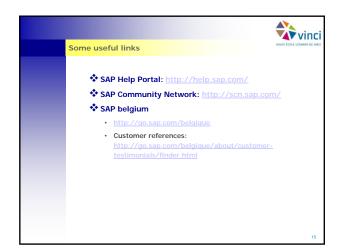
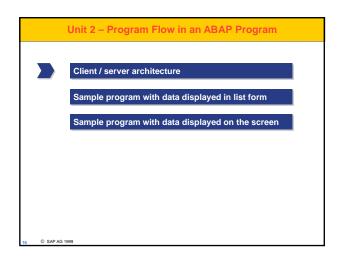
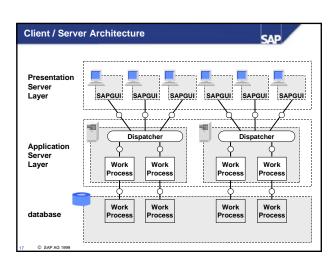




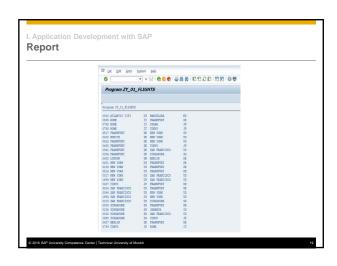
SAP - Brief History * 1972: SAP is founded 1973 SAP R/1 solutions launched (Systemanalyse und Programmentwicklung) * 1979: SAP R/2 solutions launched * 1992: SAP R/3 solutions launched * 1996: SAP R/3 solutions launched * 2001: SAP acquires Top Tier and forms SAP Portals * 2002: SAP acquires TopManage Software Solutions for its small-to mid-sized business offering. * 2003: Launch of SAP NetWeaver platform * 2004: SAP announces a timeline for the service-enablement of its own solutions. By 2007 all SAP solutions will be Enterprise Services Architecture * 2008: SAP acquired Business Objects. * 2011: SAP acquired Sybase and launched SAP HANA * 2013: SAP Business Suite on SAP HANA, hybris acquisition * 2014: acquisition of Concur

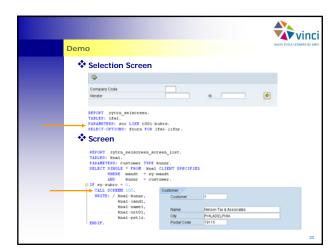


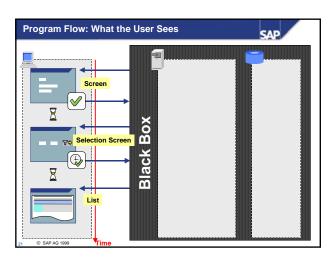


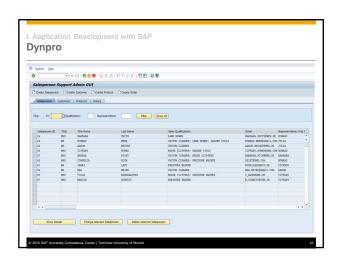


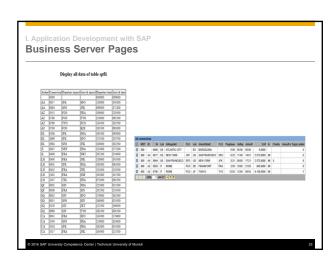
Provide						
Type	Description	Programming languages				
Report	Executable programs Processing and displaying of data	ABAP				
Dynpro	Application programs with user interface within SAP GUI Handling user interaction with SAP GUI	ABAP				
Business Server Pages	Web-based applications Creating HTML pages with ABAP	HTML, HTMLB, ABAP				
Web Dynpro	Web-based applications Creating web pages solely with ABAP	ABAP				
SAPUI5 .	Web and mobile applications Focus on increased usability	HTML5, JavaScript,				

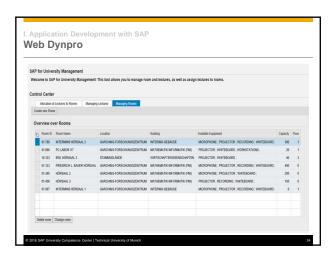


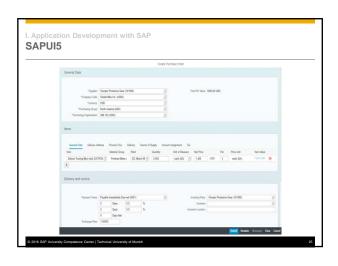


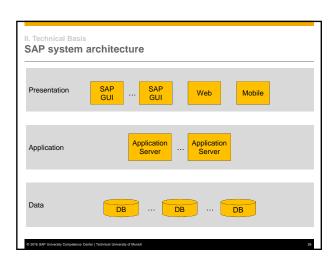


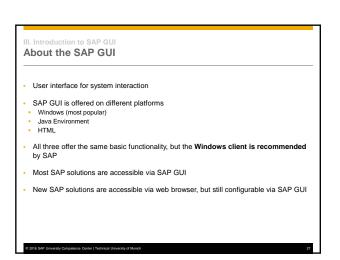


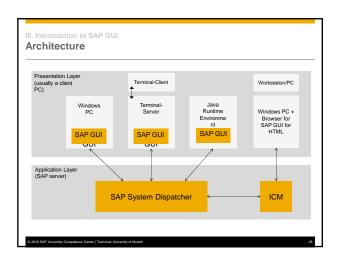


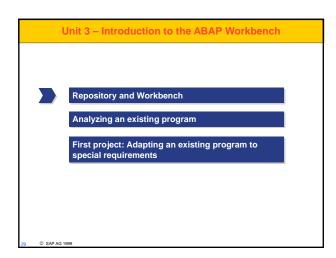


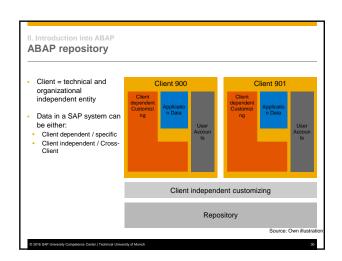


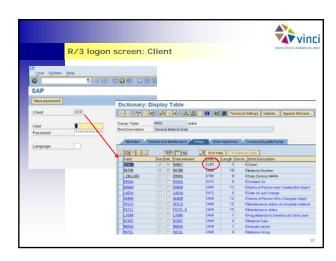


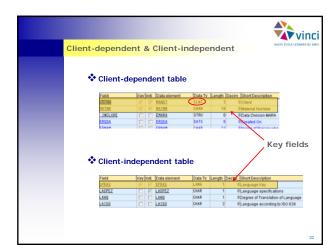


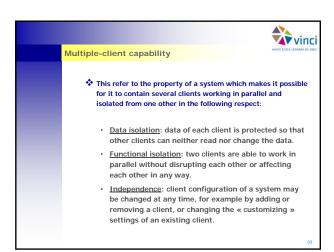












ABAP Advanced Business Application Program is designed for: Business tasks Mass data processing Currency Multilingual capabilities User dialog development Database access Open SQL Platform independence ABAP programs are portable Development in teams

Software development cycle

The ABAP Development Workbench supports the entire software development cycle with tools for:

