

Core Consultant Course (Siebel 8.0)

Volume II • Student Guide

D49247GC10

Edition 1.0

December 2007

D54096

ORACLE®

Author

Siebel Curriculum
Development team

**Technical Contributors
and Reviewers**

Siebel Review team

Publisher

Srividya Rameshkumar

Copyright © 2007, Oracle. All rights reserved.

Disclaimer

This document contains proprietary information and is protected by copyright and other intellectual property laws. You may copy and print this document solely for your own use in an Oracle training course. The document may not be modified or altered in any way. Except where your use constitutes "fair use" under copyright law, you may not use, share, download, upload, copy, print, display, perform, reproduce, publish, license, post, transmit, or distribute this document in whole or in part without the express authorization of Oracle.

The information contained in this document is subject to change without notice. If you find any problems in the document, please report them in writing to: Oracle University, 500 Oracle Parkway, Redwood Shores, California 94065 USA. This document is not warranted to be error-free.

Restricted Rights Notice

If this documentation is delivered to the United States Government or anyone using the documentation on behalf of the United States Government, the following notice is applicable:

U.S. GOVERNMENT RIGHTS

The U.S. Government's rights to use, modify, reproduce, release, perform, display, or disclose these training materials are restricted by the terms of the applicable Oracle license agreement and/or the applicable U.S. Government contract.

Trademark Notice

Oracle is a registered trademark of Oracle Corporation and/or its affiliates. Other names may be trademarks of their respective owners.

Contents

- Module i: Siebel 8.0 Essentials Training**
- Module 1: Introducing Siebel Applications**
- Module 2: Using the Siebel Web Client**
- Module 3: Working With Siebel Data**
- Module 4: Responsibilities and Views**
- Module 5: Users, Positions, and Organizations**
- Module 6: Controlling Access to Customer Data**
- Module 7: Catalogs and Master Data**
- Module 8: The Siebel Web Architecture**
- Module 9: Server Components and Parameters**
- Module 10: Server Management**
- Module 11: Siebel Client Types**
- Module 12: Securing Access to the Application**
- Module 13: Installing Siebel Applications**
- Module 14: Siebel Application Architecture**
- Module 15: Using Siebel Tools to Examine Object Definitions**
- Module 16: The Siebel Data Model**
- Module 17: Siebel Business Components**
- Module 18: Siebel Party Business Components**
- Module 19: Siebel Business Objects**
- Module 20: Configuration Strategy**
- Module 21: The Configuration Process**
- Module 22: Managing Object Definitions**
- Module 23: Editing and Compiling Object Definitions**
- Module 24: UI Layer Configuration: Web Templates**
- Module 25: UI Layer Configuration: Applets**
- Module 26: UI Layer Configuration: Applications, Screens, and Views**
- Module 27: UI Layer Configuration: Drilldowns**
- Module 28: Business Layer Configuration: Joins**
- Module 29: Business Layer Configuration: Existing Business Components and Fields**

- Module 30: Business Layer Configuration: New Business Components and Fields**
- Module 31: Business Layer Configuration: Picklists**
- Module 32: Configuring Multi-Value Groups**
- Module 33: Data Layer Configuration**
- Module 34: Siebel Business Services**
- Module 35: Building Siebel Workflow Processes**
- Module 36: Testing and Deploying Workflow Processes**
- Module 37: Executing Workflow Processes**
- Module 38: Using Workflow Policies**
- Module 39: Siebel Task UI**
- Module 40: Task UI: Creating a Task**
- Module 41: Transient Business Components and Branching**
- Module 42: Siebel Business Rules**
- Module 43: Creating Business Rules**
- Module 44: Introducing Siebel Assignment Manager**
- Module 45: Creating Assignment Rules**
- Module 46: Tailoring Assignment Manager Behavior**
- Module 47: Invoking Siebel Assignment Manager**
- Module 48: State Models**
- Module 49: Introducing Enterprise Integration Manager**
- Module 50: Creating Data Maps**
- Module 51: Running Enterprise Integration Manager**
- Module 52: Introducing Application Deployment Manager**
- Module 53: Deploying Application Customizations**

ORACLE®

Siebel 8.0 Essentials

Module 17: Siebel Business Components

17

Copyright © 2007, Oracle. All rights reserved.

Oracle Internal & Oracle Academy Use Only





Module Objectives

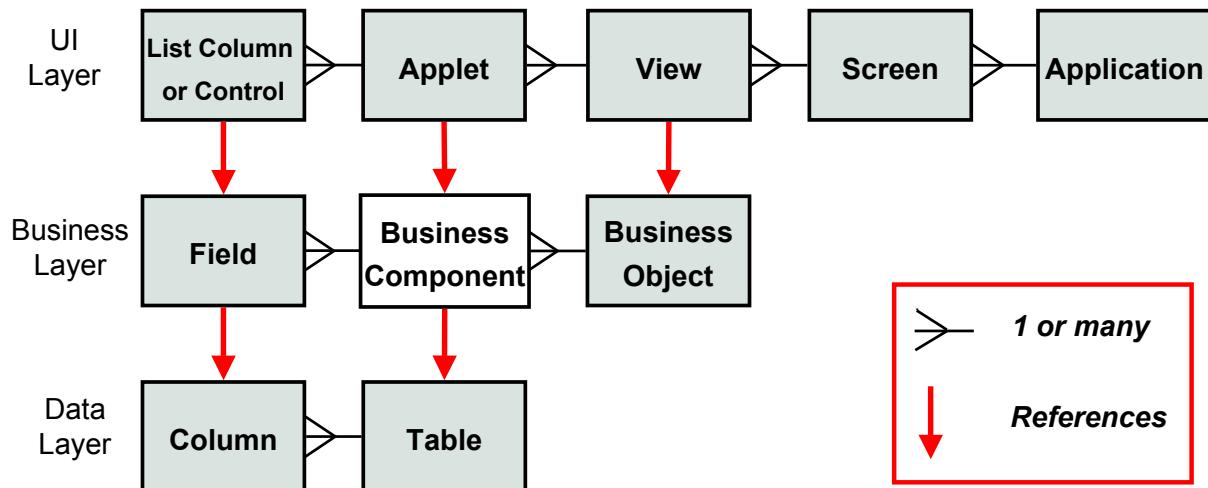
- After completing this module you should be able to:
 - ▶ Define a business component
 - ▶ Describe how business component fields at the business object layer are mapped to columns at the data layer
 - ▶ Describe how base and joined tables are used as a part of this mapping
- Why you need to know:
 - ▶ The business component is a fundamental object in the Siebel Application Architecture

ORACLE®

Application Architecture: Business Components

- Provide a way to group data according to business logic
- Are referenced by applets
- Specify access to tables (read, write, and update)
- Are used by business objects to provide data to views

Siebel Application Architecture



Copyright © 2007, Oracle. All rights reserved.

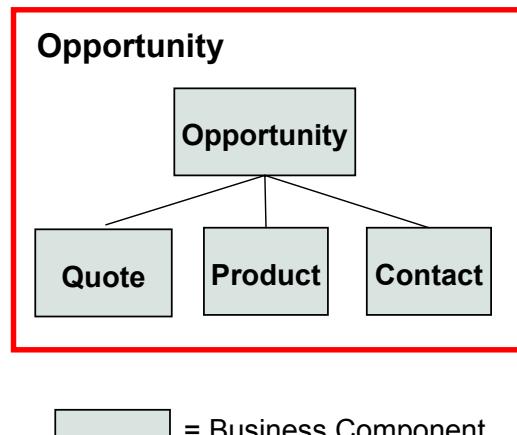
3 of 18





Business Component

- A business component (BC) is a person, place, thing, or event about which data must be stored
 - ▶ Represents a fundamental business *entity*
- Provides the foundation for controlling how data is selected, inserted, and updated in underlying tables



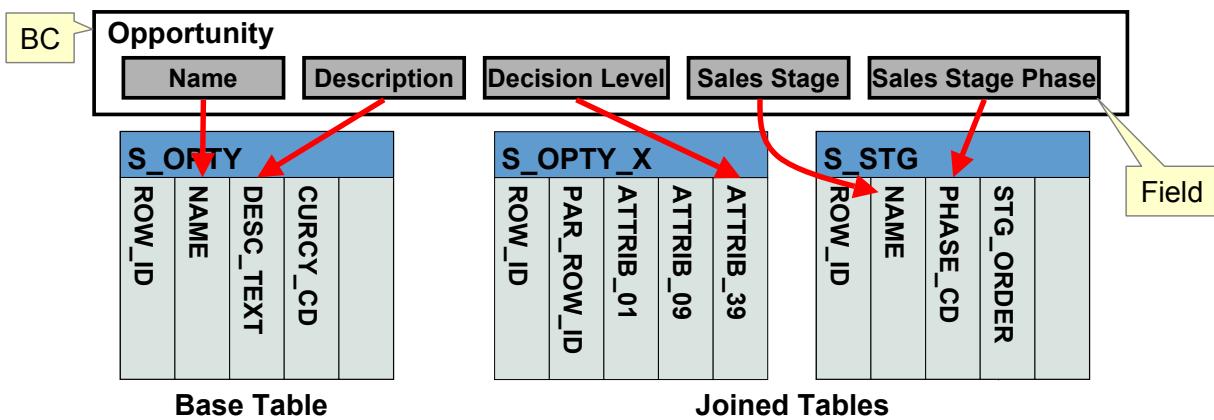
Copyright © 2007, Oracle. All rights reserved.

4 of 18

ORACLE®

Business Component Continued

- Arranges data from one or more tables into a logical grouping
- Consists primarily of fields and joins
 - ▶ Each field references a single column in a table
- Is not a table
 - ▶ Does not store data
 - ▶ Stores metadata: data about data



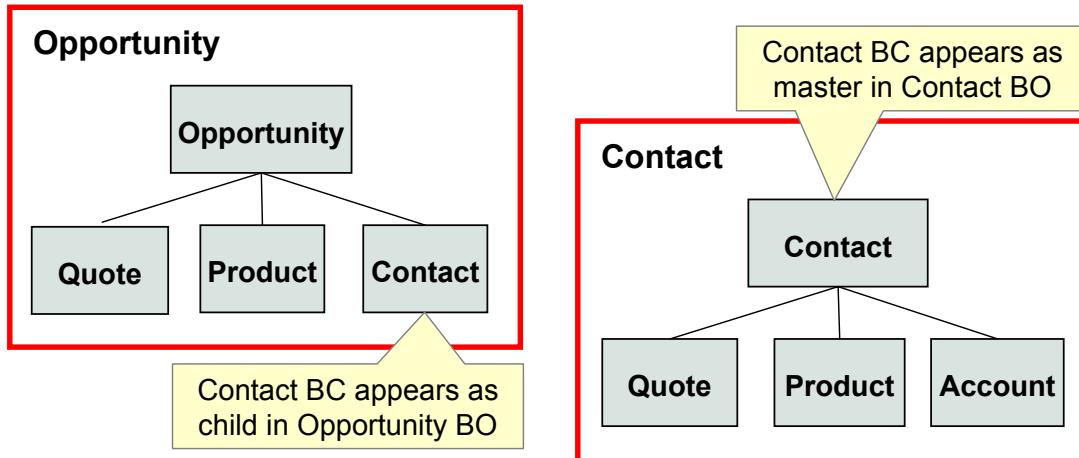
Copyright © 2007, Oracle. All rights reserved.

5 of 18



Business Component Reuse

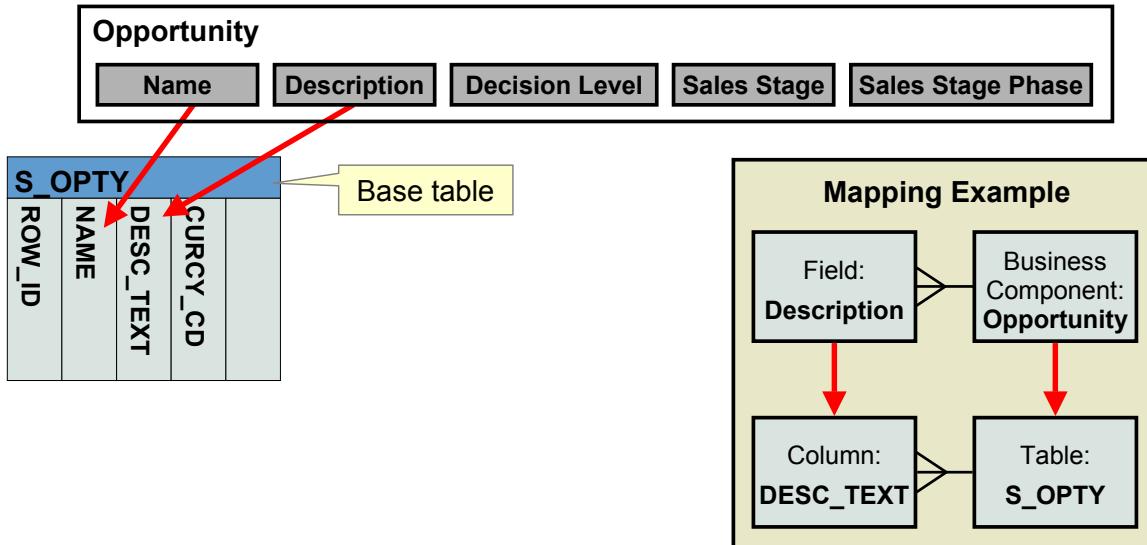
- A business component can be:
 - ▶ Defined once in terms of a logical collection of columns from one or more tables
 - ▶ Then used in many different business object contexts



ORACLE®

Base Table

- Contains the main columns for the business component
- Every business component references only one base table
- Many BC fields are mapped to base table columns
- Fields referencing base tables are editable in the UI



Copyright © 2007, Oracle. All rights reserved.

