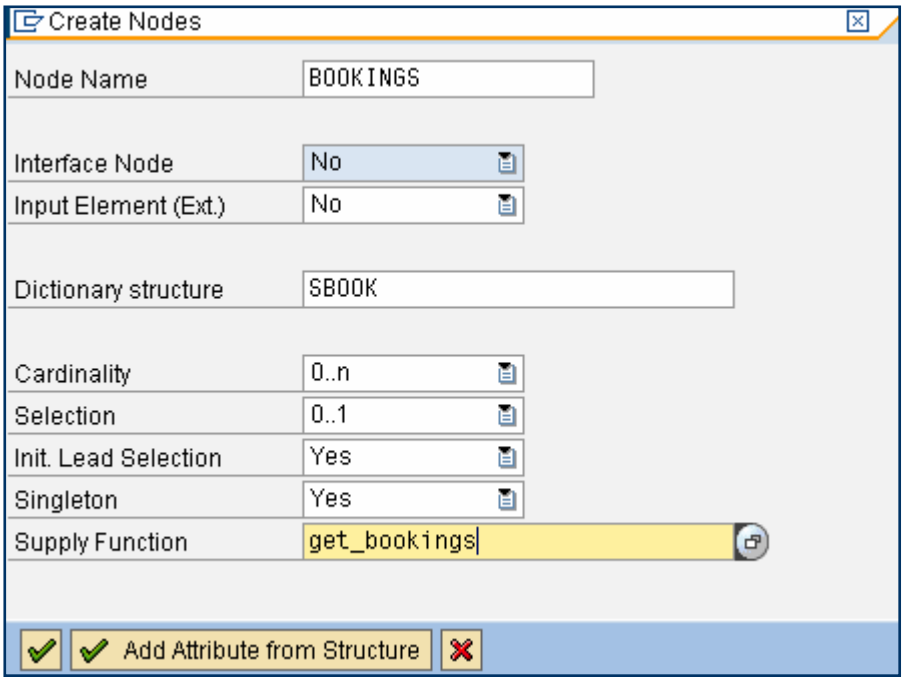
Procedure

1. Here you will create a new sub node BOOKINGS. Go to the context view of the COMPONENTCONTROLLER and add a new sub node **within** the FLIGHT_LIST node.