Programming in the ABAP language

Efficient debugging

Static and dynamic tests of different scales

Performance analyses

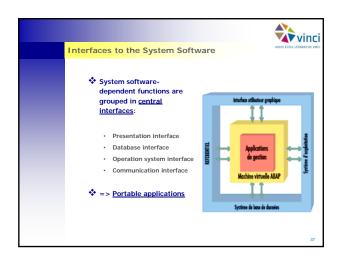
Organizing large projects

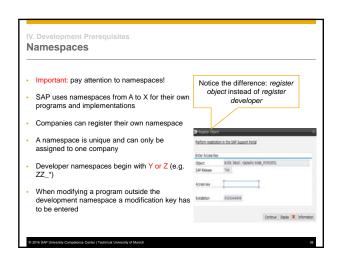
Updates and importing patches

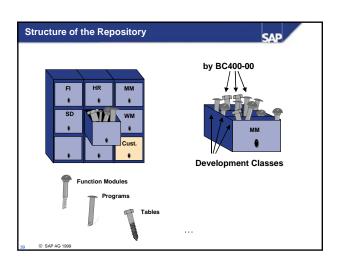
Source code management

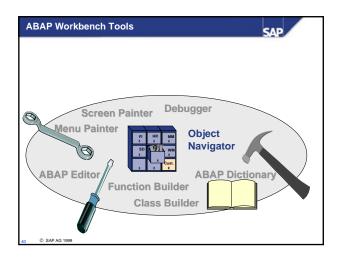
Connectors to Java and .NET world

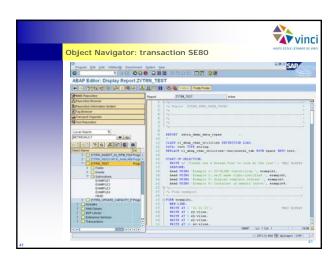
ABAP has the following features * Tailor-made data types for business programming. * Object-oriented exception handling * Support for dynamic programming; from dynamic data generation to dynamic creation of types and entire programs * Simple management of multilingual texts * Integration of the DML part of SQL into the language * Creation of database tables via the definition of metadata * Definition of various types of data structures, which are then available throughout the server * Conversion of XML data into ABAP structures * Simple handling of transactions – that is, program units – which can only be executed fully or not all, in order to ensure data integrity * Possibility to prepare program enhancements so that customers can enhance programs without modification – that is, without the changes being lost after an update or having to be merged manually * Possibility to publish procedures as Web services or access Web services as a user

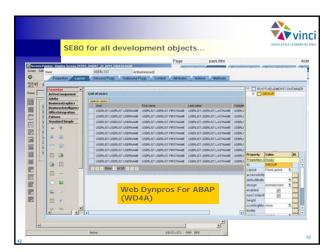


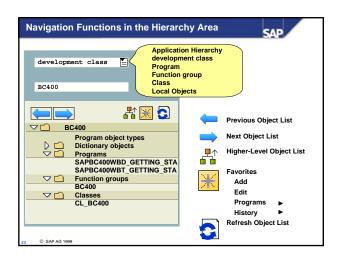


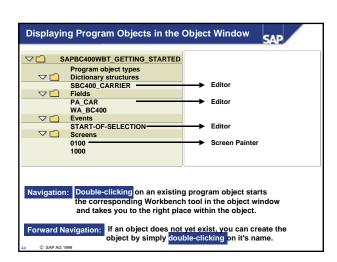


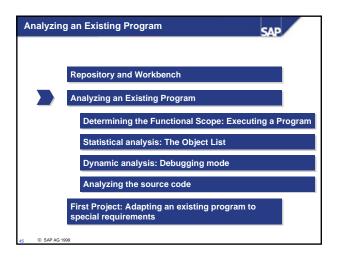


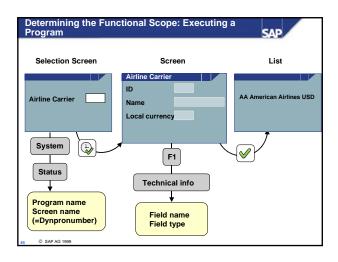


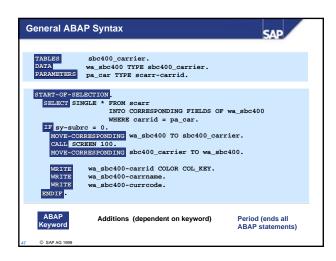


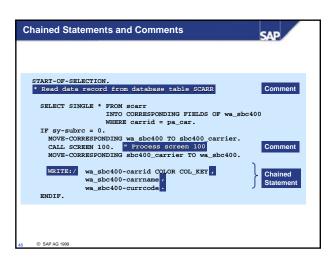


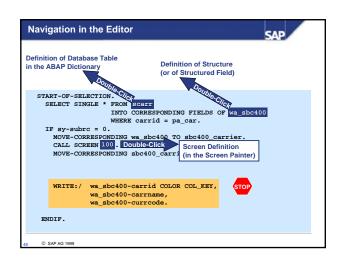


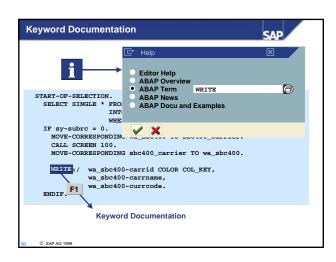


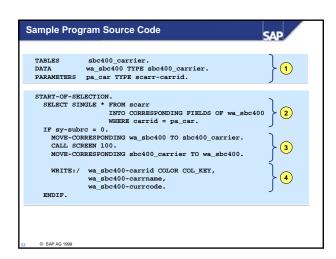


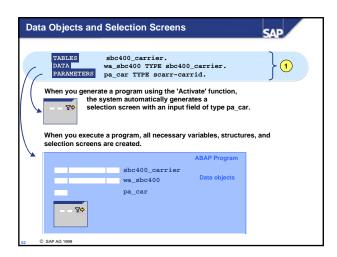


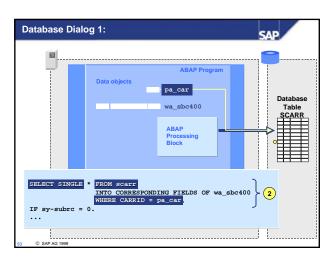


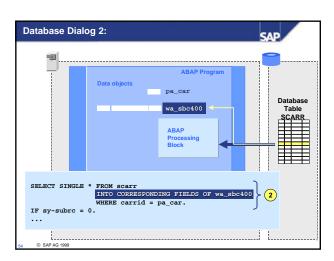


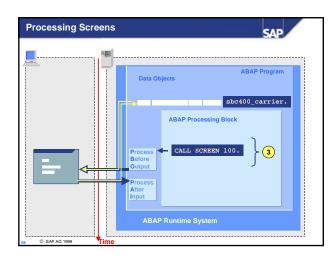


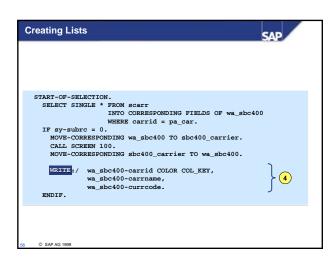


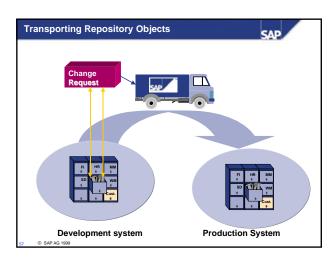


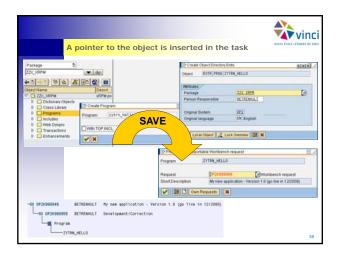




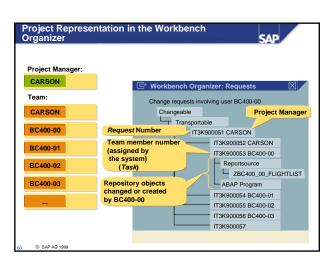


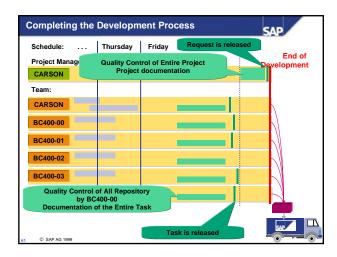


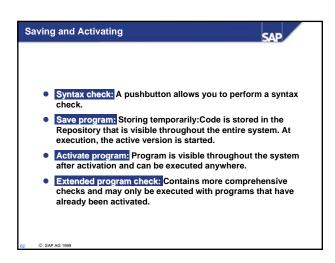


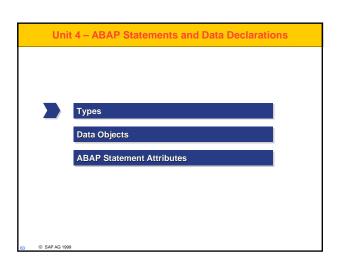


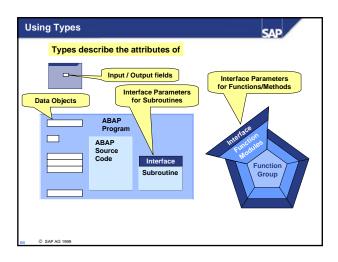
Schedule:	Monday	Tuesday	Wednesday	Thursday	Friday
Project Mana	ager:				
CARSON					
Team:					
CARSON					
BC400-00					
BC400-01					
BC400-02					
BC400-03					
		1			
Employee activities (here: Exercises)			End of Developm		