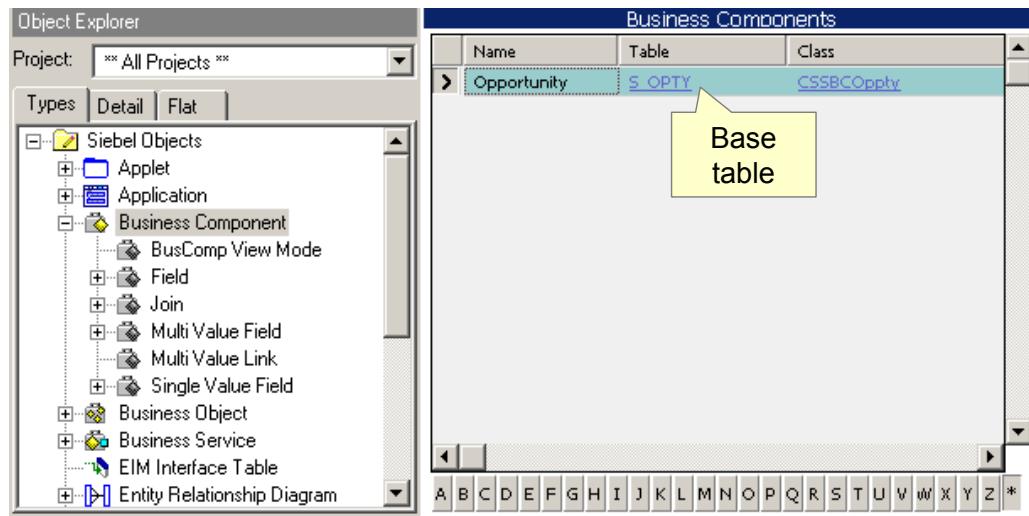
7 of 18



ORACLE®

Business Component Table Property

- Each business component contains a table property that specifies the base table



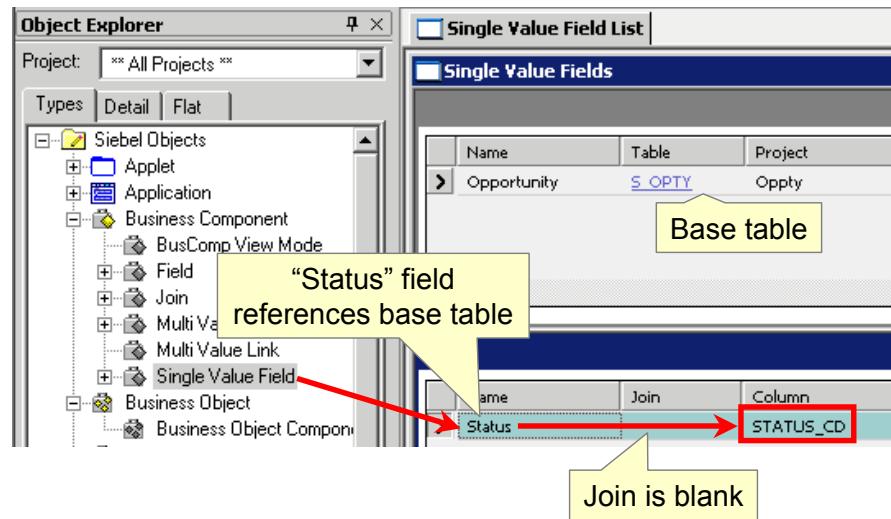
Copyright © 2007, Oracle. All rights reserved.

8 of 18

ORACLE®

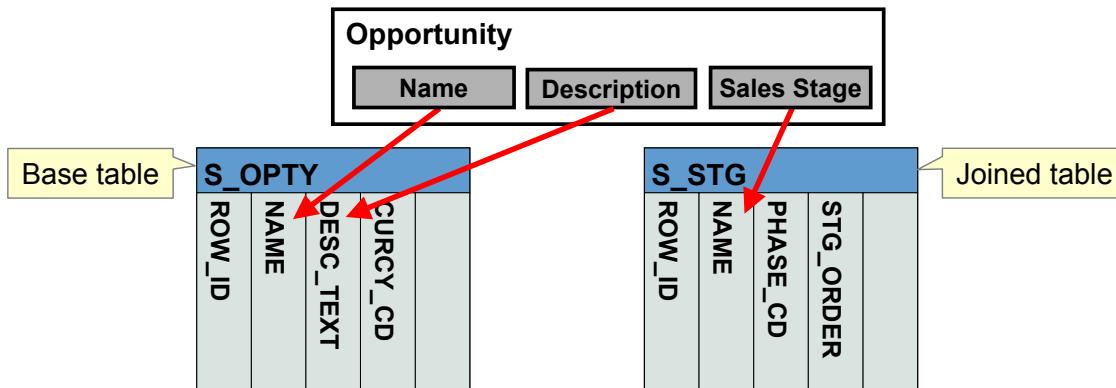
Fields That Map to a Base Table

- Column property maps to a column in the base table
 - ▶ Join property is blank



Explicit Join

- Is a join that brings in data from tables other than the base table to meet the business component's data display requirements
 - ▶ Most fields referencing explicitly joined tables are read-only in the applet
- Includes a join definition and a join specification
- BC field references the join definition



Copyright © 2007, Oracle. All rights reserved.

10 of 18

ORACLE®

Fields That Map to a Joined Table

- Join property specifies the join object definition being referenced
- Column property maps to a column in the joined table

The screenshot shows the Siebel Object Explorer and the Single Value Field List interface.

Object Explorer: Shows the project "All Projects" and the types of objects available, including Siebel Objects, Applet, Application, Business Component, Business Object, Business Service, and EIM Interface Table. A "Single Value Field" node is selected.

Single Value Field List: Shows the "Single Value Fields" for the "Opportunity" object, which is mapped to the "S_OPTY" table. The "Sales Stage" field is highlighted.

Annotations:

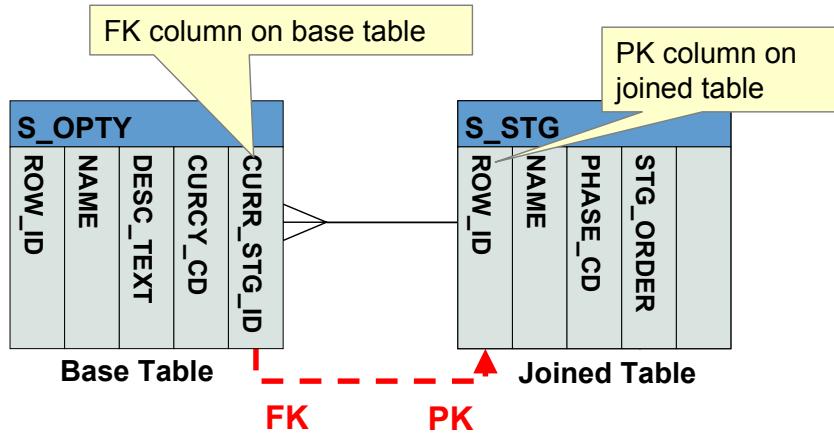
- A yellow callout points to the "Single Value Field" node in the Object Explorer with the text: "Sales Stage field maps to Sales Stage join".
- A yellow callout points to the "Sales Stage" field in the Single Value Fields list with the text: "Name of join that maps to S_STG table".
- A yellow callout points to the "NAME" column in the Single Value Fields list with the text: "The S_STG column, "NAME," contains sales stage data".
- A red arrow points from the "Sales Stage" field in the Object Explorer to the "Sales Stage" field in the Single Value Fields list.

Copyright © 2007, Oracle. All rights reserved.

11 of 18

Joining Data from Joined Tables

- Returns only one row from the joined table
- There is a 1:1 or 1:M relationship from the joined table to the BC
- Relationship is established using a FK column on the base table to join to the PK column on the joined table



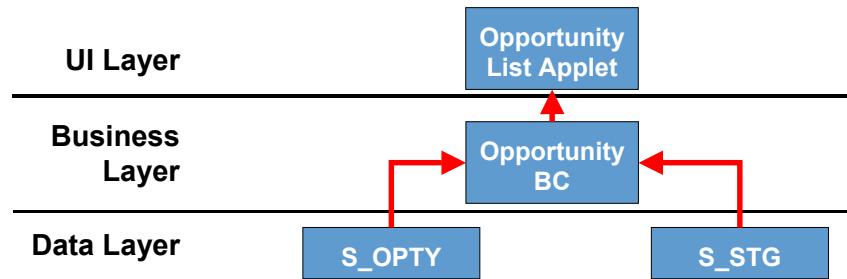


Displaying Fields from Joined Tables

- The UI layer specifies how information joined at the business layer will be displayed

Opportunities Home Opportunities List Charts Opportunity Explorer Mana		
Opportunity Name	Account	Sales Stage
200 PCS CS Laptop units	AT&T	J1 - Prospecting
200 PCS Puma Laptop EB units	Imperial Tobacco	J2 - Qualification
200 PCS Puma Laptop EB units	Broadband e2e	

From Base Table From Joined Table



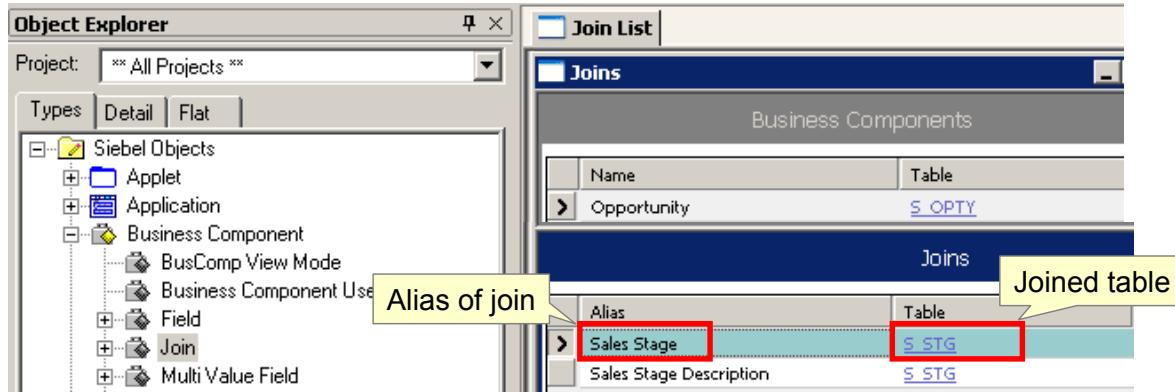
Copyright © 2007, Oracle. All rights reserved.

13 of 18

ORACLE®

Join Definition

- Specifies the joined table from which to retrieve data
 - ▶ Is a child object of the business component
- Defaults the alias property to the name of the joined table
 - ▶ Modify alias when there is more than one join to same table



ORACLE®

Join Specification

- Specifies how to retrieve the related row from the joined table
 - ▶ Based on the foreign and primary keys used to relate the base and joined tables

The diagram illustrates the relationship between the Opportunity and Sales Stage tables. The Opportunity table has columns: Name, Description, and Sales Stage Id. The Sales Stage table has columns: ROW_ID, NAME, DESC_TEXT, CURCY_CD, and Curr_Stage_Id. A dashed line with an arrow labeled 'FK' connects Curr_Stage_Id to ROW_ID, indicating a Foreign Key relationship. A dashed line with an arrow labeled 'PK' connects ROW_ID to ROW_ID, indicating a Primary Key relationship. Red arrows point from the Sales Stage Id field in the Opportunity table to the Curr_Stage_Id column in the Sales Stage table, and from the Sales Stage table's ROW_ID column to the Sales Stage Id field in the Opportunity table. The Oracle Object Explorer interface shows the Join Specifications List and Join Specifications details, where Sales Stage is selected as the alias, and Sales Stage Id is mapped to S_STG. A callout box labeled 'Foreign Key field' points to Sales Stage Id, and another labeled 'Primary key column' points to ROW_ID.

Object Explorer

Project: ** All Projects **

Types Detail Flat

- Siebel Objects
 - Applet
 - Application
 - Business Component
 - BusComp View Mode
 - Business Component User
 - Field
 - Join
 - Join Constraint
 - Join Specification
 - Multi Value Field
 - Multi Value Link
 - Multi Value User Field

Join Specification List

Join Specifications

Alias	Table	Outer Join Flag
S_STG	Sales Stage	✓

Join Specifications

Name	Source Field	Destination Column
Sales Stage Id	Sales Stage Id	ROW_ID

Opportunity

Name	Description	Sales Stage Id
------	-------------	----------------

S_OPTY

ROW_ID	NAME	DESC_TEXT	CURCY_CD	Curr_Stage_Id
--------	------	-----------	----------	---------------

S_STG

ROW_ID	NAME	PHASE_CD	STG_ORDER
--------	------	----------	-----------

Copyright © 2007, Oracle. All rights reserved.

15 of 18



Implicit Joins

- Base tables are automatically joined to their extension tables
 - ▶ Known as implicit joins
 - ▶ Make extension table rows available to the BC
 - ▶ An explicit join to describe the relationship is not needed
- Come pre-configured as part of the Siebel object architecture
 - ▶ Name of an implicit join is always the name of the _X table
- Do not appear as Join object definitions
- Do appear in the picklist for the Join property in an SVF

The screenshot shows three windows illustrating Siebel's join mechanism:

- Join Definition Window:** A table titled "Joins" with columns "Alias" and "Table". It lists several joins, including "S_LST_OF_VAL" to "S_LST_OF_VAL", "Parent Opportunity" to "S_OPTY", "S_ORG_EXT" to "S_ORG_EXT", "S_ORG_EXT_T" to "S_ORG_EXT_T", "S_POSTN" to "S_POSTN", "Sales Method" to "S_SALES_METHOD", and "S_OPTY_X" to "S_OPTY_X". A yellow callout box points to the last entry: "S_OPTY_X does not appear here".
- Single Value Fields Window:** A table titled "Single Value Fields" with columns "Name", "Join", and "Column". It shows a single row: "An Extension Field" under "Join".
- Join Picklist Window:** A window titled "Join" with a search bar "Find Name" and a dropdown "starting with". It lists several joins: "S_OPTY_T", "S_OPTY_X" (which is highlighted in blue), "S_ORG_EXT", and "S_ORG_EXT_T". A yellow callout box points to "S_OPTY_X" with the text "S_OPTY_X does appear here".

Copyright © 2007, Oracle. All rights reserved.

16 of 18

ORACLE®

Fields That Map to an Extension Table

- Join property specifies the extension table being referenced
- Column property references a column in the extension table
- Field is editable in the UI

The S_OPTY_X column, "ATTRIB_44," contains budget amount data

The Budget Amt field references the S_OPTY_X table

Name	Join	Column
Budget Amt	S_OPTY_X	ATTRIB_44
Budget Amt Score		
Budget Available		BDGT_AVAILABLE_DT
Budgeted	S_OPTY_X	ATTRIB_09

Copyright © 2007, Oracle. All rights reserved.

17 of 18





Module Highlights

- A BC is a person, place, thing, or event about which data is stored
- A BC can be defined once then re-used by different BOs
- Each BC references a single base table
- A BC can include data from joined tables
- A join definition specifies joined table
- A join specification specifies how to access joined table via PK/FK
- An extension table extends data in the BC
- Fields on extension tables use implicit joins

Copyright © 2007, Oracle. All rights reserved.

18 of 18

ORACLE®

Siebel 8.0 Essentials

Module 18: Siebel Party Business Components

18

Copyright © 2007, Oracle. All rights reserved.