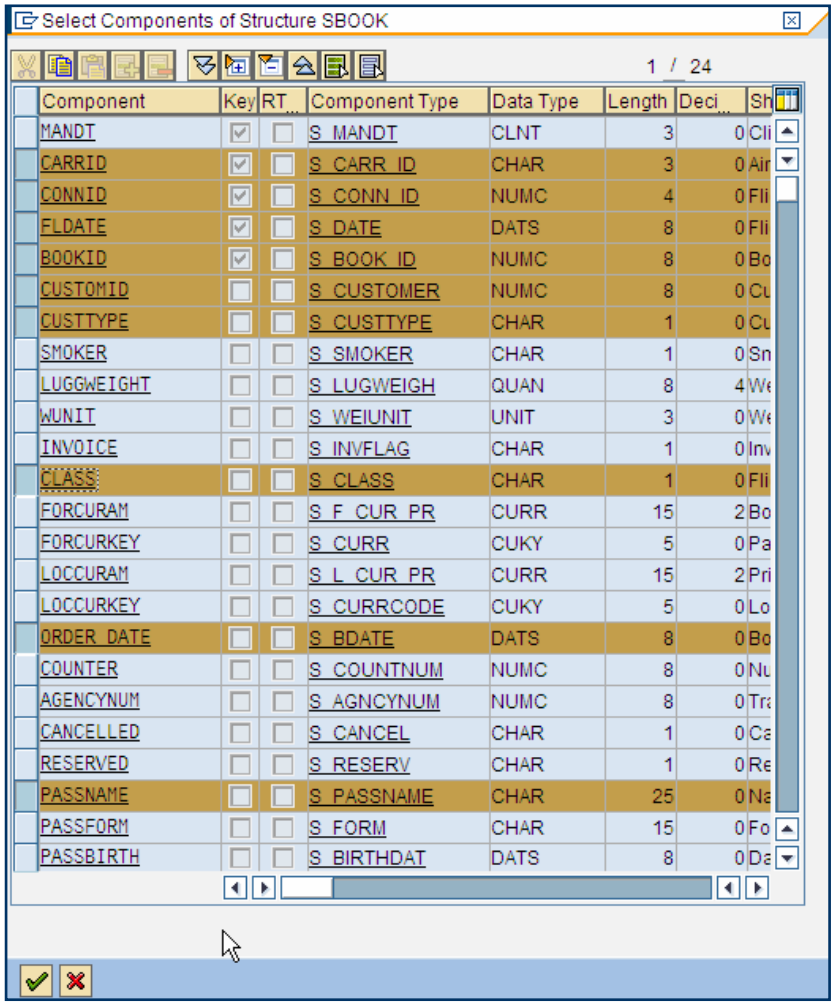


It is important that the new node is a sub node of the FLIGHT_LIST node and not a node at the same level.

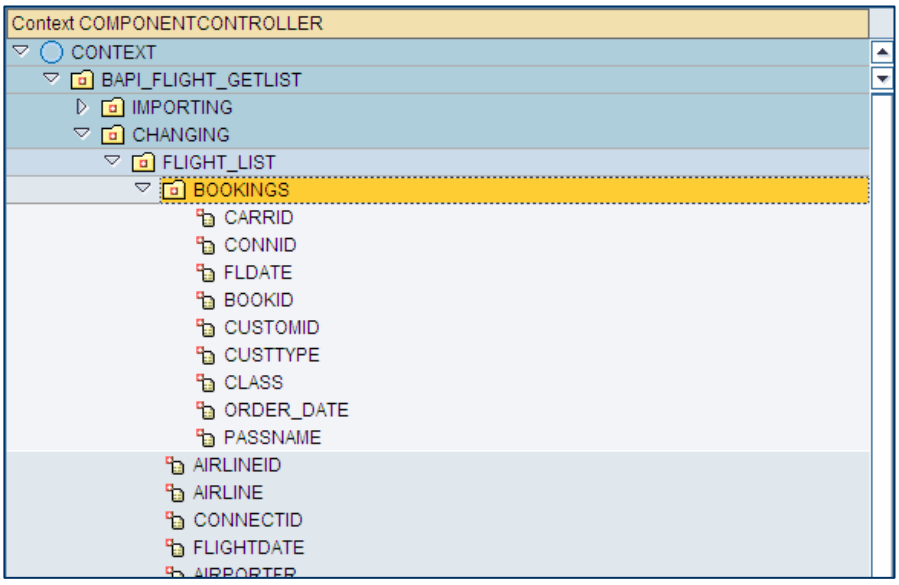
2. Name the new node *BOOKINGS* and select *0..n* for the cardinality of the new node. Choose *SBOOK* as dictionary structure and enter *get_bookings* as name for the supply function and press *Add Attribute from Structure*.