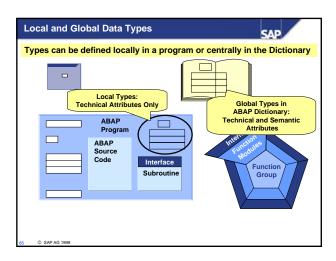


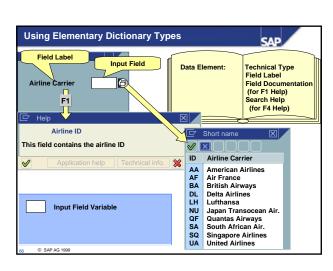


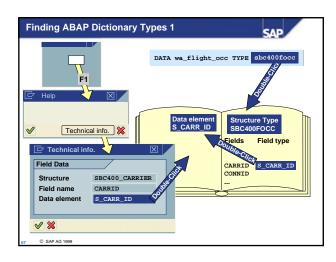


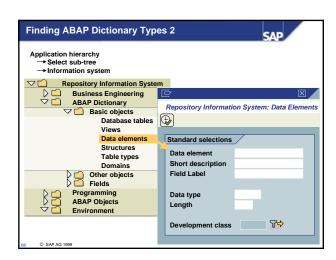


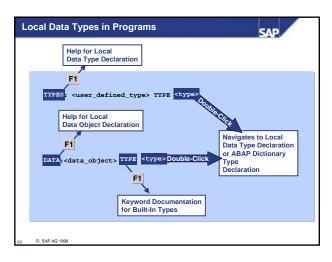


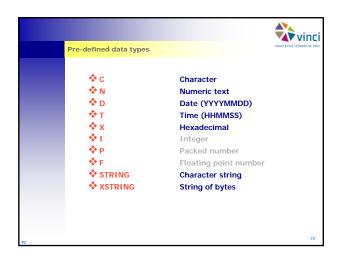


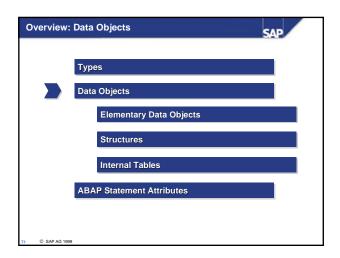


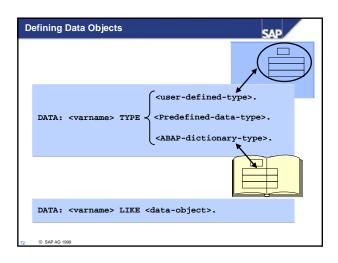


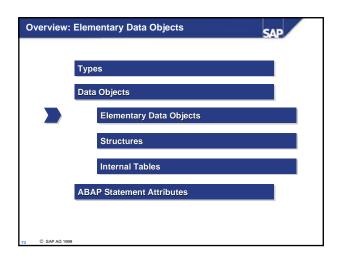


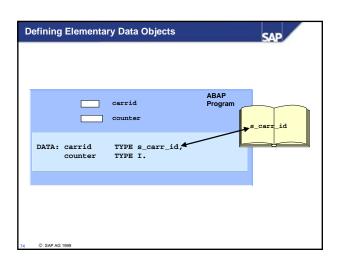


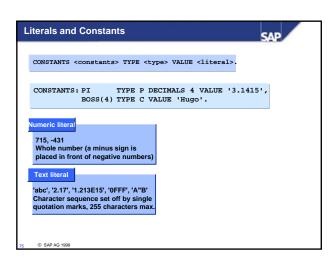


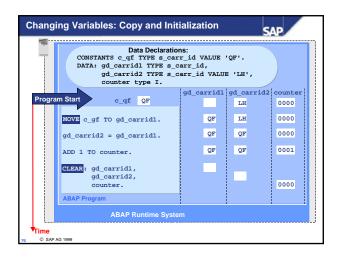


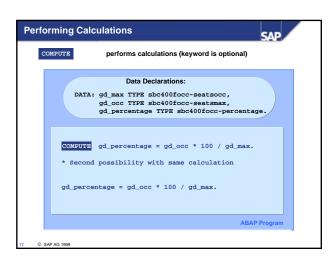


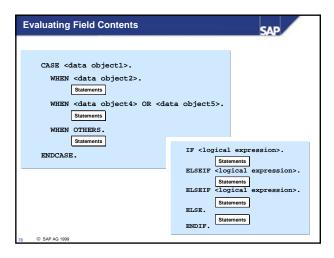


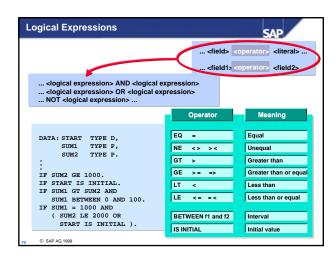


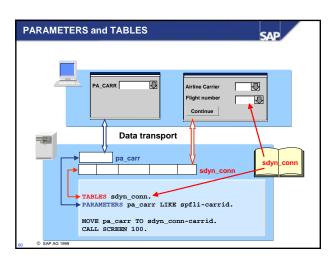


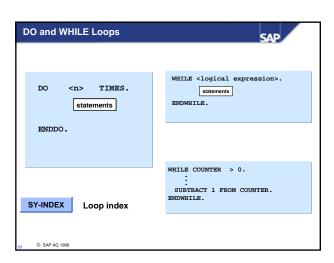


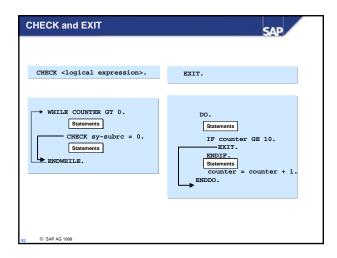


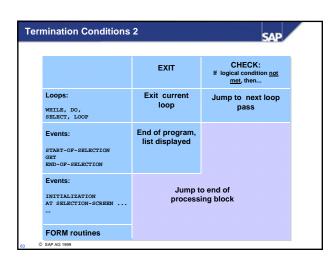


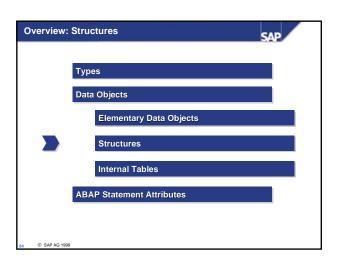


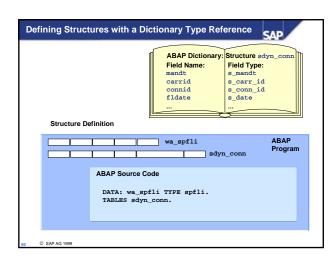


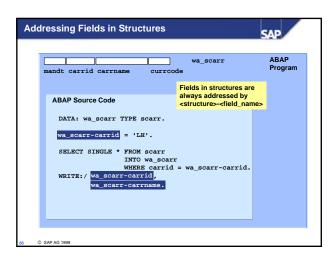


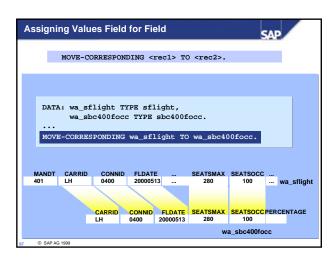


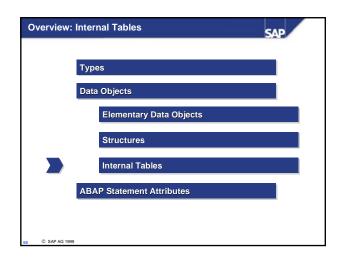


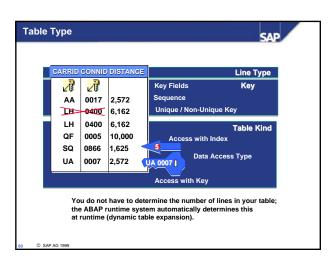


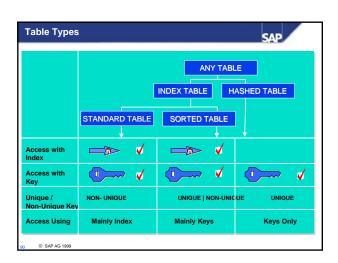


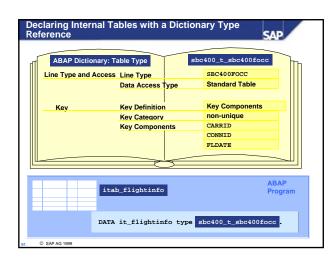


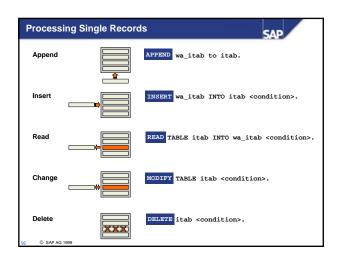


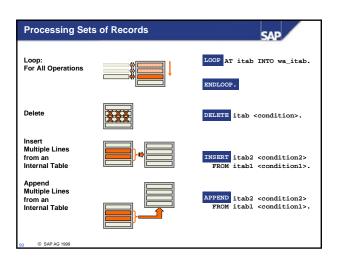


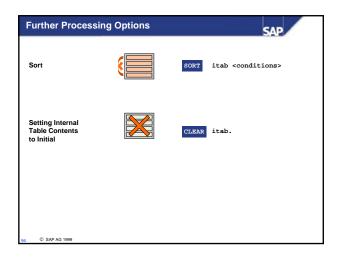


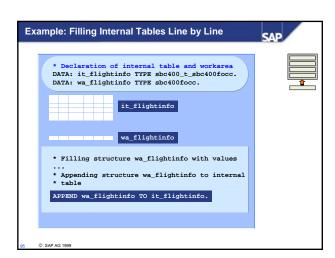


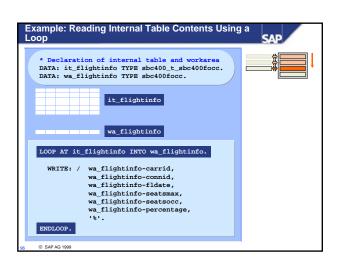


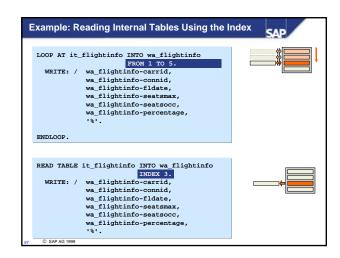


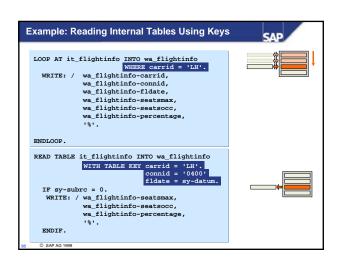


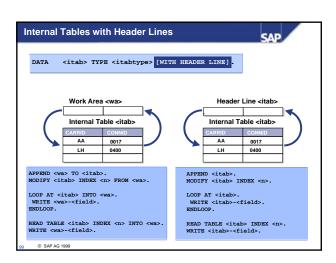


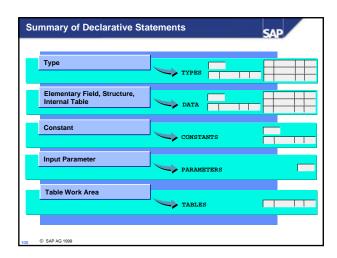


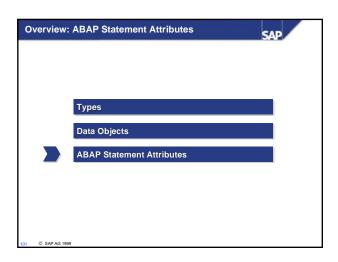


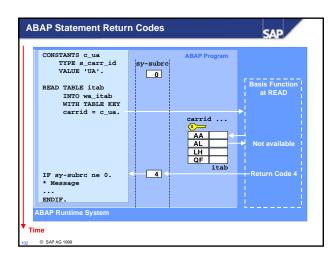


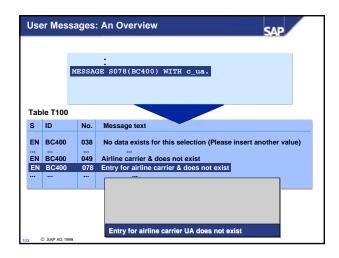


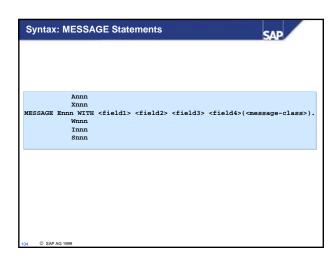


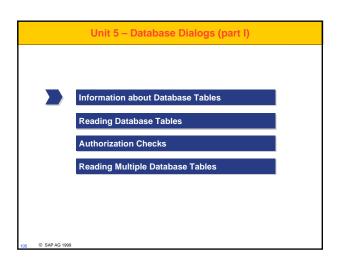




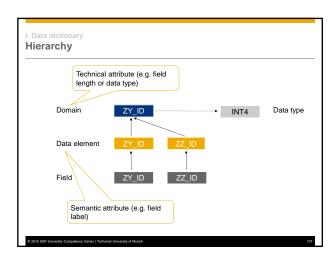


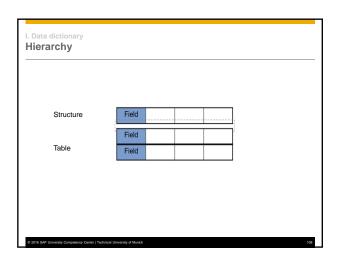




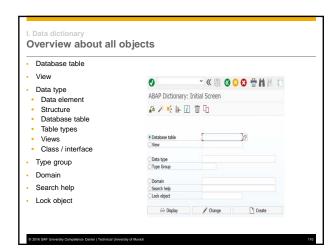


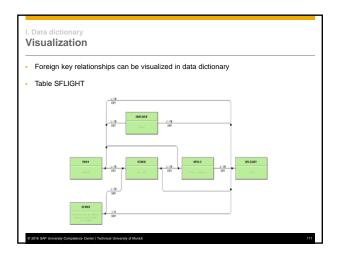
I. Data dictionary What is the data dictionary? Data dictionary = global directory for data types Assignment of help texts and explanations for data types in different languages Entity—relationship model can be shown as a figure automatically Most important objects: structure, table, data element and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