18

Oracle Internal & Oracle Academy Use Only



Module Objectives

- After completing this module you should be able to:
 - ▶ Define a party business component
 - ▶ Describe the role of S_PARTY and its extension tables in storing party business component data
 - ▶ Describe how data is stored differently for non-party business components and party business components
 - ▶ Describe how implicit and explicit joins are used with party business components
- Why you need to know:
 - ▶ Party business components are a fundamental element of the application architecture



Business Challenge: Modeling Party Data

- Party data is the ubiquitous information found in most RDBMS
 - ▶ Some prominent *party types* include Contacts, Employees, Positions, Accounts, User Lists, Organizations, and Access Groups
- *Party types* are often related to each other
 - ▶ Employees are related to positions
 - ▶ Positions are related to accounts
 - ▶ Access groups are related to organizations
 - ▶ and so forth
- Relationships may be dynamic or ad hoc
 - ▶ A contractor becomes an intern, then an employee
 - ▶ An access group is required for participants of the January 2007 Big Release Roll-Out Event

Copyright © 2007, Oracle. All rights reserved.

3 of 26

Reference

- Siebel Security Guide: Configuring Access Control
- Configuring Siebel Business Applications: Configuring Tables and Columns: About the S_PARTY Table

18

ORACLE®

Business Challenge: Modeling Party Data Continued

- Representing these relationships could result in data models that do not have optimal normalization
 - ▶ Multiple records may be created for each relationship
 - A separate record is created for a contractor, another for an intern, and a third for an employee
 - ▶ A special entity may have to be created for each new ad hoc relationship

Contractor Table

1. An individual is a contractor . . .

First Name	Last Name	Phone
Navdeep	Singh	321-654-0987

Intern Table

2 . . . becomes an intern . . .

First Name	Last Name	Phone
Navdeep	Singh	321-654-0987

Same data, three different places

Employee Table

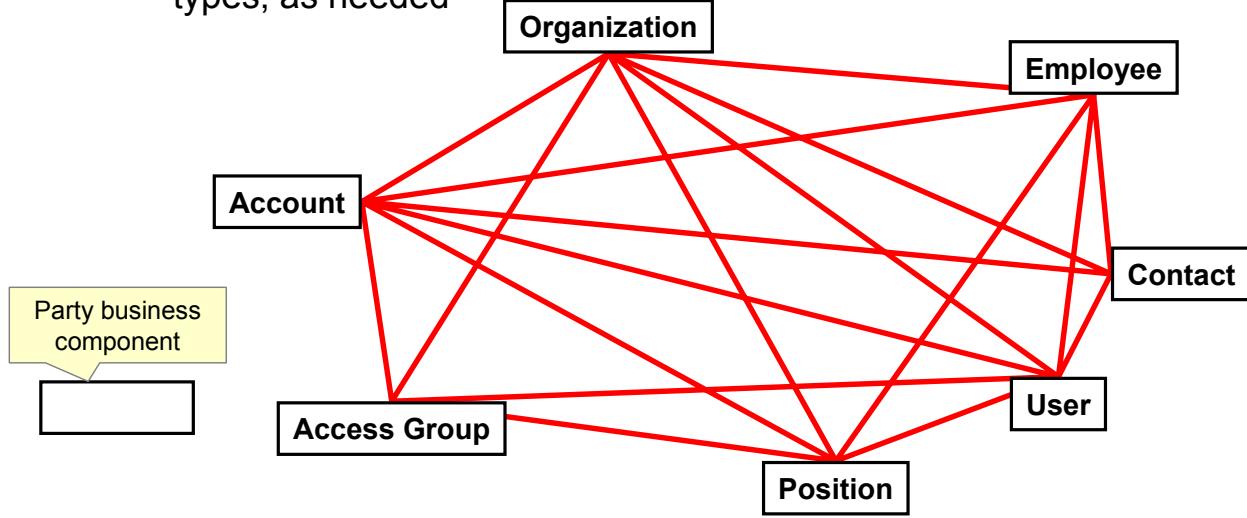
3 . . . then an employee.

First Name	Last Name	Phone
Navdeep	Singh	321-654-0987

ORACLE®

Business Solution: Party Business Component

- Provides a way to create a network of relationships between party types to reflect changes and complexities in the business environment
- Infuses flexibility into the data model
 - ▶ Can establish and change relationships between various party types, as needed



Copyright © 2007, Oracle. All rights reserved.

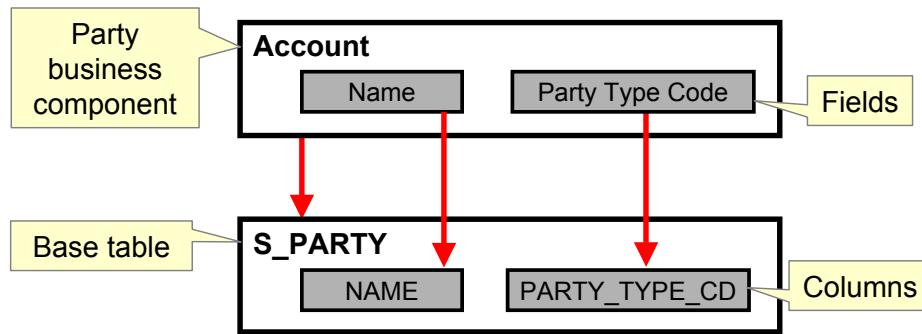
5 of 26

18

ORACLE®

Party Business Components

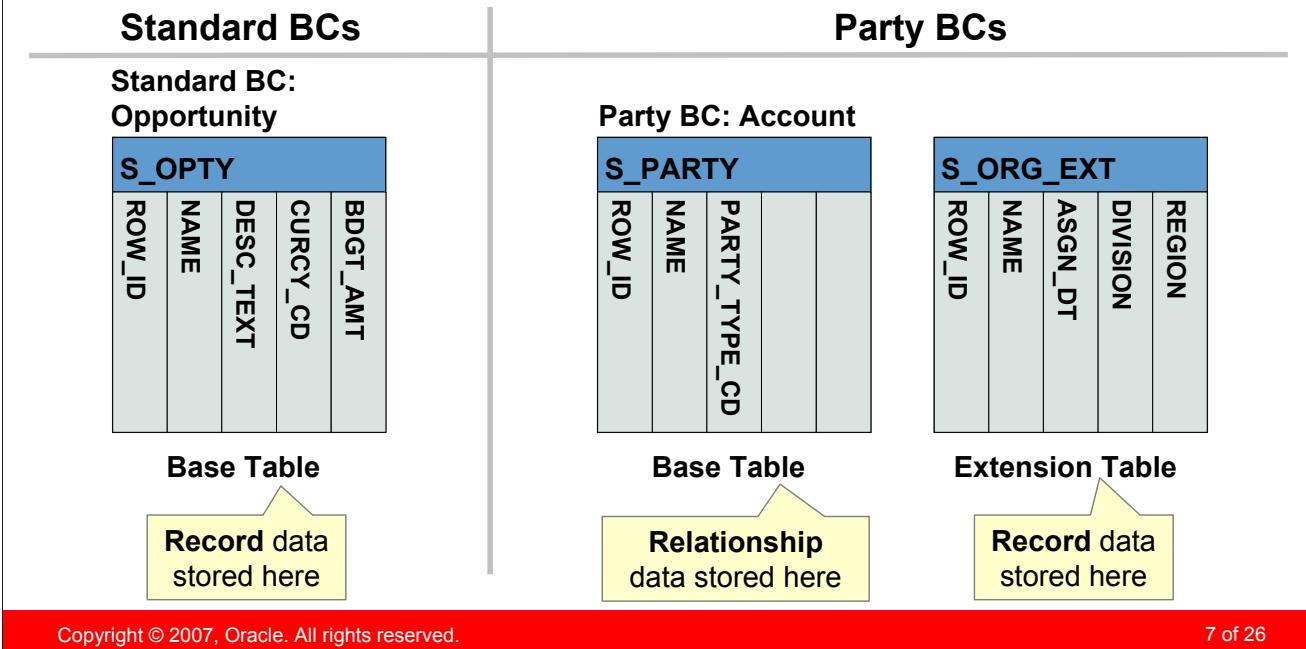
- Are similar to standard business components; they:
 - ▶ Group data according to business logic
 - ▶ Are referenced by applets
 - ▶ Specify access to tables
 - ▶ Have fields that map to columns



ORACLE®

Differences Between Party and Non-Party BCs

- In standard BCs, most data is stored in the base table
- In party BCs, data is stored in extension tables
- S_PARTY acts solely as a linking mechanism between types



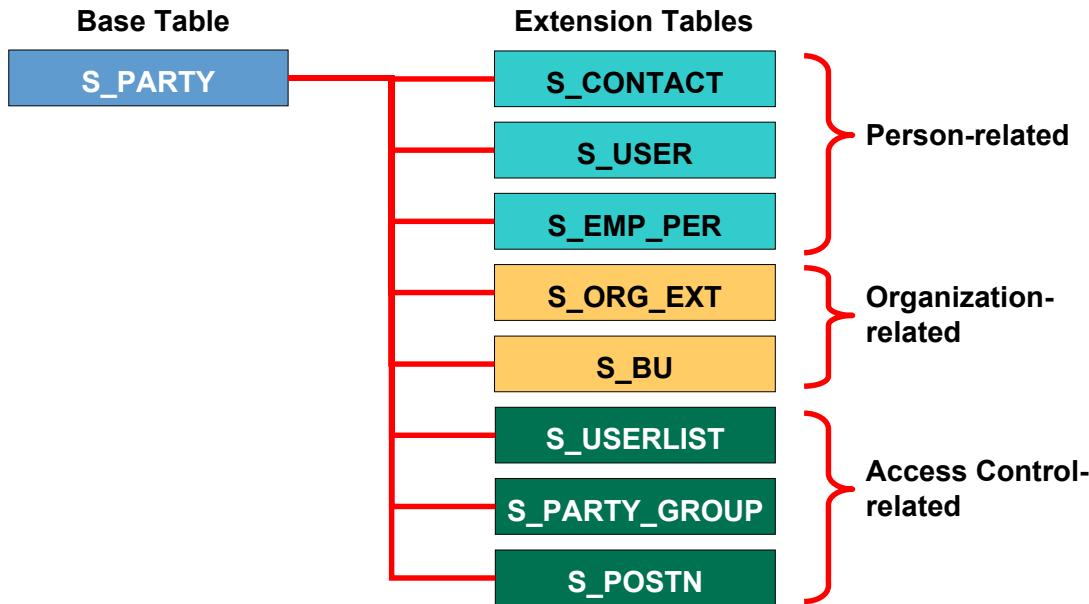
Copyright © 2007, Oracle. All rights reserved.

7 of 26

18

S_PARTY and Its Extension Tables

- Eight prominent S_PARTY extension tables store the data



Copyright © 2007, Oracle. All rights reserved.

8 of 26

Other S_PARTY Extension Tables

Tables such as **S_CONTACT_X** (which is the 1:1 extension table for the **S_CONTACT** table) are formally an **S_PARTY** extension table.

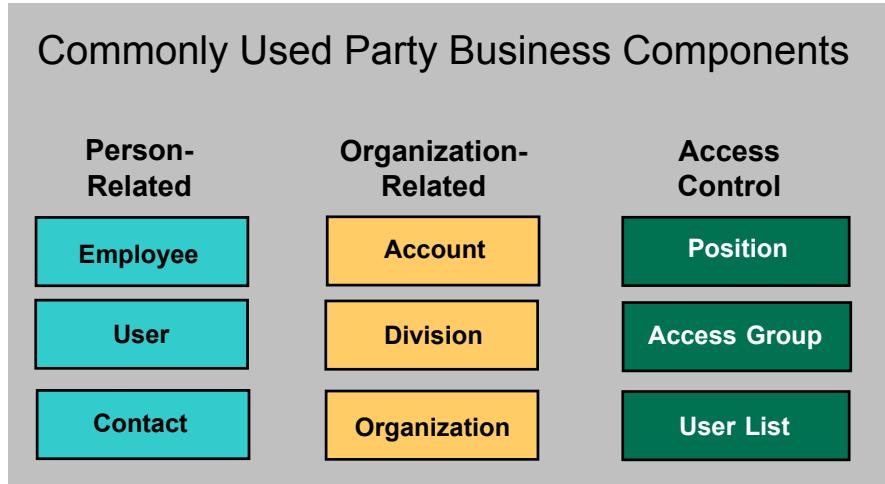
To list the complete set of **S_PARTY** extension tables, execute a query in Siebel Tools to retrieve all tables with the Base Table property = **S_PARTY**.

0/3

ORACLE®

Commonly Used Party Business Components

- Represent a variety of entities that can be arranged into groups related to persons, organizations, or access control



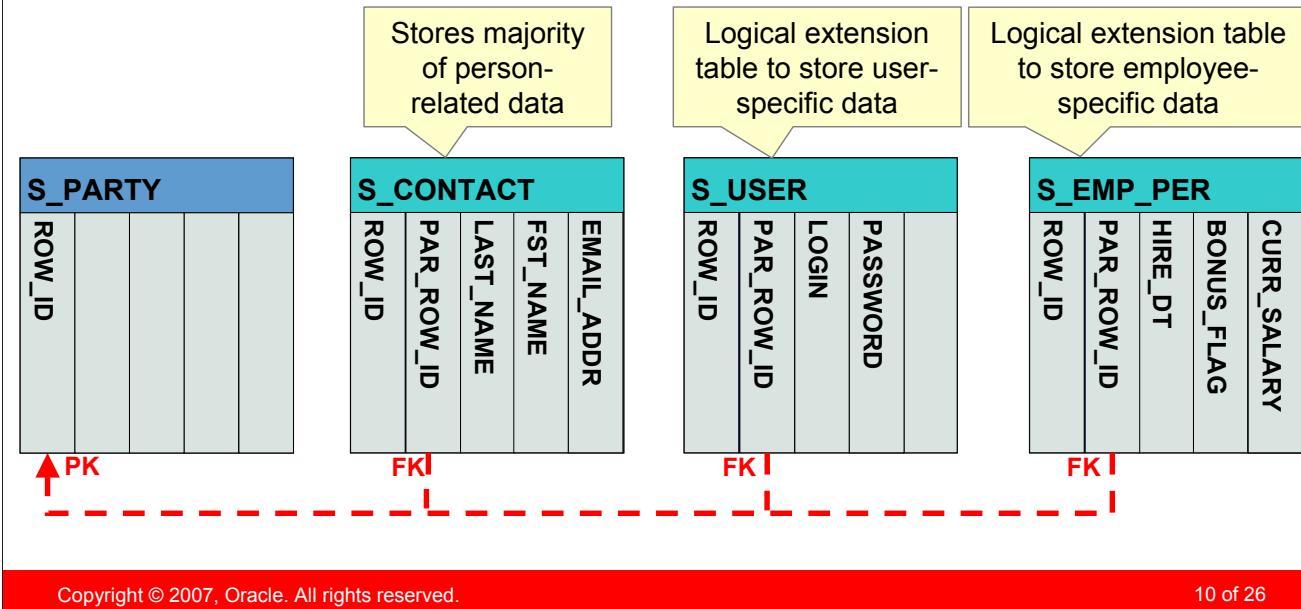
Copyright © 2007, Oracle. All rights reserved.