3. Select CARRID, CONNID, FLDATE, BOOKID, CUSTOMID, CUSTTYPE, CLASS, ORDER_DATE, PASSNAME, and confirm.



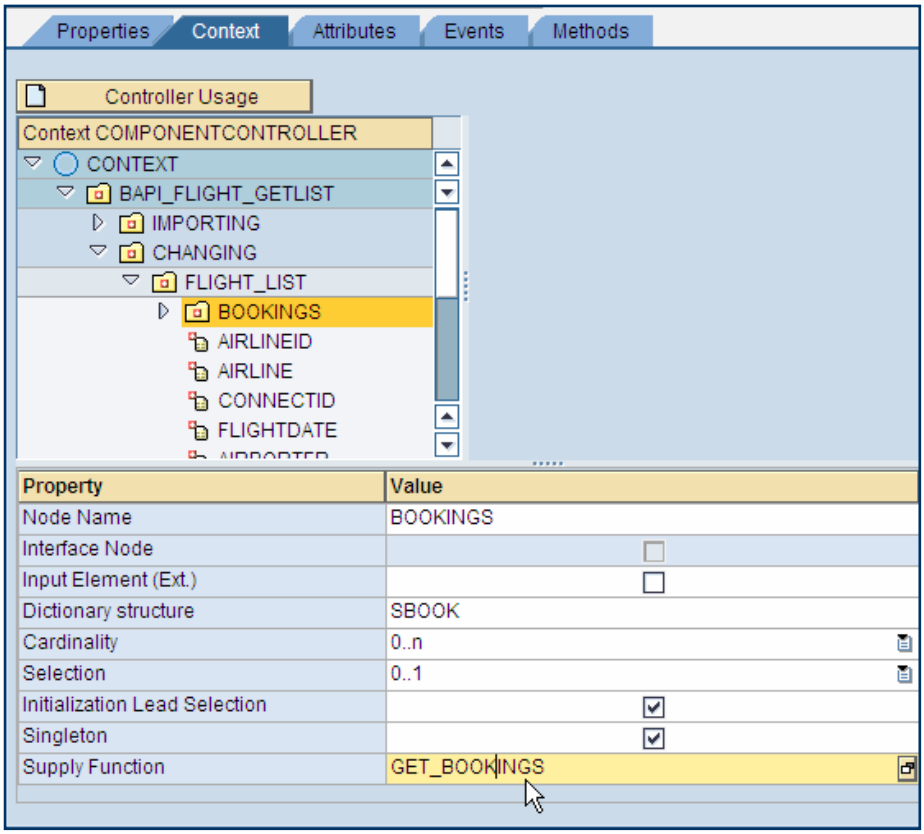
4. The new node BOOKINGS should now be a sub node of the node FLIGHT_LIST with the selected attributes:



Implement the Supply method GET_BOOKINGS

Procedure

1. Double click at the supply function *GET_BOOKINGS* at the properties panel of the newly created BOOKINGS node.



This will lead you to the implementation of the GET_BOOKINGS method. Here the node BOOKINGS will be filled with bookings data dependent of the current parent node's lead selection (the specific flight selected in the visible table at the screen).
Enhance the empty method GET_BOOKINGS (you may delete the comment lines) with coding that fills the current node. First the current data of the parent node can be retrieved via the parent_element object.

2. For retrieving the bookings data use the static method of the CL_WDABAP_FLIGHT_MODEL class.:

```
method GET_BOOKINGS .
  data:
    Itab_Bookings   type IF_COMPONENTCONTROLLER=>Elements_Bookings,
    Stru_Bookings   like line of Itab_Bookings,
    Stru_FlightList type if_componentcontroller=>Element_FLIGHT_LIST.

  parent_element->get_static_attributes(
         importing static_attributes = Stru_FlightList ).

  Itab_bookings =  CL_WDABAP_FLIGHT_MODEL=>GET_BOOKINGS(
      CARRID   = Stru_FlightList-airlineid
      CONNID   = Stru_FlightList-connectid
      FLDATE   = Stru_FlightList-flightdate
    ).
  node->bind_table( Itab_bookings ).
endmethod.
```