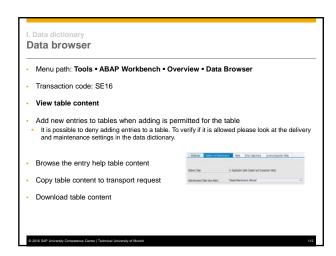


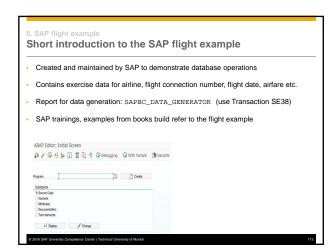


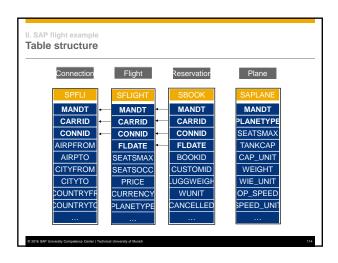
Nenu path: Tools • ABAP Workbench • Development • Dictionary Menu path: Tools • ABAP Workbench • Development • Dictionary Transaction code: SE11 View, edit, delete, create tables, data types, domains definitions etc. Tables and views from the ABAP dictionary represent tables and views from the database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

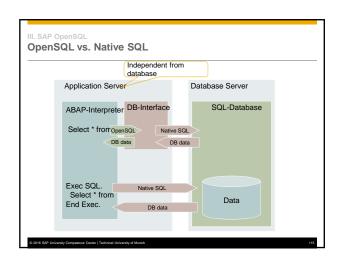


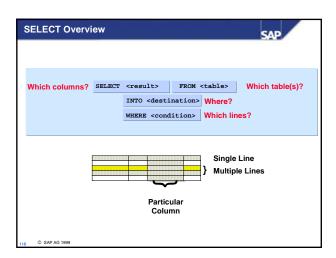


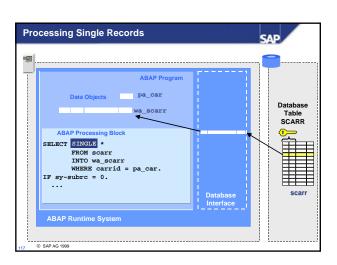


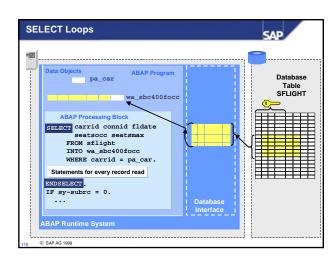


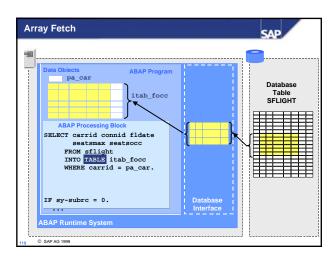


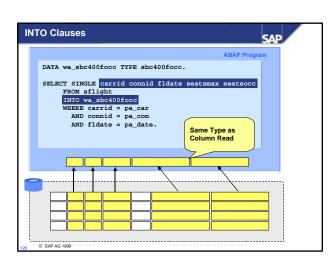


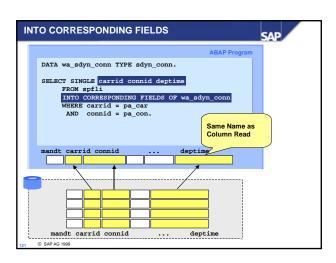




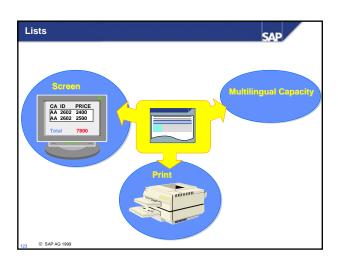


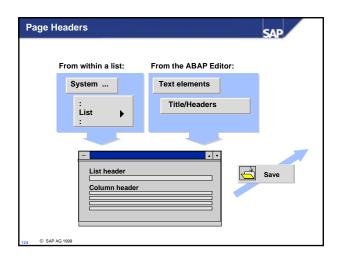


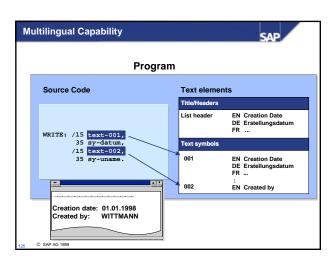


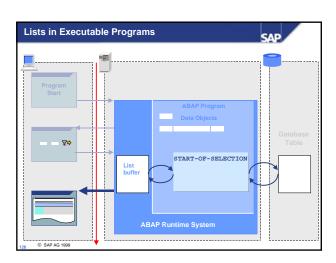


Contents: List attributes and strengths Basic lists List events Interactive lists Example with syntax: Detail lists

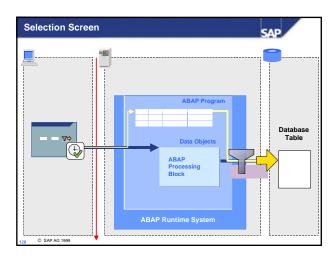


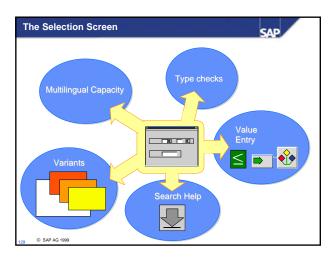


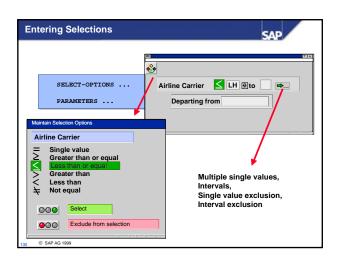


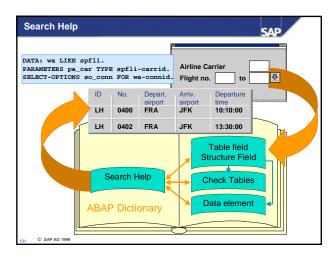


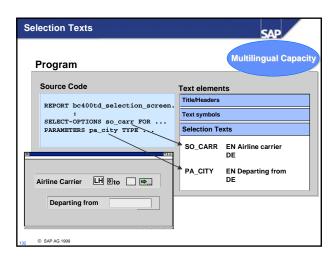
Contents: Selection screen attributes and strengths Defining selection screens Evaluating user input to restrict database selection Selection screen events Example with syntax: Additional input checks with error dialog