9 of 26

18

Person-Related Party Business Components

- Store their main data in S_CONTACT
- May store additional data in S_USER and S_EMP_PER
 - ▶ Serve as logical extension tables

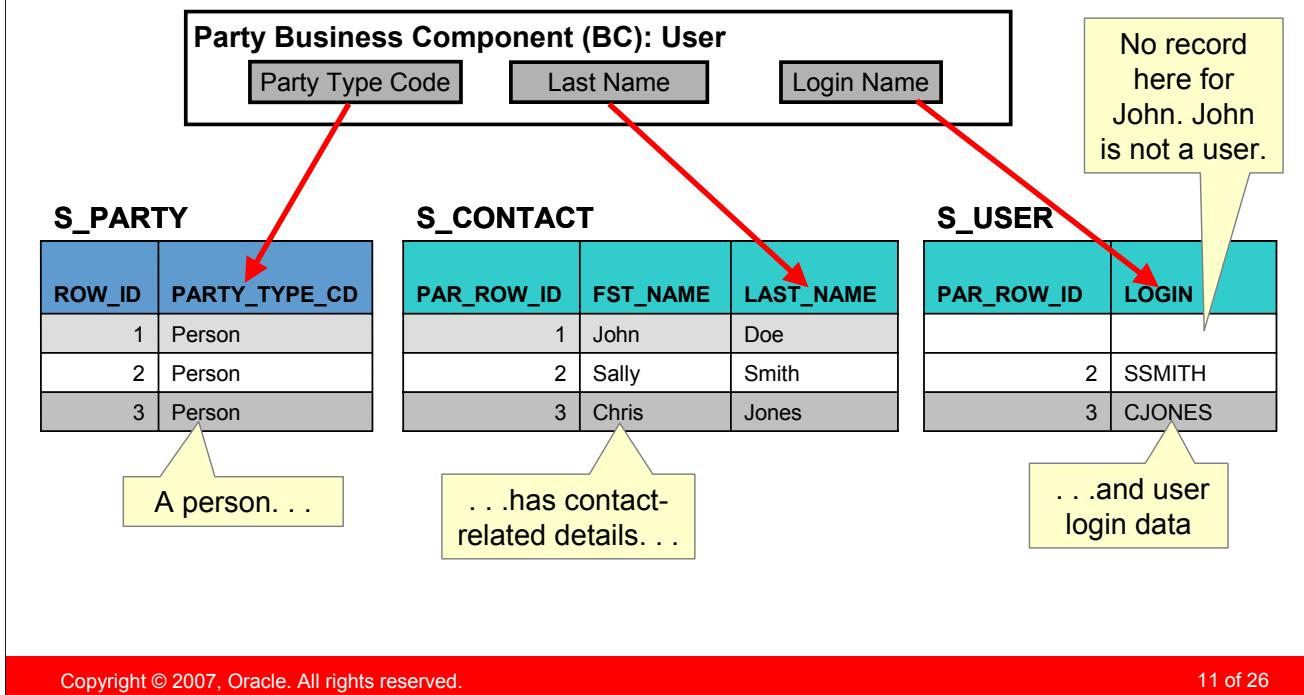


Copyright © 2007, Oracle. All rights reserved.

10 of 26

Person-Related Party Business Components Continued

- Primarily reference person-related S_PARTY extension tables

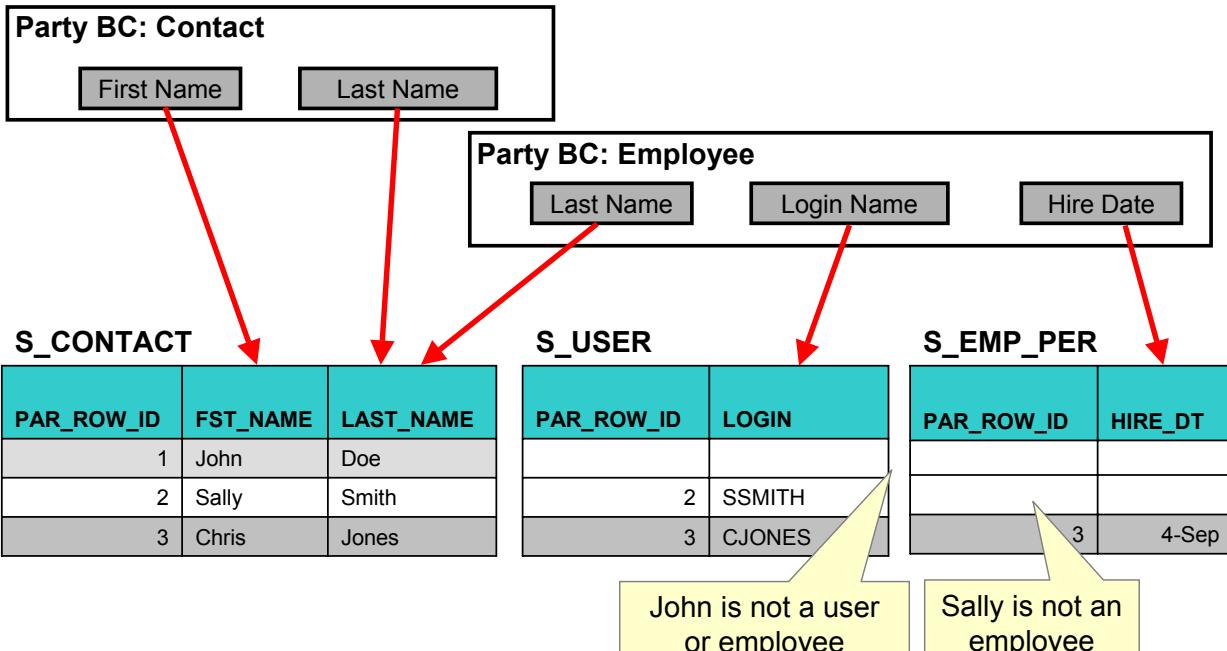


Copyright © 2007, Oracle. All rights reserved.

11 of 26

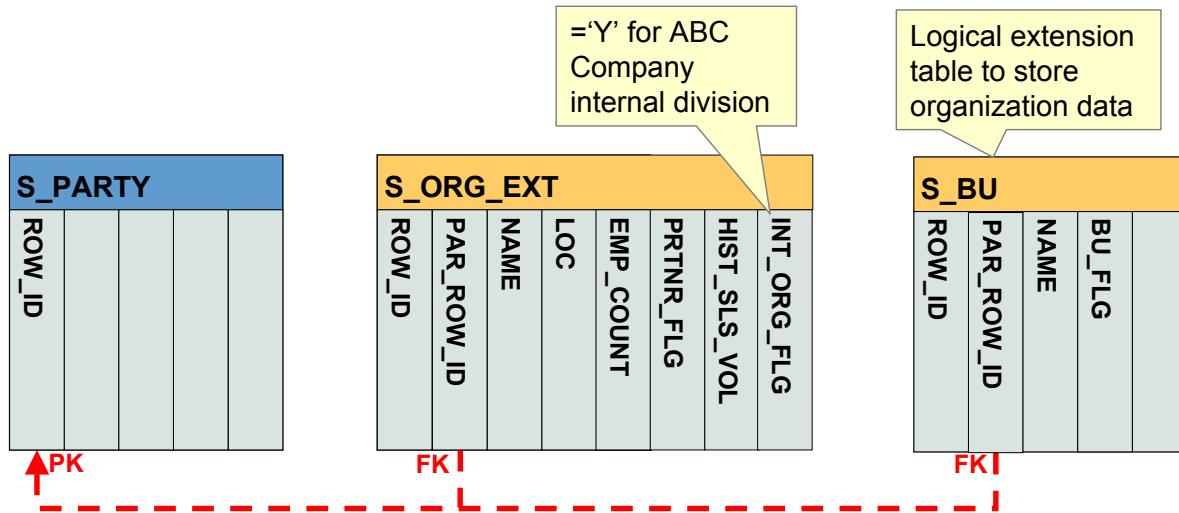
Person-Related Party Business Components Continued

- A number of person-related business components use these tables



Organization-Related Party Business Components

- Store their main data in S_ORG_EXT
- May store additional data in S_BU
- May include account, division, organization, or household data

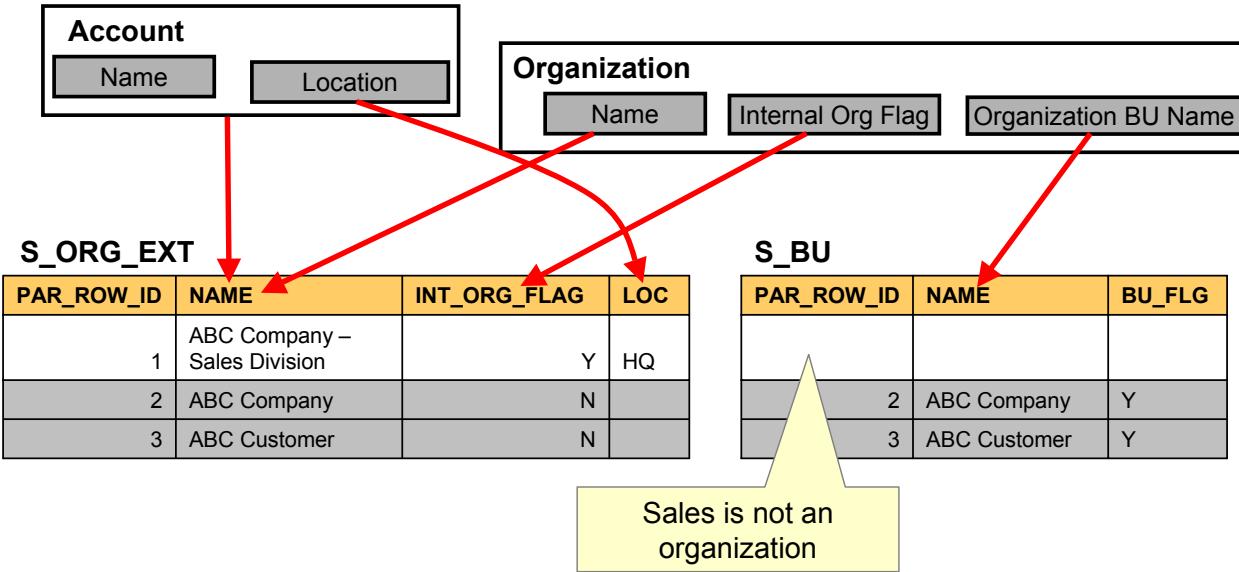


Copyright © 2007, Oracle. All rights reserved.

13 of 26

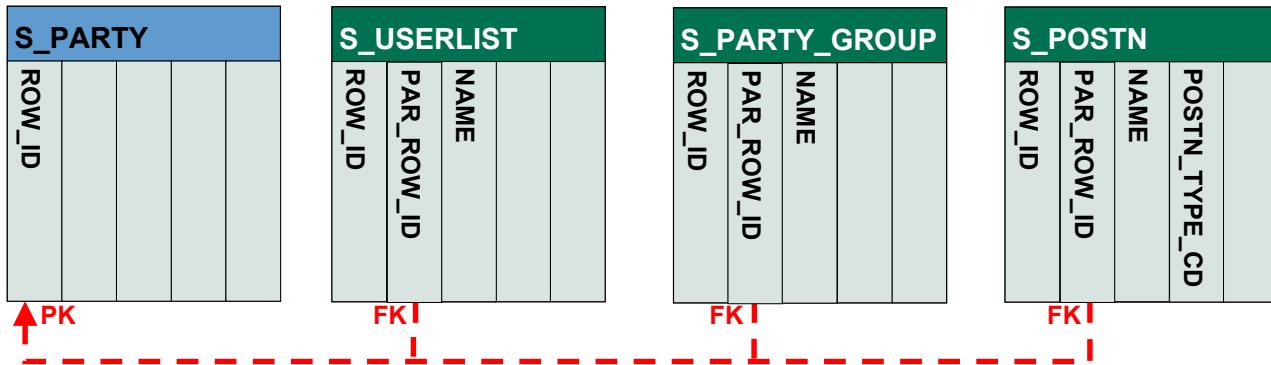
Organization-Related Party Business Components Continued

- Multiple organization-related business components use these tables



Groupings for Access Control

- Represent groupings of party instances
 - ▶ User List
 - ▶ Access Group
 - ▶ Position



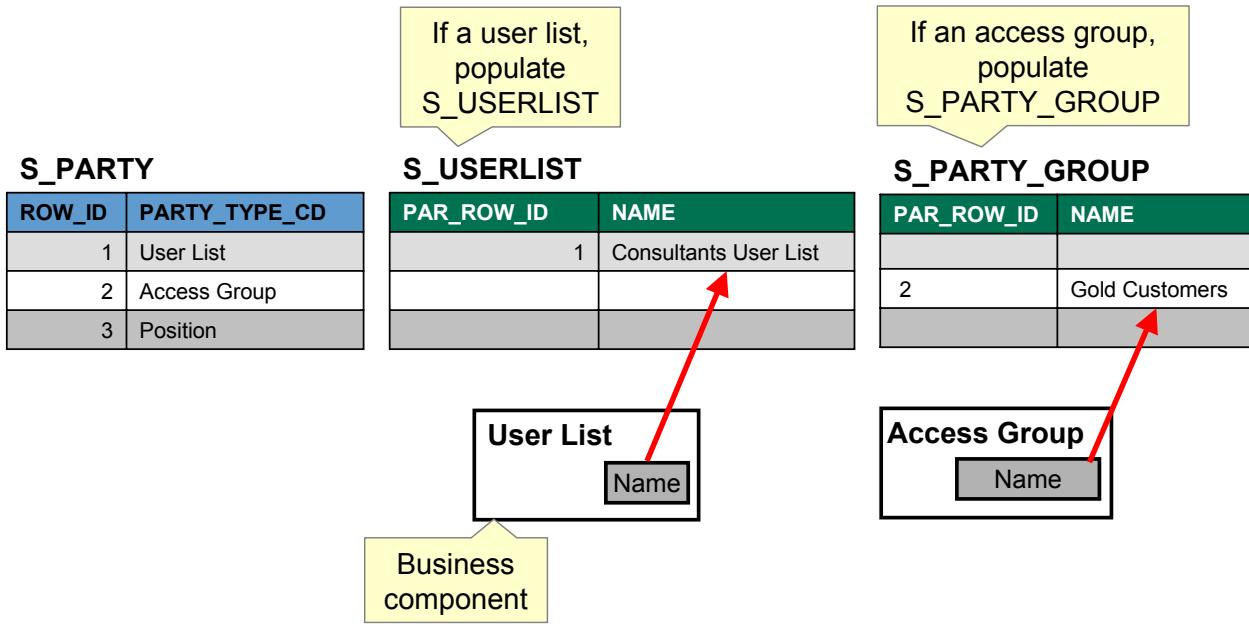
Copyright © 2007, Oracle. All rights reserved.