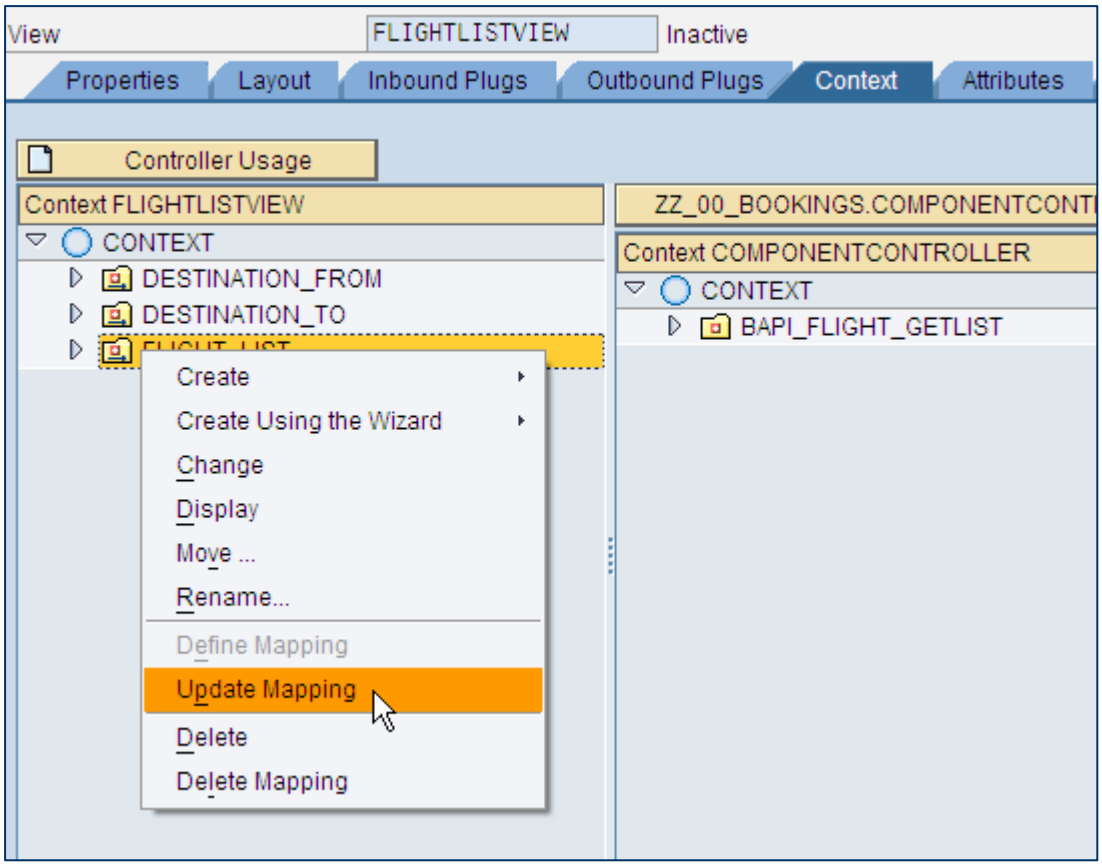
Update the Mapping between View Context and Component Controller Context

Procedure

Because the node FLIGHT_LIST of the component controller has changed you have to update the mapping to the mapped FLIGHT_LIST node of the view controller of FLIGHTLISTVIEW.

1. Go to the tab *Context* of the view FLIGHTLISTVIEW and select *Update Mapping* for the node FLIGHT_LIST by right click (context menu):

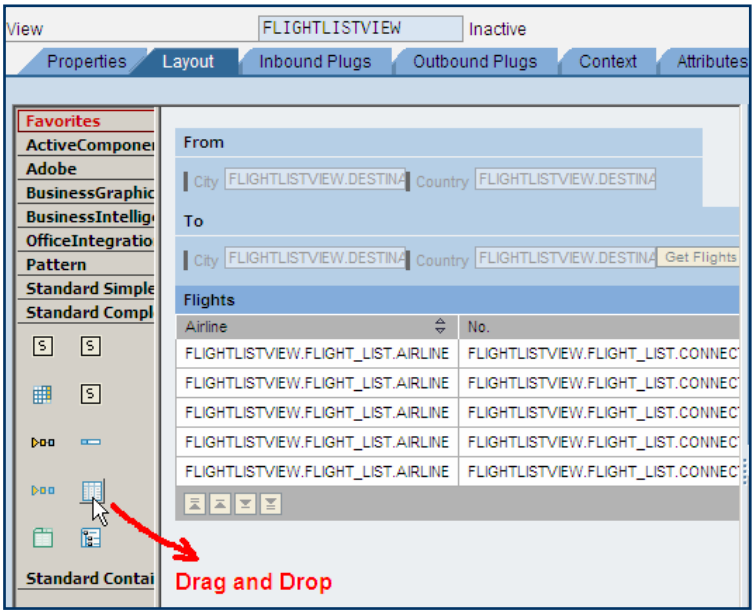
2. Confirm the following dialog and check, that the BOOKINGS node is now also available as sub node of the node FLIGHT_LIST.