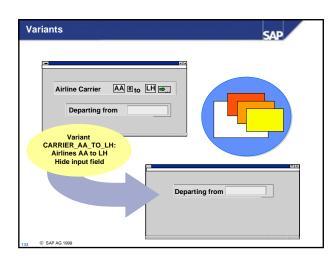


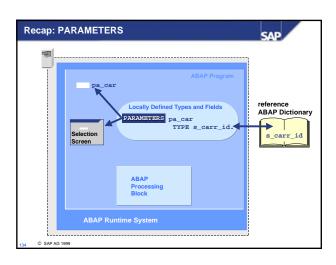


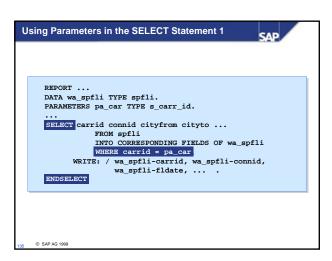


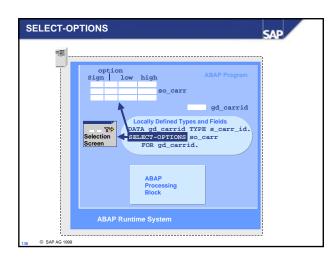


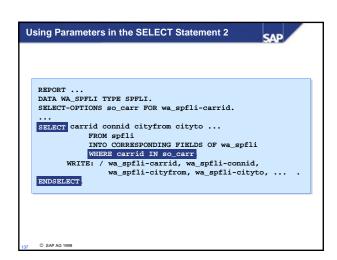


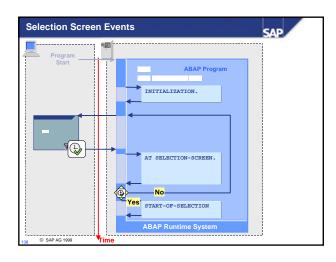


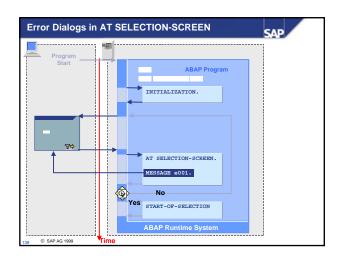


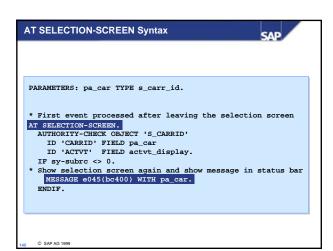




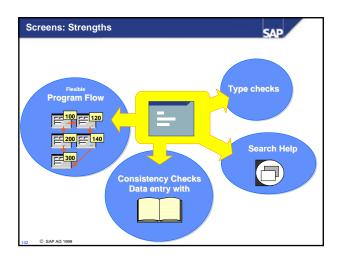


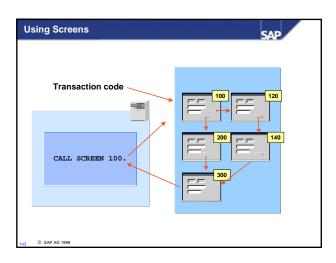


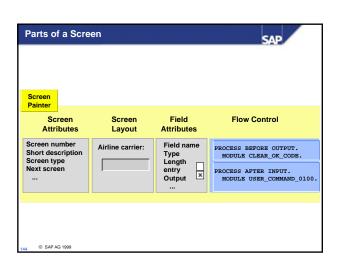


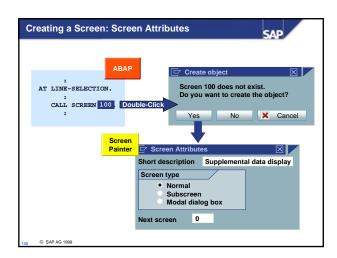


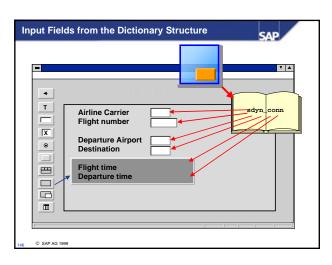
Contents: Screen attributes and strengths Creating screens Layout Field attributes Flow Logic Data transport Using pushbuttons and evaluating user actions