15 of 26

18

Groupings for Access Control continued

- Access Group, User List, and Position are party business components



ORACLE®

Example, Relating Party Data

- A user list can be associated with persons via the S_PARTY intersection table S_PARTY_PER
- There are relationships in S_PARTY
 - ▶ They are represented in S_PARTY_PER
 - ▶ ROW_ID in S_PARTY is used to relate party types

S_PARTY_PER

PARTY_ID	PERSON_ID
003	001
003	002

S_PARTY

ROW_ID	PARTY_TYPE_CD	NAME
001	Person	Smith, Mary
002	Person	Smith, John
003	User List	ABC User List

Copyright © 2007, Oracle. All rights reserved.

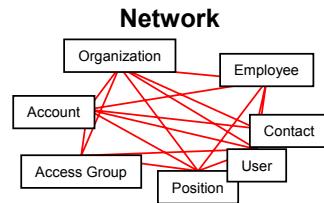
17 of 26

18

ORACLE®

Example, Relating Party Data Continued

- An access group can then be networked with those users, or other user lists, or most any other S_PARTY type
 - ▶ Person, User List, Organization and Account party types can be related to an Access List party type

**S_PARTY_PER**

PARTY_ID	PERSON_ID
003	001
003	002
005	003
005	004

S_PARTY

ROW_ID	PARTY_TYPE_CD	NAME
001	Person	Smith, John
002	Person	Smith, Mary
003	User List	ABC User List
004	Organization	ABC Org
005	Access List	ABC Access Group

Copyright © 2007, Oracle. All rights reserved.

18 of 26



Party Implicit Joins

- Used to populate the prominent S_PARTY extension tables
- Created automatically for these extension tables
- Are similar to implicit joins for standard business components
 - ▶ Do not appear in join object definitions
 - ▶ ROW_ID in base table is always the PK
 - ▶ PAR_ROW_ID in extension is always the FK

S_PARTY	
ROW_ID	PARTY_TYPE_CD
1	Person
2	Person
3	Person

S_CONTACT		
PAR_ROW_ID	FST_NAME	LAST_NAME
1	John	Doe
2	Sally	Smith
3	Chris	Jones

S_USER	
PAR_ROW_ID	LOGIN
2	SSMITH
3	CJONES



Copyright © 2007, Oracle. All rights reserved.

19 of 26

ORACLE®

Explicit Join: Non-Party Business Component

- Data in a party table can be joined into a non-party business component
- Example: Bringing account data into the Opportunity business component for display in an Opportunity applet

The screenshot shows the Siebel Opportunities List screen. At the top, there's a navigation bar with links like Home, Accounts, Contacts, Opportunities, Quotes, Sales Orders, and Service. Below that is a sub-navigation bar with Opportunities List, Charts, Opportunity Explorer, Manager's Explorer, and Opportunities Administration.

The main area is titled "My Opportunities" and contains a table with columns: New, Opportunity Name, Account, and Revenue. The "Opportunity Name" column and the "Account" column are highlighted with red boxes. Red arrows point from these boxes down to a callout box labeled "Non-Party BC: Opportunity" which contains four buttons: Name, Description, Account, and Account Location.

New	Opportunity Name	Account	Revenue
*	Fast Ethernet NIC PCI 10/100 - 2500 units	Marriott International HQ	\$687,500.00
*	Digi Phones for mobile field engineers	Economy Printing & Copying	\$500,000.00
*	Pentium Server for new web sites	RS Semiconductors	\$250,000.00
*	Monitor - 17" CRT - flat tube - 1000 units	Marriott International HQ	\$200,000.00
*	New Digi Phones for Field	Apex Graphics Inc	\$200,000.00
*	Client desktops for new video tracking system	Video On Demand, Inc	\$150,000.00
	2111x PCS Chev Desktop ES 1-22ONP	Imperial Tobacco	\$0.00
	262x PCS Chev Desktop ES 1-22ONP	Citicorp Capital Asia Ltd	\$0.00

Non-Party BC: Opportunity

Name
Description
Account
Account Location

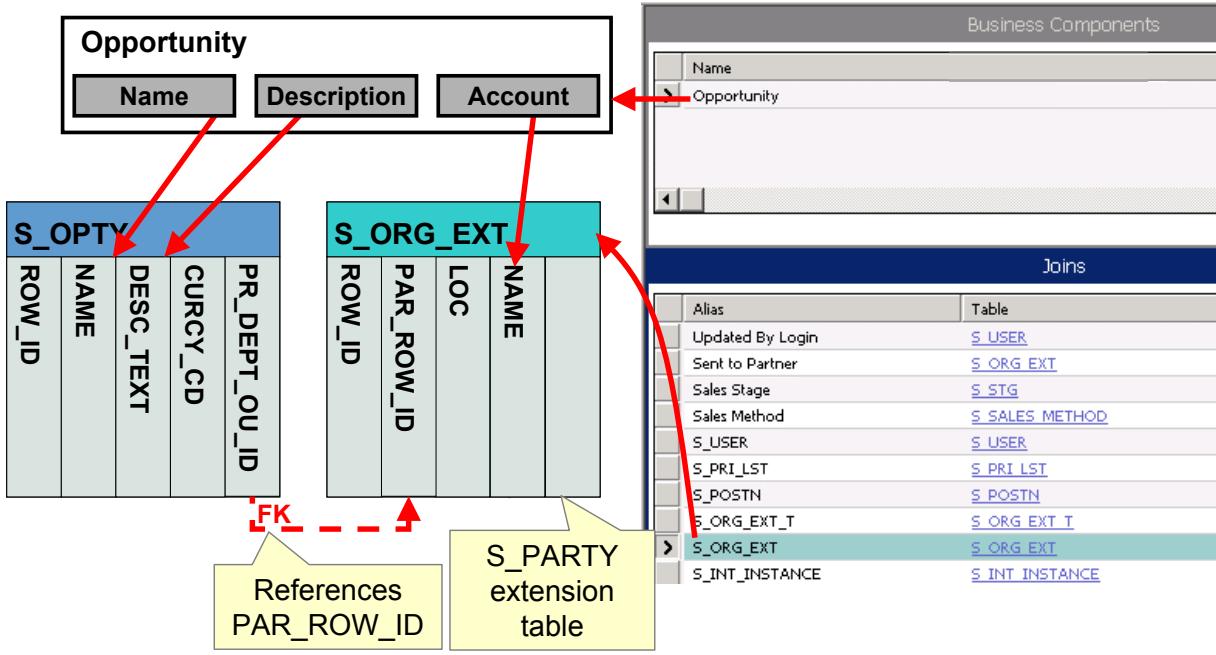
Copyright © 2007, Oracle. All rights reserved.

20 of 26

ORACLE®

Explicit Join Definition

- References the extension table that contains data of interest



Copyright © 2007, Oracle. All rights reserved.

21 of 26

ORACLE®

Explicit Join: Another Party Business Component

- Uses an explicit join to the party table, and not the implicit one
 - ▶ Example: Bringing parent account data into the Account business component for display in an Account applet

The screenshot shows the Siebel application interface with the following components:

- Top Navigation Bar:** Home, Accounts, Contacts, Opportunities, Quotes, Sales Orders, Service.
- Breadcrumb:** Accounts Home | Accounts List | Charts | Account Explorer | Service Explorer
- Table Header:** All Accounts Across Organizations, Menu, New, Delete, Query, Collaborate, Create Team Space.
- Table Data:** A grid with columns: Account Name, Site, Parent Account Name, Parent Account Site. The first column is highlighted with a red border.
- Bottom Diagram:** A diagram showing the relationship between the Account table and an Account applet. Red arrows point from the highlighted columns in the table to the corresponding fields in the Account applet: Name, Location, Parent Account Name, and Parent Account Location.

Account Name	Site	Parent Account Name	Parent Account Site
Harley-Davidson Europe Ltd.	HQ-WINDSOR-503337735	Harley-Davidson Holding Co Inc	HQ-MILWAUKEE-796217339
Kelly Industries	UK	Kelly Industries	HQ
Corporate I/T		Empire Chemical, Inc.	HQ
Chemical/Specialty	Brunswick, NH	Empire Chemical, Inc.	HQ
Petroleum/Retail		Empire Chemical, Inc.	HQ
Continental Distribution Europe	EMEA Headquarters - Paris	Continental Distribution	HQ
Continental Distribution	Americas Headquarters	Continental Distribution	HQ

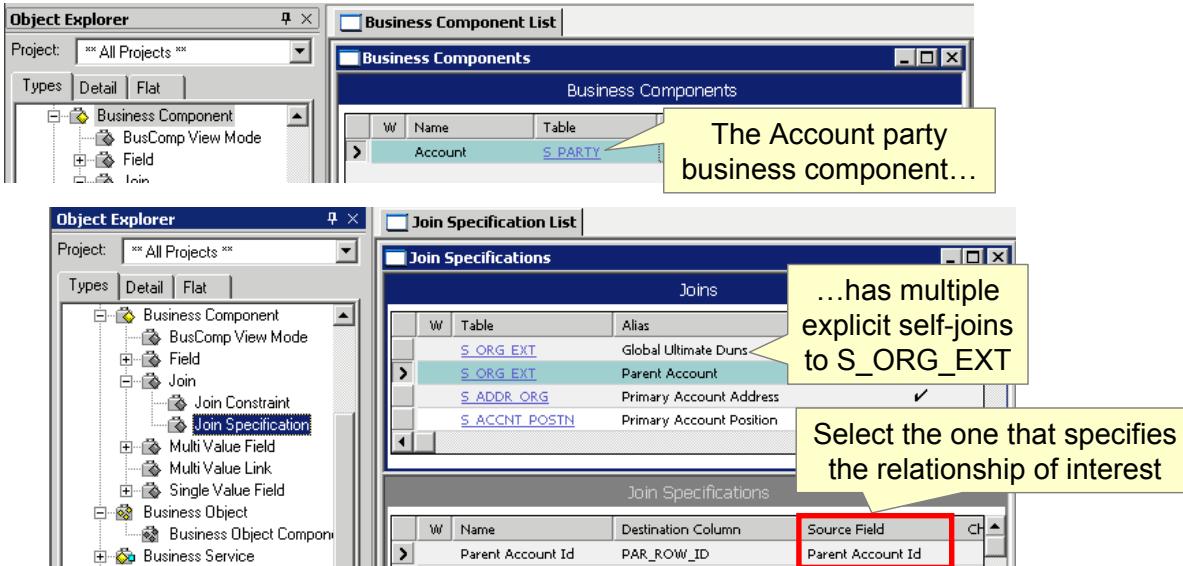
Copyright © 2007, Oracle. All rights reserved.

22 of 26

ORACLE®

Explicit Join Definition

- Select or create an explicit join to the desired S_PARTY extension table
 - ▶ Use a join specification to specify the relationship



Copyright © 2007, Oracle. All rights reserved.

23 of 26



Module Highlights

- Party types have dynamic or ad hoc relationships
- Party BCs create relationships between party types
- Party BCs are similar to standard BCs but data is stored in extension tables
- Eight prominent S_PARTY extension tables store data
- Person-related party BCs store data in S_CONTACT
- Organization-related party BCs store data in S_ORG_EXT
- Access Group, User List, and Position are party BCs
- Party implicit joins populate S_PARTY extension tables
- Explicit joins reference extension tables that contain data



Lab

- In the lab you will:
 - ▶ Examine how fields in a (non-party) business component map to columns in base and joined tables
 - ▶ Examine how fields in a party business component map to columns in S_PARTY, its extension tables, and joined tables

Copyright © 2007, Oracle. All rights reserved.

25 of 26

18

ORACLE®

Siebel 8.0 Essentials

Module 19: Siebel Business Objects

19

Copyright © 2007, Oracle. All rights reserved.

Oracle Integral & Oracle Academy Use Only





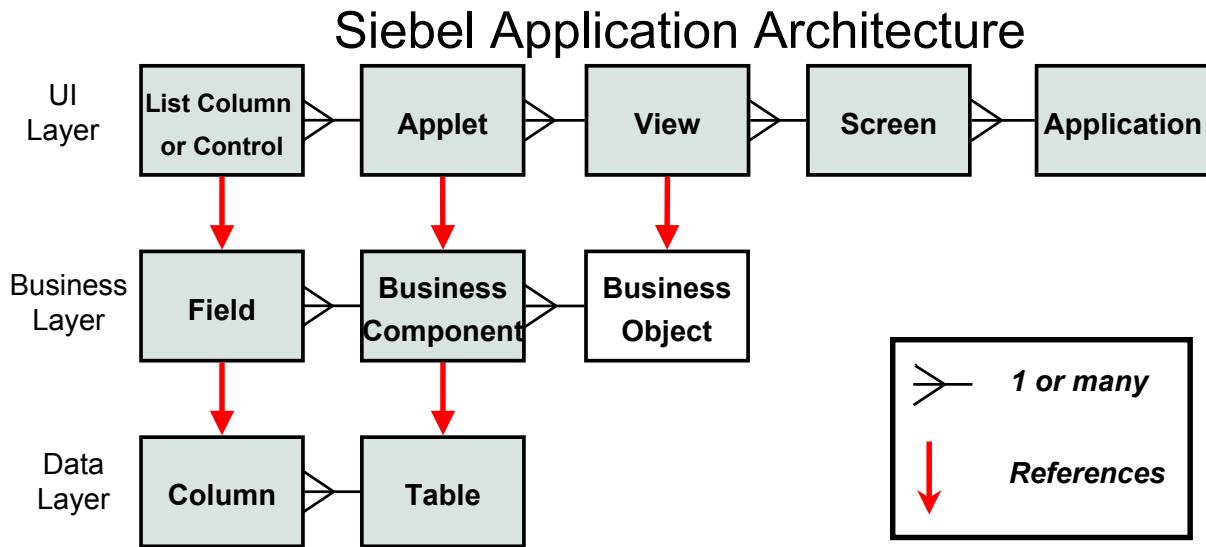
Module Objectives

- After completing this module you should be able to:
 - ▶ Describe how business objects focus data presented in the UI based on context
 - ▶ Describe how views reference business objects
 - ▶ Describe how links are used to relate parent business components to child business components
- Why you need to know:
 - ▶ Business objects enable you to configure your company's business logic

ORACLE®

Application Architecture: Business Objects (BO)

- Provide a way to organize BCs into major areas according to your business logic requirements
- Provide context to views



Copyright © 2007, Oracle. All rights reserved.