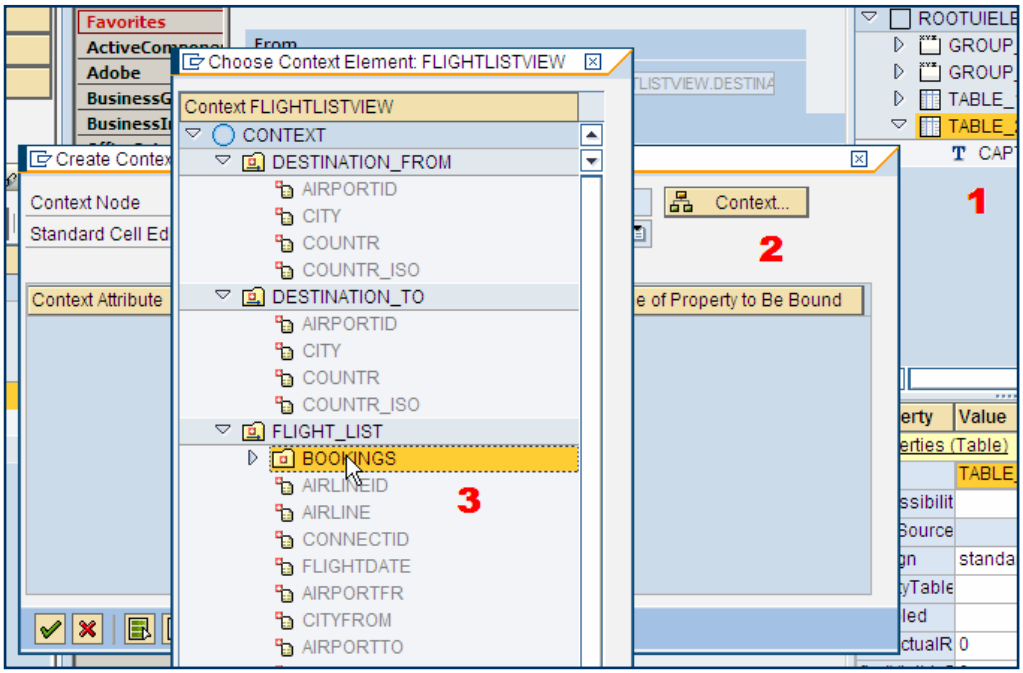
Add a new table UI element in View Layout and Bind it

Procedure

1. Go to the tab *Layout* of the FLIGHTLISTVIEW and add a second table UI element from the **Standard Complex** UI elements section to the view.



2. To bind the new table UI element to the new BOOKINGS sub node right click on the TABLE_2 node in the hierarchical view at the left hand side (**1**) and choose *Context…* at the appearing dialog box (**2**) to select the BOOKINGS node (**3**).



In the appearing dialog box the visible columns of the table can be chosen. You may leave the default selection of all columns and confirm.

Getting Error:

If you are not using the NW 2004s ABAP sneak preview edition, you might get the following error on NW2004s Systems.

The type CL_WDABAP_FLIGHT_MODEL=>get_bookings is unknown or not found.

Class WDABAP_FLIGHT_MODEL is only provided in our NW 2004s ABAP sneak preview systems.

Please do the following if you encounter the error above.