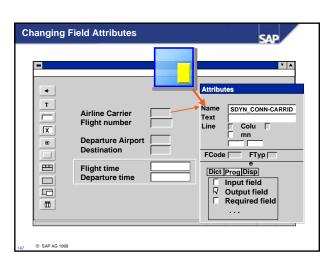


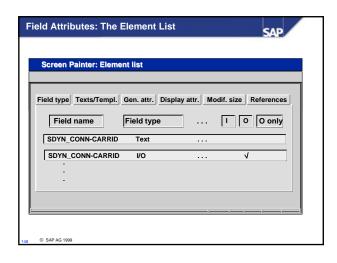


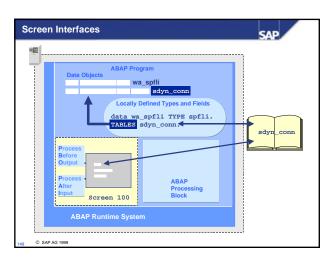


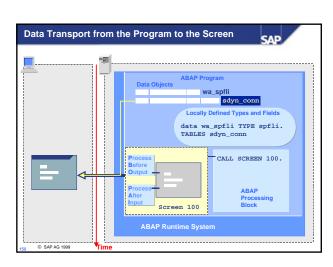


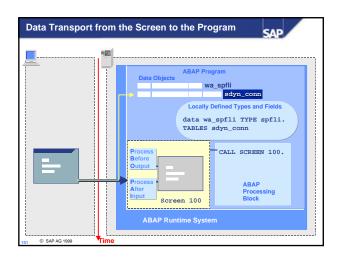


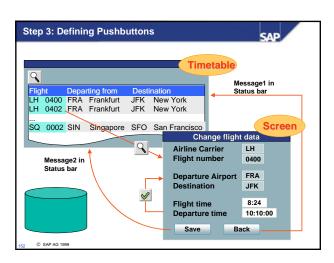


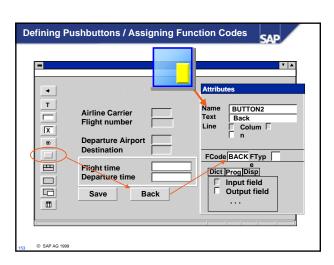


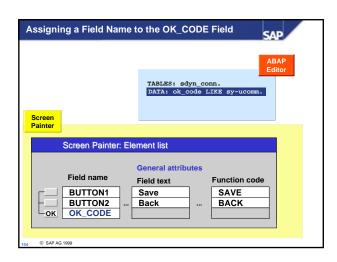


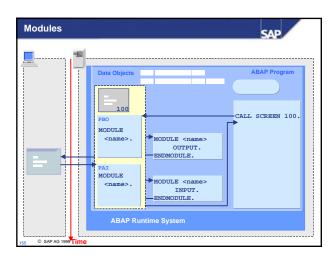


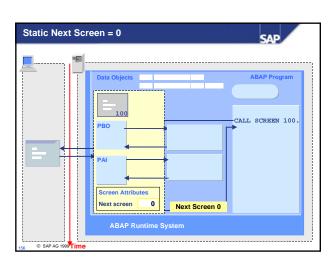


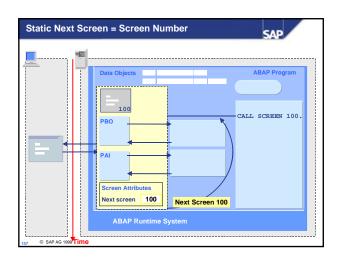


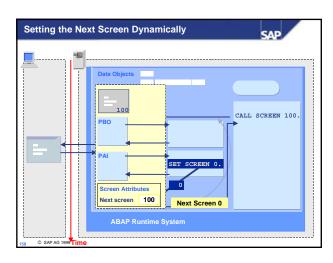


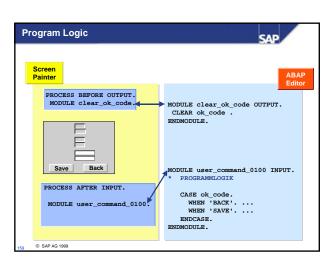


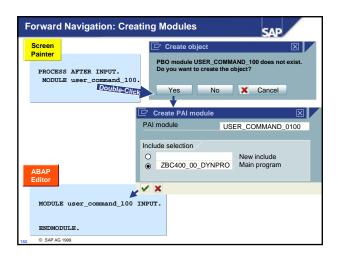


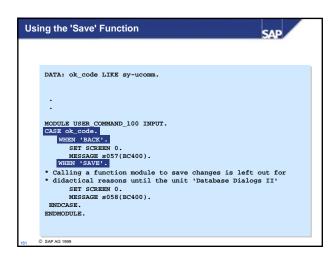


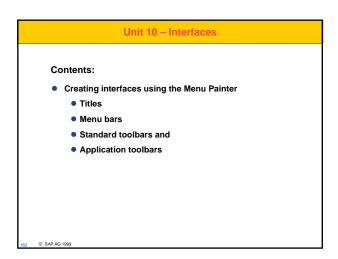


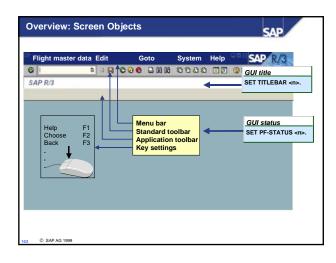


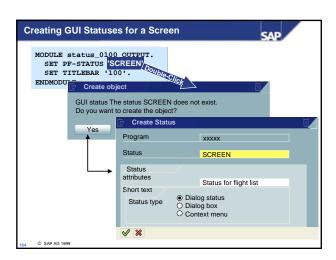


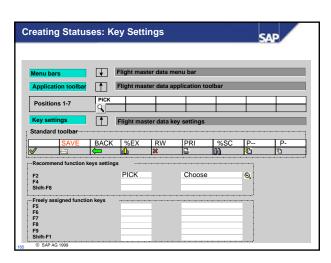


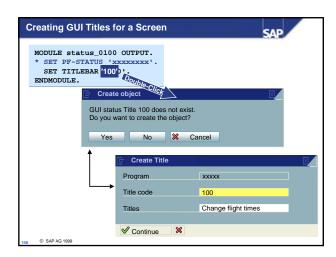


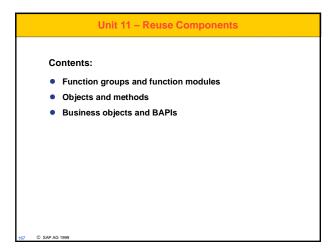


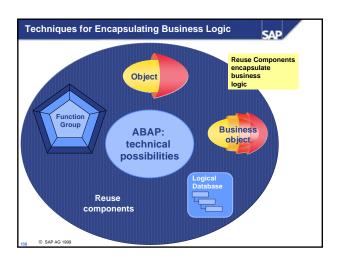


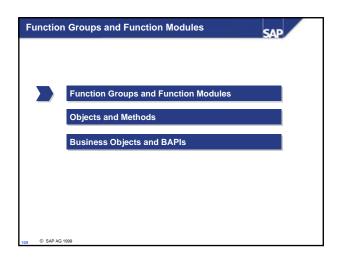


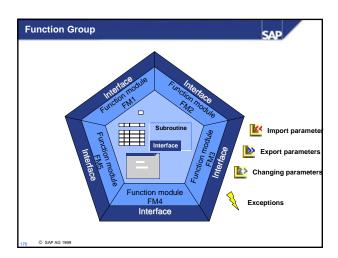


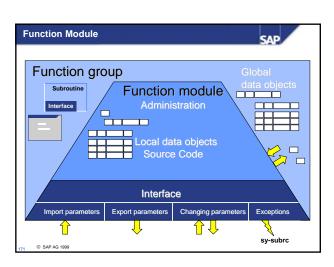


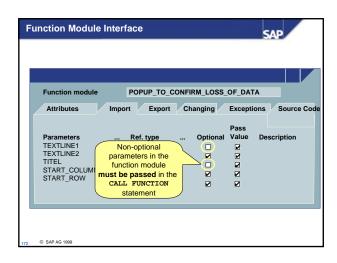


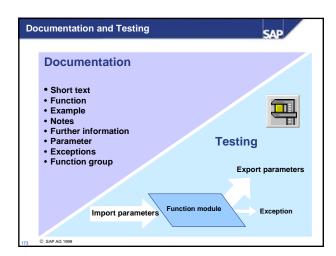