3 of 16

Reference

Configuring Siebel Business Applications: Configuring Business Objects

ORACLE®

Business Challenge

- Business components provide a way to group data according to business logic, but they do not address the need for context
 - ▶ For example, the opportunity BC defines how to retrieve opportunity data, but not related activity data

The screenshot shows a Siebel application window titled "Opportunity". The main title bar says "Opportunity: Opportunities List". Below it is a navigation bar with links: Home, Accounts, Contacts, Opportunities (highlighted), Quotes, Sales Orders, and Service. Underneath the navigation bar are links: Opportunities Home, Opportunities List, Charts, and Opportunity Explorer.

The main content area is titled "Laptops for Kaboom". It contains a form with fields for Opportunity Name (Laptops for Kaboom), Revenue (\$25,000.00), Currency (USD), Account (JRUBIN), Close Date (9/14/2006), Committed (unchecked), Sales Team (JRUBIN), Sales Stage (dropdown menu), Lead Quality (1-Excellent), Territories (dropdown menu), Probability % (70%), and Organization (PCS Americas). Below the form is a tab bar with "More Info", "Activities" (highlighted), "Assessments", "Attachments", "Contacts", "Quotes", and "Revenue".

A yellow callout box with a red question mark icon points to the "Activities" tab. The text inside the callout box reads: "How is it that the Activities view shows only activities related to the Kaboom opportunity?".

Copyright © 2007, Oracle. All rights reserved.

4 of 16

ORACLE®

Business Solution: Business Objects Provide Focus

- Business objects contain relationships used to access records from a child business component
 - ▶ Identifies records to display depending on context in which they are viewed
 - ▶ Ensures that only records related to parent BCs are returned from queries

The screenshot shows a Siebel application window titled "Laptops for Kaboom". At the top, there are fields for Opportunity Name, Account, Sales Team, Territories, Revenue, Close Date, Currency, Committed, Sales Stage, Lead Quality, and Probability %. Below this, a navigation bar includes "More Info", "Activities", "Assessments", "Attachments", "Contacts", "Quotes", and "P...". A red arrow points from the "Activities" tab to a callout box. The "Activities" section displays a table with columns: Created, Created By, Type, and Description. The table contains three rows of activity data:

Created	Created By	Type	Description
9/14/2006 7:23:29 PM	JRUBIN	Assessment	Call to perform initial assessment
9/14/2006 7:25:12 PM	JRUBIN	In Store Visit	Visit Kaboom to validate assessment
9/14/2006 7:27:29 PM	JRUBIN	Demonstration	Visit Kaboom to demo assembled configuration

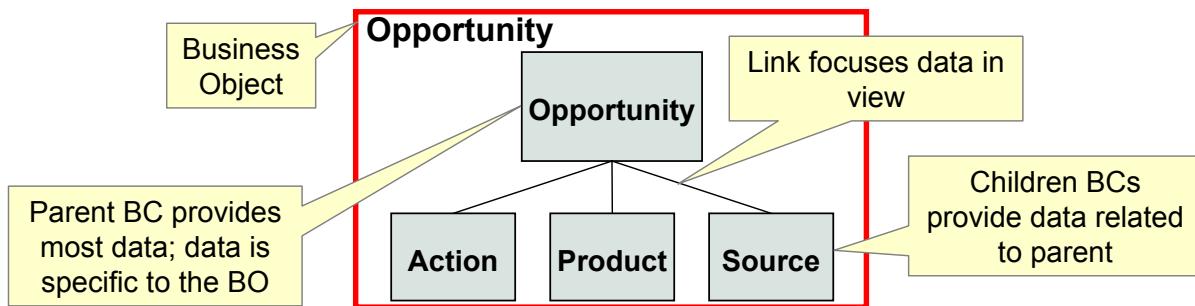
A yellow callout box with a black border and a black arrow pointing to the third row of the table contains the text: "Activities pertaining to Kaboom opportunity, and only Kaboom, are displayed".

Copyright © 2007, Oracle. All rights reserved.

5 of 16

Business Objects

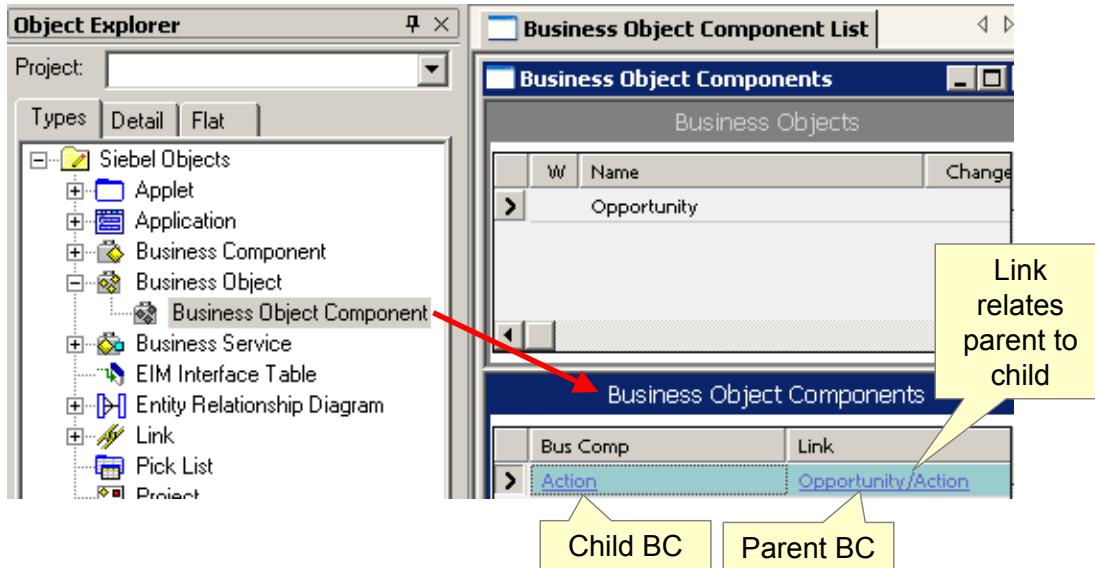
- Are a collection of related business components that represent a major area of the business
 - ▶ Contain parent and child business components
 - ▶ Relate parent and child components via links
 - ▶ Links focus records displayed based on the context in which they are viewed
- Provide a container for grouping business components
- Provide context for views



ORACLE®

Business Object Component

- Contains Link and BusComp properties that relate the parent and child business components to each other

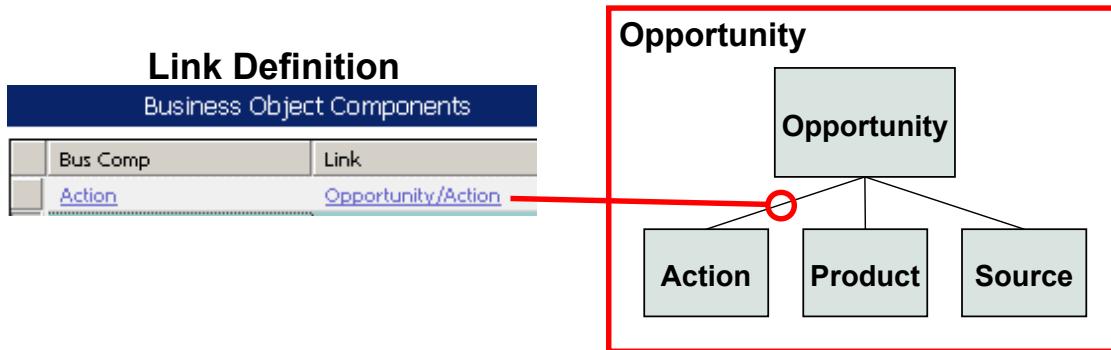


Copyright © 2007, Oracle. All rights reserved.

7 of 16

Link Definition

- Identifies the PK/FK relationship
 - ▶ Identifies which records to retrieve from the child business component
 - ▶ Identifies the foreign keys to populate when new child records are created
- Is used with both 1:M and M:M relationships between parent and child data



Copyright © 2007, Oracle. All rights reserved.

8 of 16

1:M Link

- Used for 1:M relationship between parent and child business components

Link Definition

The screenshot shows the Siebel Object Explorer interface. On the left, the 'Object Explorer' pane displays a tree structure of Siebel objects including Applet, Application, Business Component, Business Object, Business Object Component, Business Service, EIM Interface Table, Entity Relationship Diagram, Link, and Pick List. The 'Project' dropdown is set to 'All Projects'. In the center, the 'Business Object Component List' shows a table with columns 'Name' and 'Type'. It lists 'Opportunity' under 'Business Objects' and 'Action' under 'Business Object Components'. Below this is a 'Business Object Components' list with tabs for 'Bus Comp', 'Link', and 'Action'. The 'Action' tab is currently selected.

Link [Opportunity/Action]	
Alphabetic Categorized	
Cascade Delete	None
Child Business Component	Action
Comments	
Destination Field	Opportunity Id
Inactive	FALSE
Inter Child Column	
Inter Child Delete	
Inter Parent Column	
Inter Table	
Module	
Name	Opportunity/Action
No Associate	FALSE
No Delete	FALSE
No Insert	FALSE
No Inter Delete	FALSE
No Update	FALSE
Object Language Locked	
Object Locked	FAL
Object Locked By Name	
Object Locked Date	
Parent Business Component	Opp
Primary Id Field	
Search Specification	
Sort Spec	
Source Field	

Copyright © 2007, Oracle. All rights reserved.

9 of 16

M:M Link

- Used for M:M relationship between parent and child business components
- Uses an intersection table to resolve the link

Business Object Components

Bus Comp	Link
Source	Opportunity/Source

Properties

Link [Opportunity/Source]

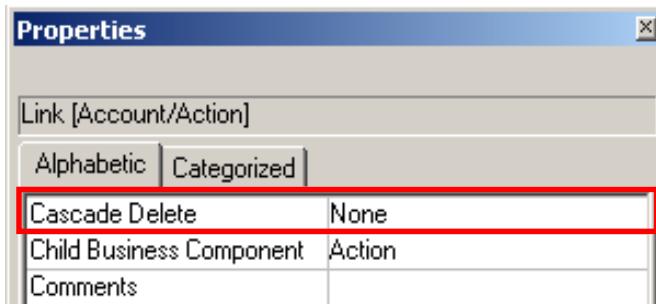
Cascade Delete	None
Child Business Comp	Source
Comments	
Destination Field	
Inactive	FALSE
Inter Child Column	SRC_ID
Inter Child Delete	FALSE
Inter Parent Column	OPTY_ID
Inter Table	S_OPTY_SRC

Intersection table with FK columns



Cascade Delete Property

- The Cascade Delete property of a link specifies whether child records of a 1:M relationship are deleted when the parent record is deleted
 - ▶ None: Neither deletes nor clears the foreign key column
 - ▶ Clear: Does not delete, but clears the foreign key column
 - ▶ Delete: If a parent record is deleted, all child records are deleted



Copyright © 2007, Oracle. All rights reserved.

11 of 16

ORACLE®

Cascade Delete Property continued

- Determined by your business logic
 - ▶ Should all activities be deleted if parent contact is deleted?
 - ▶ Should all child opportunities be deleted if the parent account is deleted?

The screenshot shows the Siebel interface with two main windows. The top window is an 'Account' view for 'AT'. It displays fields like Account Name (AT&T), Address (350 Main Street), City (Edison), Zip Code (08675), Site (Edison, NJ), Account Team (JRUBIN), Main Phone # (800) 788-1000, Main Fax # (800) 788-1500, and URL (www.att.com). A yellow callout box labeled 'Deletion cascades' points to the 'Account' field. The bottom window is an 'Opportunities' list view. It shows two entries: 'Fast Ethernet NIC PCI 10/100 Server' and 'Enterprise Agreement - AT&T', both associated with 'AT&T'. A large red 'X' is drawn across the entire bottom window, indicating that opportunities will be deleted when the account is deleted.

Properties

Link [Account/Action]

Alphabetic | Categorized

Cascade Delete Delete

Child Business Component Action

Comments

AT

Account Name: * AT&T

Address: 350 Main Street

City: Edison

Zip Code: 08675

Site: Edison, NJ

Account Team: JRUBIN

Address Line 2:

State: NJ

Country: US

Main Phone #: (800) 788-1000

Main Fax #: (800) 788-1500

URL: www.att.com

More Info Activities Attachments Contacts Notes Opportunities Service Requests Orders Quotes

Opportunity Name	Account	Revenue	Committed	Team Space	Sales Stage
> Fast Ethernet NIC PCI 10/100 Server	AT&T	\$150,000.00	✓		09 - Closed Won
Enterprise Agreement - AT&T	AT&T	\$1,175,000.00			01 - Prospecting

Copyright © 2007, Oracle. All rights reserved.

12 of 16

ORACLE®

Grandchild Business Components

- Business objects may include grandchild business components
- Used in parent-child-grandchild views

The screenshot illustrates the navigation through three levels of business components:

- Parent BC: Account**: The top section shows the account details for "Art.net". A yellow callout box labeled "Activity plans for Art.net" points to the "Activity Plans" tab.
- Child BC: Activity Plan**: The middle section shows the activity plan for "1/8/2007 7:40:08 AI Quarterly Status Che". A yellow callout box labeled "Activities for selected Art.net activity plan" points to the "Activities" tab.
- Grandchild BC: Activity Plan Action**: The bottom section shows the list of activities under the selected plan, including "Check up on progre: Call" and "Send out survey car Email - Outbound".

Arrows indicate the flow from the Parent BC to the Child BC, and then to the Grandchild BC.