- Simply create a ZCL_WDABAP_FLIGHT_MODEL class in your *get_bookings* method.

Its purpose is only to select from the SBOOK table.

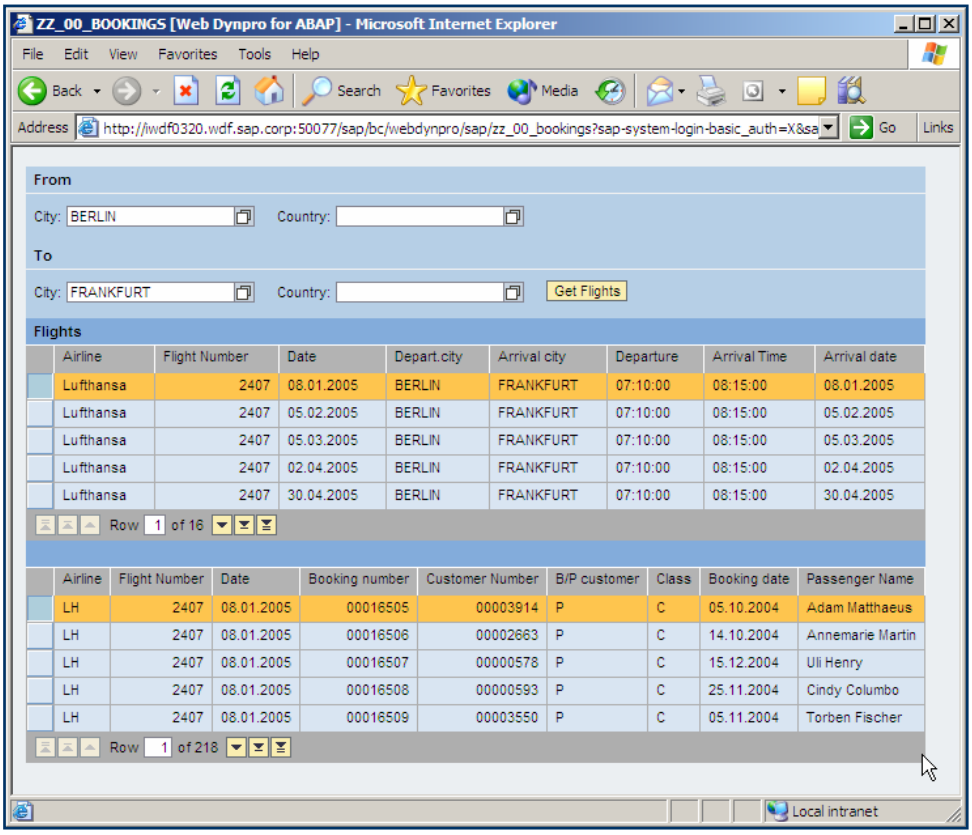
```
select * from sbook into table bookings where
carrid = carrid
connid= connid
fldate = fldate.
```

->>*bookings* is the return parameter of type *TY_BOOKINGS*

Activation, Creation of a Web Dynpro Application and Execution

1. Activate **all objects** of Web Dynpro component *ZZ_00_BOOKINGS*.
2. Create the Web Dynpro application ZZ_00_BOOKINGS and assign it to package $TMP (local object).
3. Run your application.

The result should look like this:



Result

As a result of this tutorial you were able to understand the supply function concepts and learned to how to display data in nested tables within a Web Dynpro application.

SAP Online Help

More information on Web Dynpro for ABAP can be found at the SAP Help Portal under the short link http://help.sap.com/saphelp_nw04s/helpdata/en/77/3545415ea6f523e10000000a155106/frameset.htm or via path help.sap.com → Documentation → SAP NetWeaver → SAP NetWeaver 2004s → English → SAP NetWeaver Library → SAP NetWeaver by Key Capability → Application Platform by Key Capability → ABAP Technology → UI Technology → Web UI Technology → Web Dynpro for ABAP.