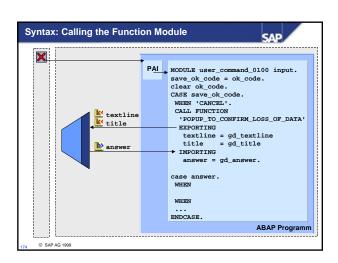


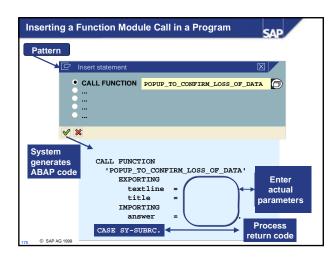


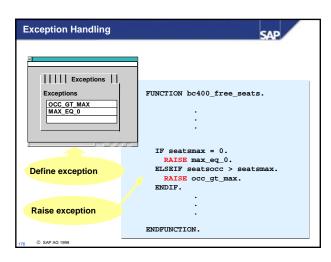


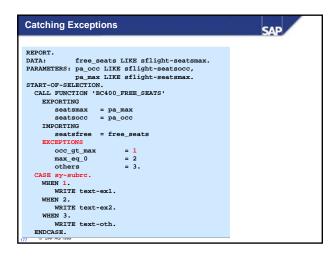


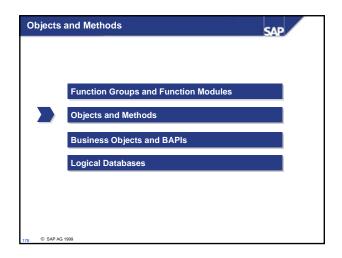


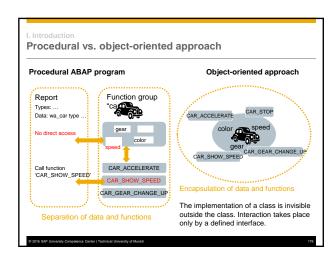


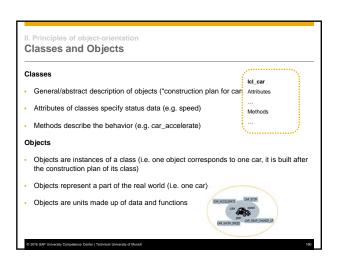


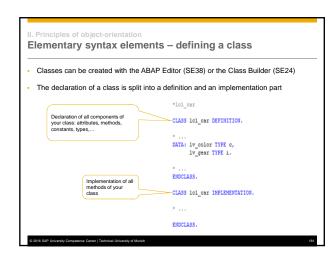


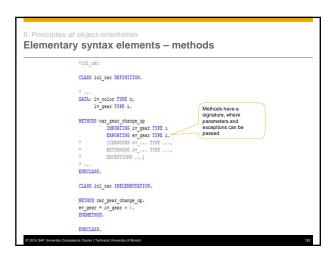


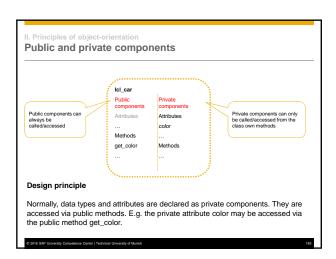


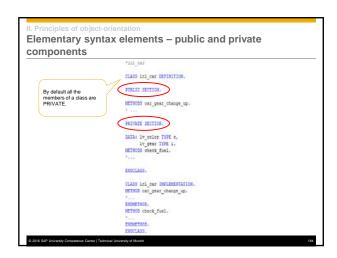


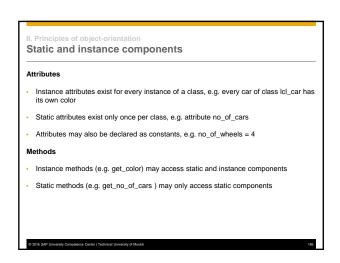


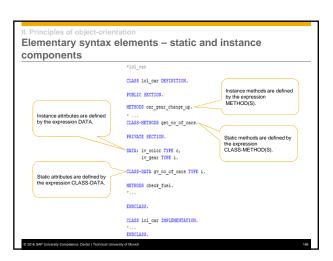




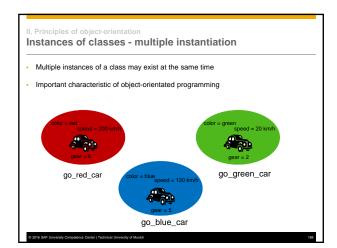


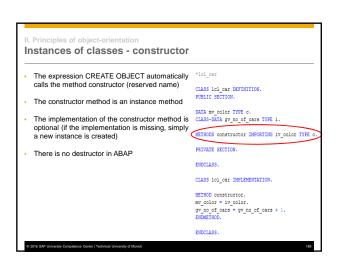


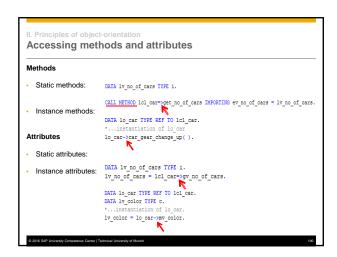


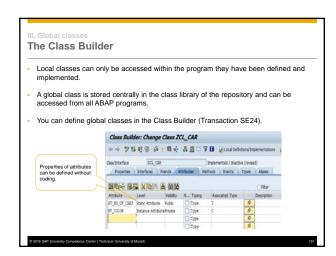


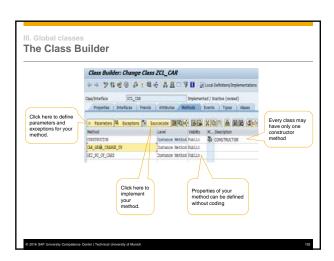
II. Principles of object-orientation Instances of classes One object of a class = one instance of a class The instantiation of an object is triggered by the expression CREATE OBJECT ... DATA go_red_car TYPE REF TO lcl_car. DATA go_blue_car TYPE REF TO lcl_car. CREATE OBJECT go_red_car EXPORTING ev_color = 'RED'. CREATE OBJECT go_blue_car EXPORTING ev_color = 'BLUE'.