Description	Type	Start	End	Status	Priority	Employees
Check up on progre: Call	Call	2/8/2007 3:42:37 PM	2/13/2007 3:42:37 PM	Unscheduled	2-High	
Send out survey car	Email - Outbound	2/27/2007 3:43:59 PM	2/28/2007 3:43:59 PM	Scheduled	3-Medium	

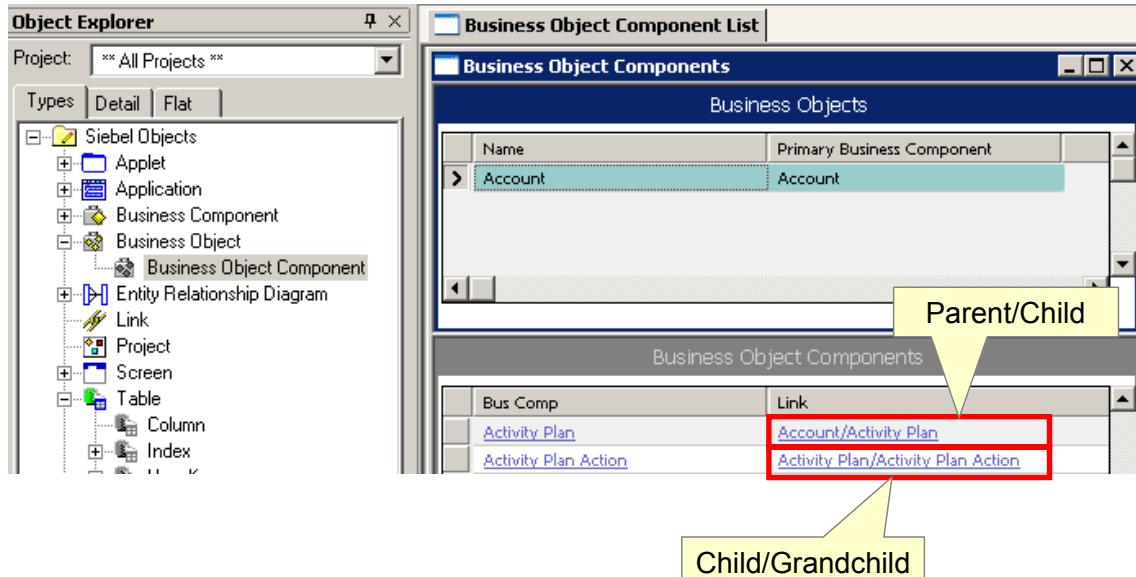
Copyright © 2007, Oracle. All rights reserved.

13 of 16

ORACLE®

Links for Grandchild Data

- Specify how the grandchild and child business component are related
 - ▶ Used to retrieve grandchild records on parent-child-grandchild views



Copyright © 2007, Oracle. All rights reserved.

14 of 16



Module Highlights

- Business objects
 - ▶ Provide a way to group BCs according to business logic
 - ▶ Provide context to views
 - ▶ Contain parent and child BCs associated via links
- Links provide a way to focus records based on context
- Links relate parent and child BCs
- Link definitions specify which records to retrieve from child BCs
- A 1:M link is used for a 1:M relationship between parent and child BCs
- A M:M link is used for a M:M relationship between parent and child BCs
 - ▶ A M:M link uses an intersection table to resolve the link
- BOs may include grandchild BCs

ORACLE®

Siebel 8.0 Essentials

Module 20: Configuration Strategy

20

Copyright © 2007, Oracle. All rights reserved.

20
Oracle Internal & Oracle Academy Use Only



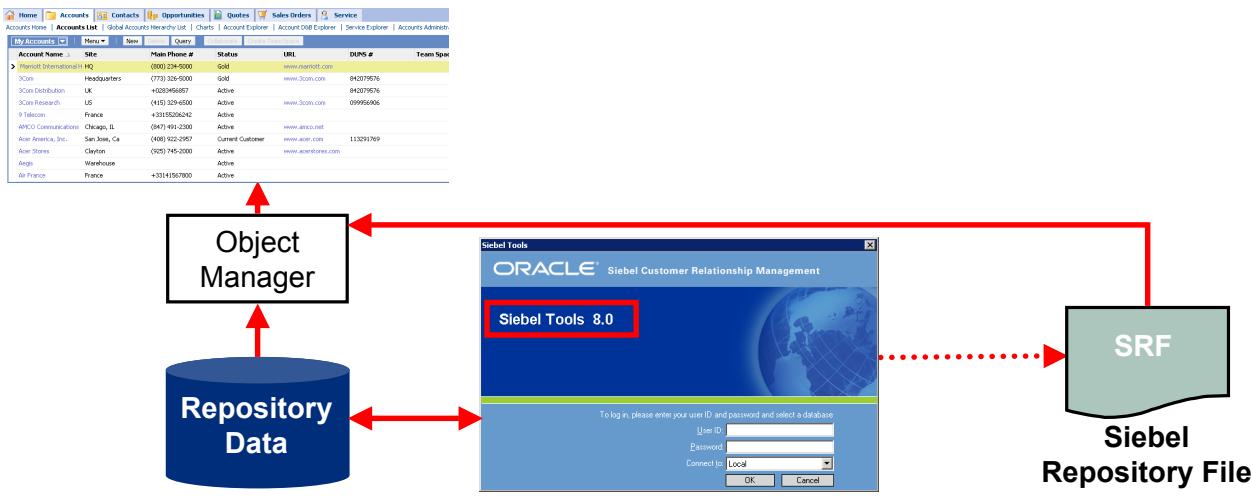
Module Objectives

- After completing this module you should be able to:
 - ▶ List the critical elements of the Siebel configuration strategy
- Why you need to know:
 - ▶ Following a reasonable configuration strategy can expedite your configuration efforts and make your configured application robust and upgradeable



Configuring a Siebel Application

- Configuring is the process of using Siebel Tools to modify an as-delivered Siebel application to meet business needs
- Object definitions are edited and created
 - ▶ Developers do not modify code in siebel.exe
 - ▶ Developers do not write SQL directly



Copyright © 2007, Oracle. All rights reserved.

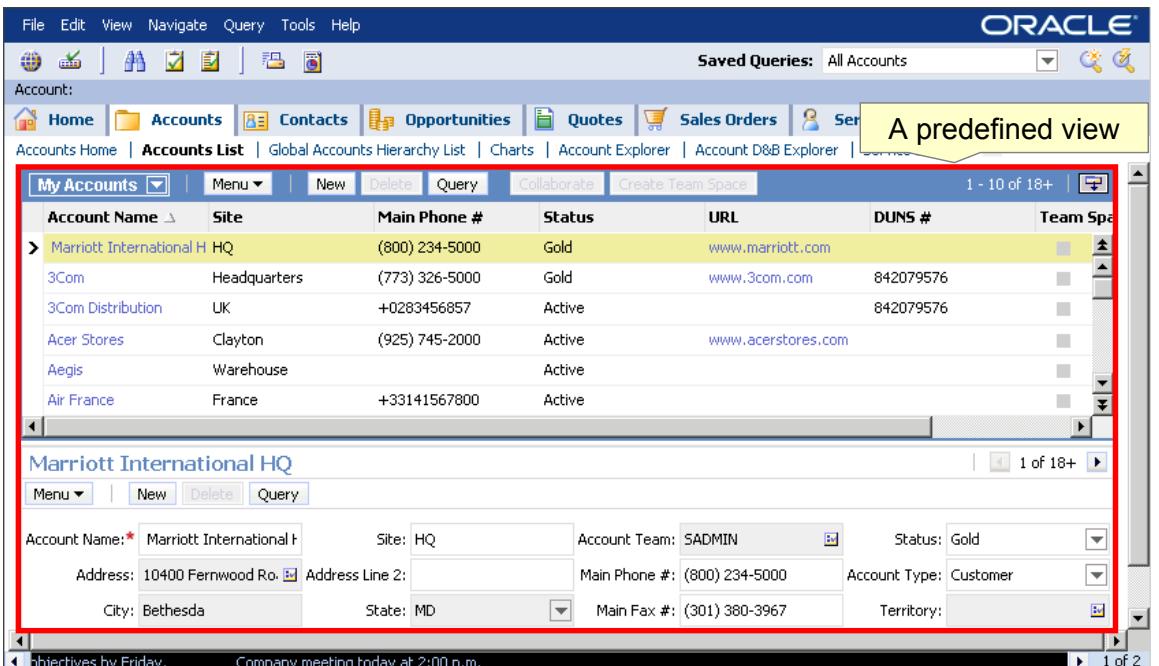
3 of 16

Reference

Using Siebel Tools

Developing and Deploying Siebel Business Applications: Setting Up a Developer's Local Database

Siebel Applications



The Siebel Applications interface is shown, featuring a top navigation bar with File, Edit, View, Navigate, Query, Tools, and Help. A toolbar below includes icons for Home, Accounts, Contacts, Opportunities, Quotes, Sales Orders, and Services. The main area is titled "Saved Queries: All Accounts". A yellow callout box points to the title bar with the text "A predefined view". Below the title bar, a sub-menu bar shows "My Accounts" selected. The main content area displays a grid of account records. One record for "Marriott International HQ" is highlighted in yellow and expanded into a detailed view. This expanded view includes fields for Account Name, Site, Main Phone #, Status, URL, DUNS #, and Team Space. The detailed view for Marriott International HQ shows the following information:

Account Name:	Marriott International HQ	Site:	HQ	Account Team:	SADMIN	Status:	Gold
Address:	10400 Fernwood Rd.	Address Line 2:		Main Phone #:	(800) 234-5000	Account Type:	Customer
City:	Bethesda	State:	MD	Main Fax #:	(301) 380-3967	Territory:	

Below the expanded view, there are two status messages: "Objectives by Friday." and "Company meeting today at 2:00 p.m."

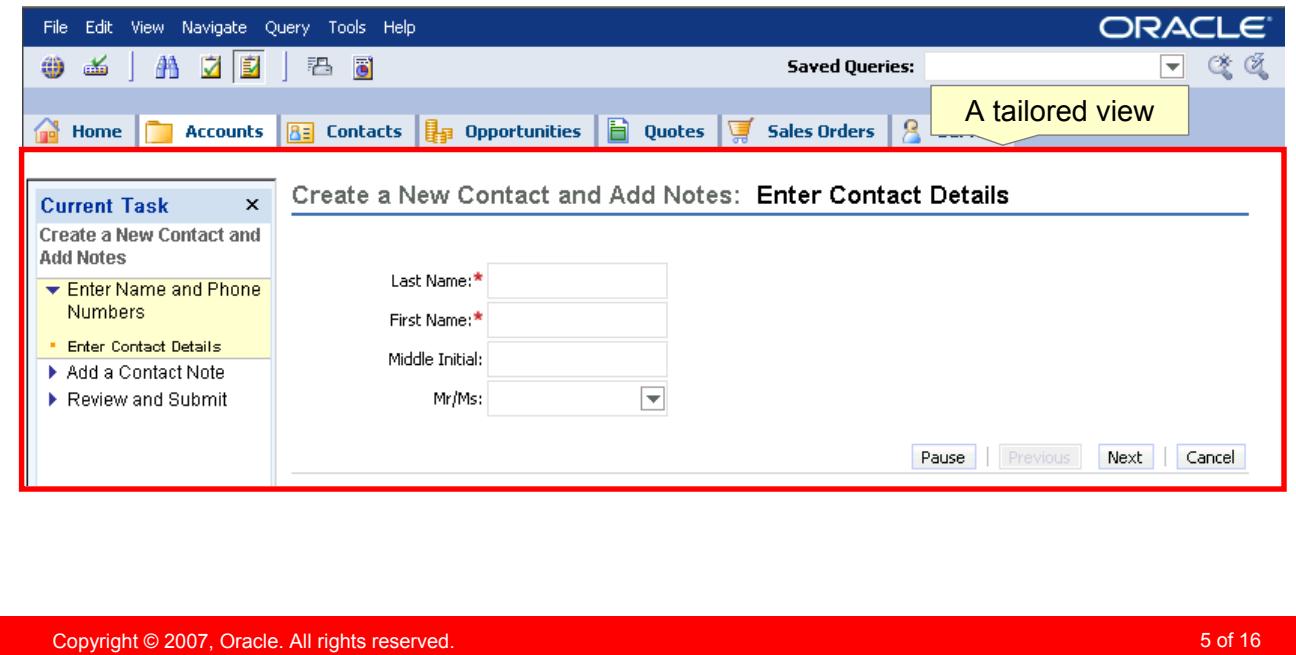
Copyright © 2007, Oracle. All rights reserved.

4 of 16

ORACLE®

Tailoring the Logical User Interface

- Developers tailor the as-delivered Siebel screens, views, lists, and forms to better support users' business needs



Copyright © 2007, Oracle. All rights reserved.