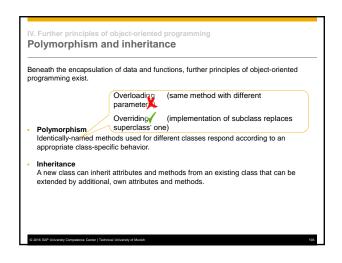


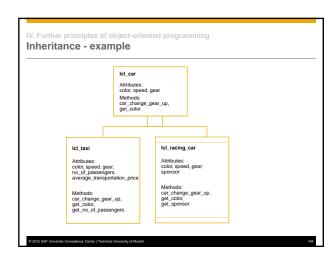


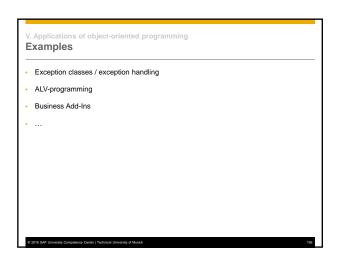


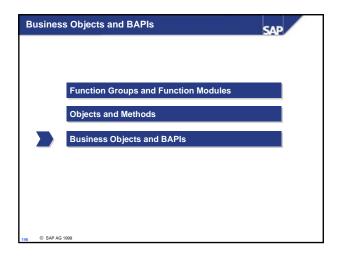


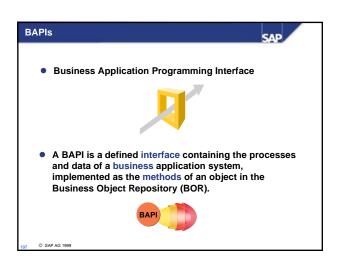


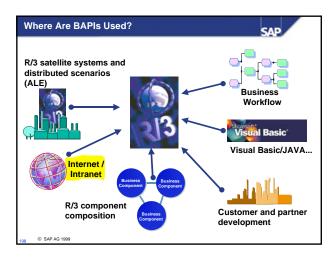




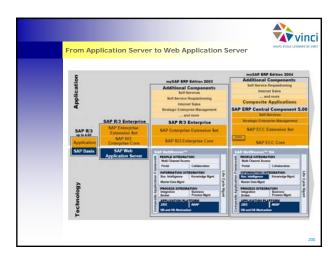


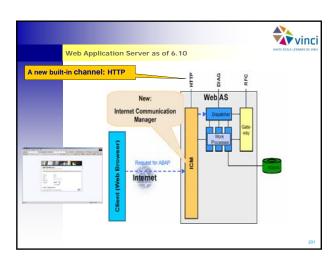


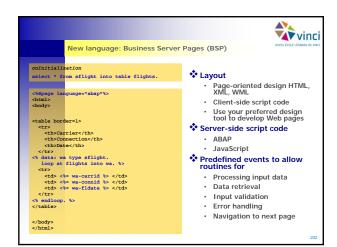


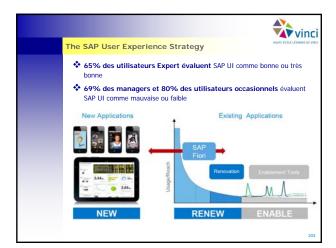




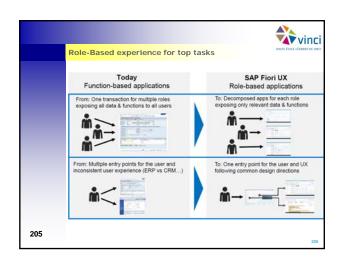




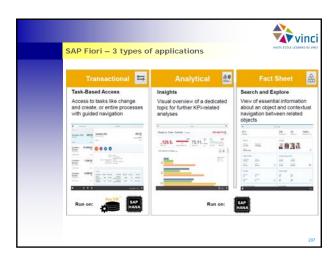


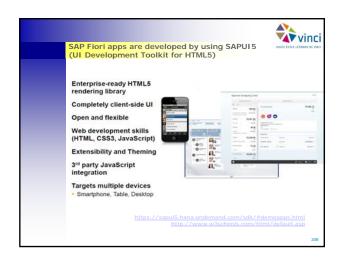


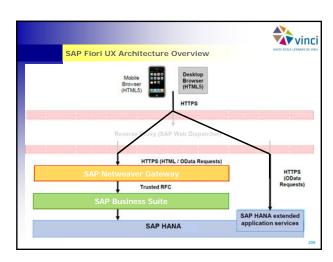


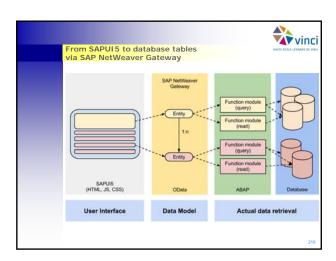


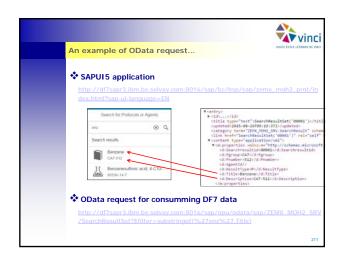


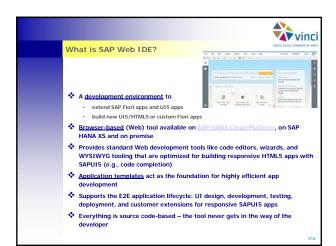


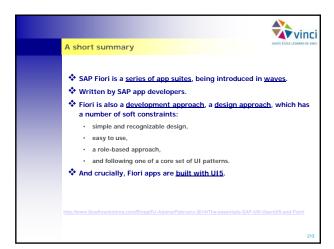


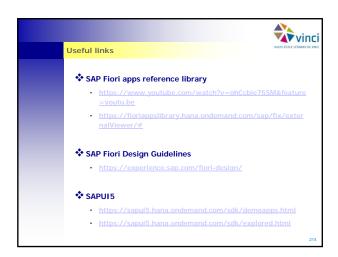


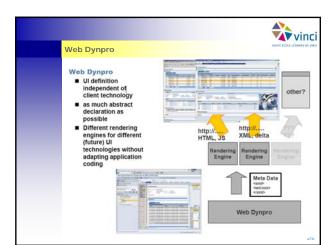


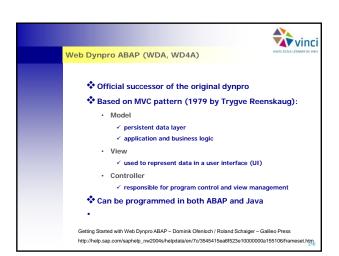


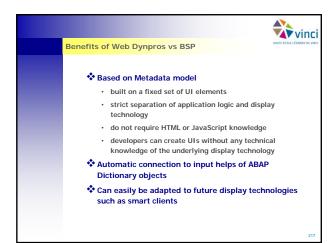












Disadvantages of Web Dynpros vs BSP

❖ Very rigid UI building

❖ No CSS features (but...)

❖ Design is limited to provided UI elements

❖ No Javascript enhancement possible (but...)