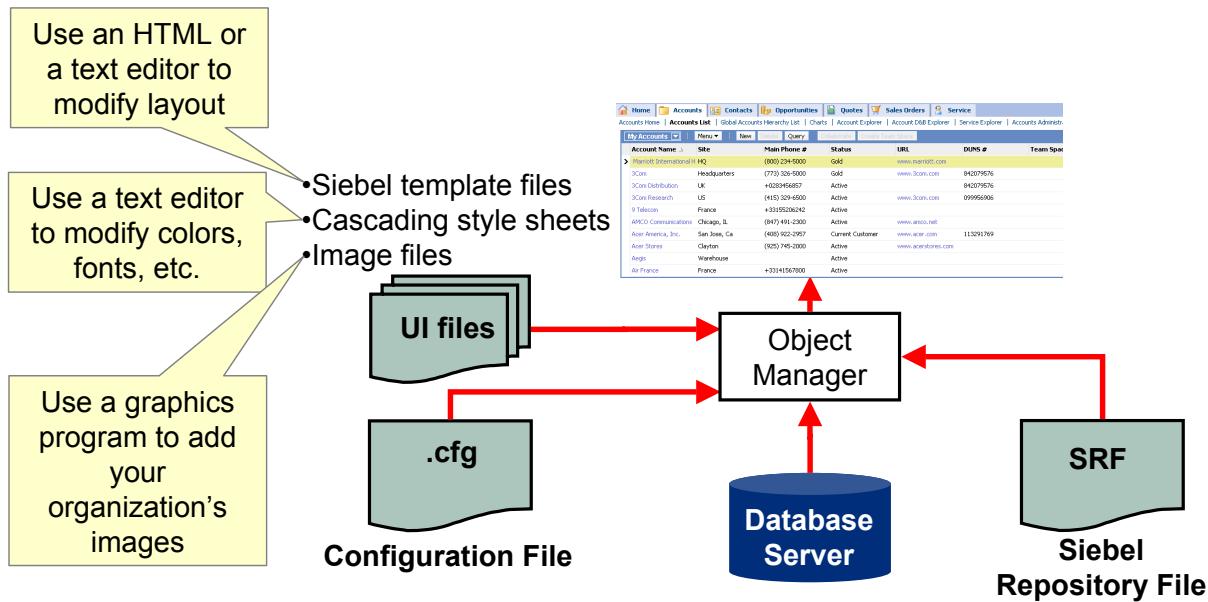
5 of 16

20
Oracle Internal & Oracle Academy Use Only

ORACLE®

Tailoring the Physical User Interface

- Use the existing physical UI files whenever possible
- When necessary, copy existing files and modify as needed



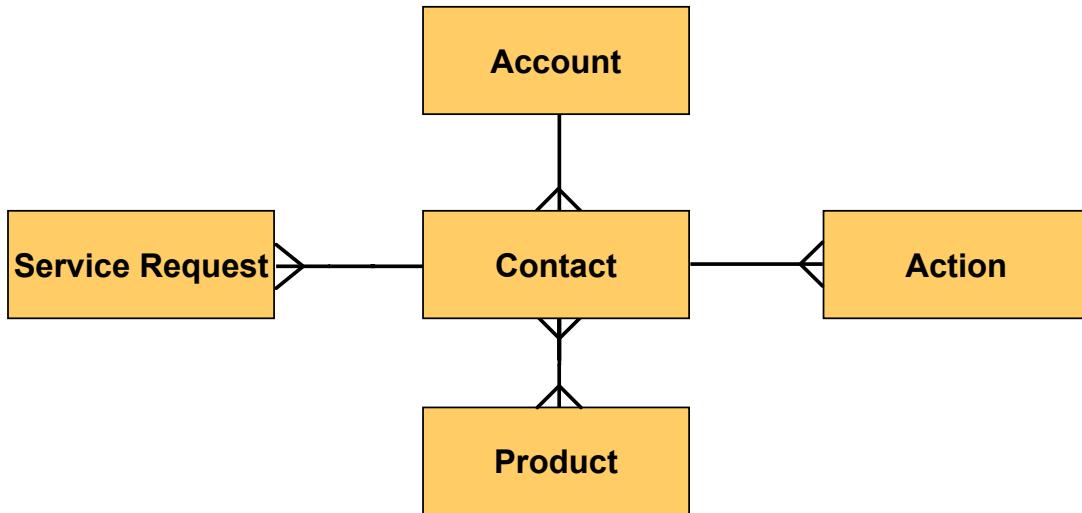
Copyright © 2007, Oracle. All rights reserved.

6 of 16

ORACLE®

Siebel Business Entities

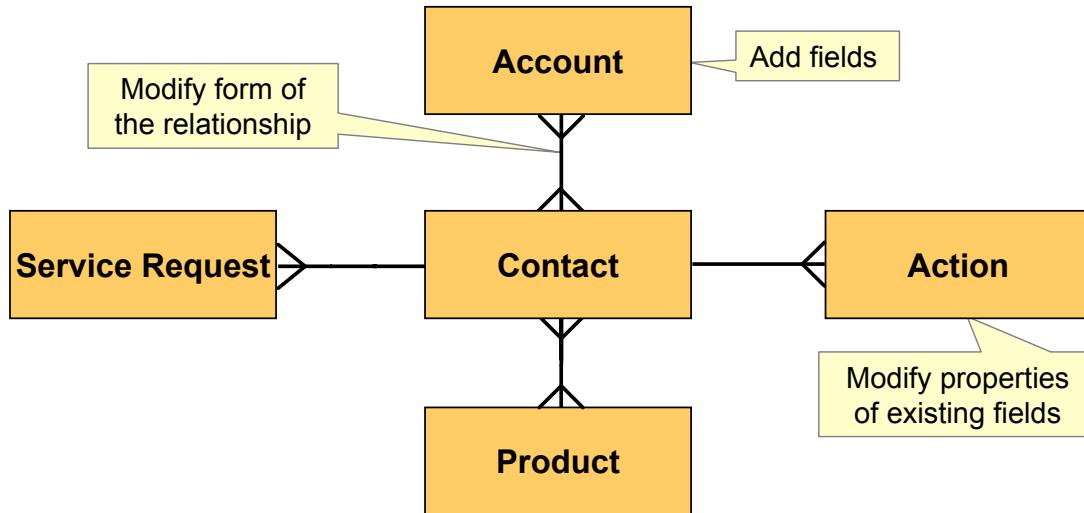
- Siebel as-delivered applications utilize a set of Siebel business components that implement the defined business logic



ORACLE®

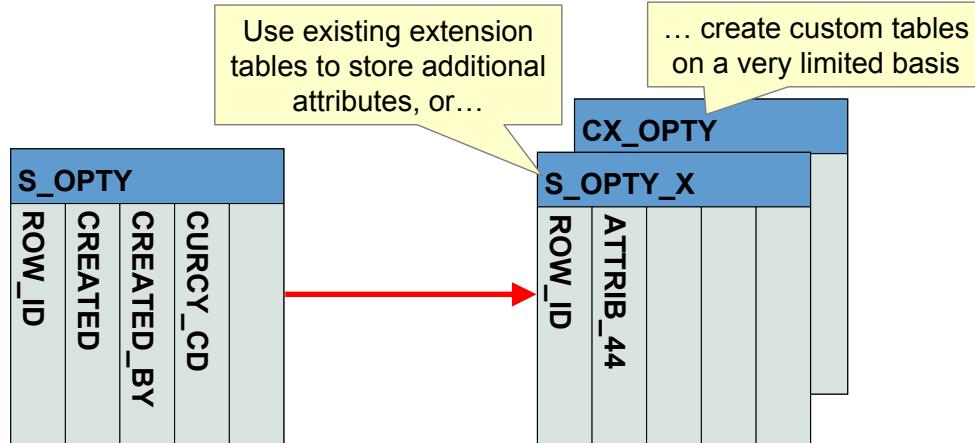
Tailoring the Business Logic

- Developers also tailor the application by modifying the definitions of the business components to implement the business logic appropriate to the users' organization



Extending the Data Layer

- Siebel as-delivered applications contain a large, predefined set of database tables
- Developers can tailor the application by extending the set of database tables
 - ▶ Recommendation is to perform this in a limited and controlled manner





Configuration Strategy

- Make minimal changes to the as-delivered application
- Use existing object definitions in the as-delivered repository whenever possible
 - ▶ Ensures that a new configuration can be upgraded with minimal effort
- Modify definitions as required, rather than creating new ones
 - ▶ Creating new object definitions can lead to redundant configuration and increase the maintenance effort

ORACLE®

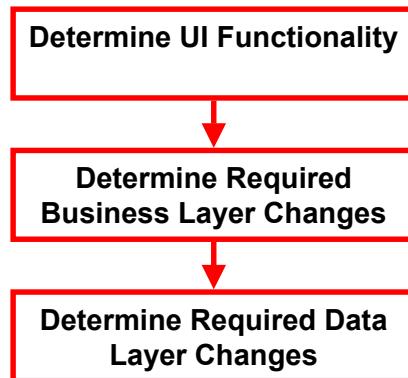
Configuration Strategy Continued

- Do not delete, make inactive, or rename seemingly unused object definitions
 - ▶ Other object definitions might reference them
 - ▶ Deleting does not save memory, storage space, or improve performance
- Use existing template files
 - ▶ Modify only when changes are essential
 - ▶ Modifying a template for one view or applet can have unexpected consequences if another view or applet uses the same template

ORACLE®

Configuration Strategy Continued

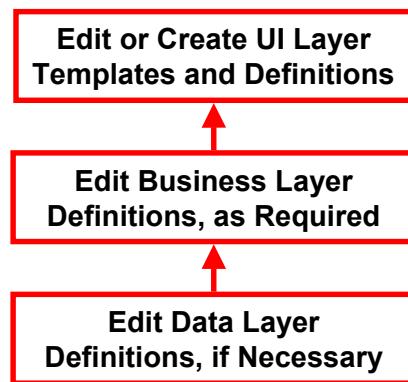
- Plan your configuration project from the top down
 - ▶ First, determine UI and application functionality
 - ▶ Then, determine what changes are necessary at the Business layer to implement UI functionality
 - ▶ Finally, determine what changes are necessary at the Data layer to implement Business layer changes
 - Keep changes to a minimum





Configuration Strategy Continued

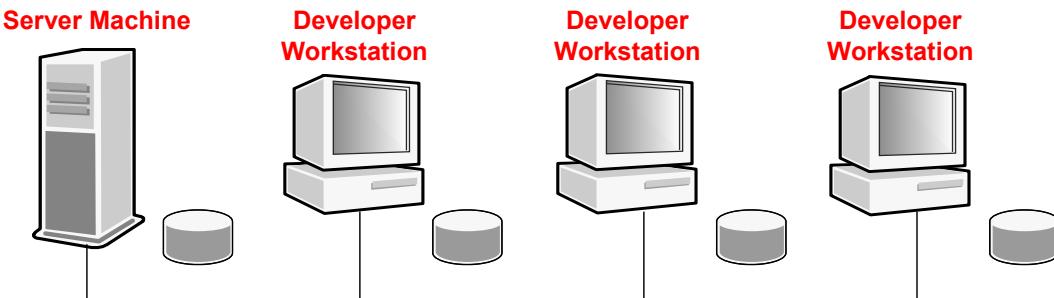
- Make changes from the bottom up
 - ▶ First, edit Data layer definitions, if necessary
 - ▶ Then, edit Business layer definitions, as required
 - ▶ Finally, edit or create the templates and UI layer definitions to display the data correctly





Create a Separate Development Environment

- To isolate the development effort from the enterprise's production database, set up a development environment
- Each developer or group can work on a different aspect of the development effort
 - ▶ Use Siebel-supplied mechanisms to separate the development effort into projects
- Test all customization and extensions thoroughly before deploying to end users



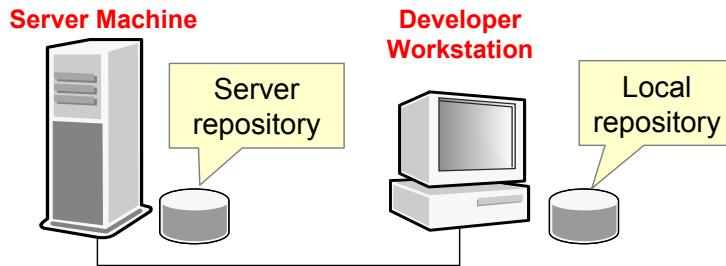
Copyright © 2007, Oracle. All rights reserved.

14 of 16



Develop on the Local Repository

- Always make changes to the object definitions in the local repository
 - ▶ Cannot undo or back out changes when made directly on the server
 - ▶ Changes made directly on the server are immediately available to other developers
 - Incomplete changes on the server cause problems
- Use Siebel-supplied mechanisms to copy definitions between server and local databases



Copyright © 2007, Oracle. All rights reserved.

15 of 16



Module Highlights

- Configuring is the process of using Siebel Tools to modify an as-delivered Siebel application
- Recommended configuration strategy:
 - ▶ Tailor existing logical UI, physical UI, and business entities to minimize development and support effort
 - ▶ Do not delete, make inactive or rename unused object definitions
 - ▶ Use existing template files
- Plan configuration project from the top down, make changes from the bottom up
- Development and production environments are separate
 - ▶ Enables developer or group to work independently
 - ▶ Enables developer to test customizations before deploying



Siebel 8.0 Essentials

Module 21: The Configuration Process

21

Copyright © 2007, Oracle. All rights reserved.

21

Oracle Internal & Oracle Academy Use Only



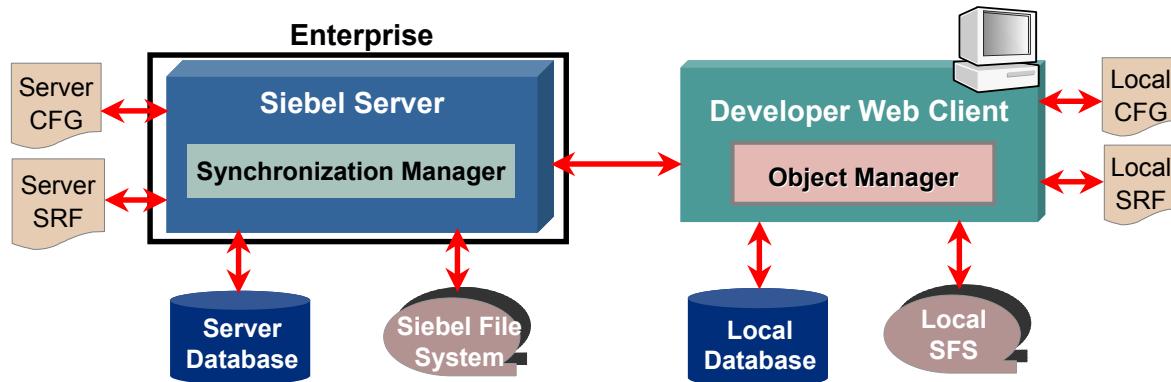
Module Objectives

- After completing this module you should be able to:
 - ▶ Describe the development environment architecture
 - ▶ Set up the development environment
- Why you need to know:
 - ▶ Being able to set up the development environment is critical to carrying out a successful configuration



The Development Environment

- Siebel Remote is used to establish a local development environment
- Siebel Developer Web Client provides full Siebel application functionality in a local environment
 - ▶ Accesses a local DB containing a subset of the server database
 - ▶ Synchronizes with the server database, as necessary
 - ▶ Implements a development and testing environment for individual developers



Copyright © 2007, Oracle. All rights reserved.

3 of 20

Reference

Using Siebel Tools

Siebel Remote and Replication Manager Administration Guide

Developing and Deploying Siebel Business Applications: Setting Up a Developer's Local Database

Procedure for Setting Up the Development Environment

1. Enable the Development Environment
2. Create the Developer
3. Create a User Login
4. Generate the Local Database Template
5. Extract User Data
6. Modify and Verify Configuration Settings
7. Initialize Local Database
8. Extract Repository Data

Enable the Development Environment

- Enable the Siebel Remote component group to enable development using a local database
 - ▶ After enabling a component group, do not forget to restart the Siebel Server and synchronize components

The screenshot shows the Siebel Administration - Server Configuration interface. The top navigation bar includes Home, Accounts, Contacts, Opportunities, Quotes, Sales Orders, Service, Administration - Server Configuration, Enterprise Explorer, Enterprises, Servers, and Job Templates. Below this is the Enterprise Servers menu with options for Menu, Backup Enterprise, and 1 - 1 of 1. The main content area displays the Component Groups screen. It has tabs for Component Groups, Component Definitions, System Alerts, Profile Configuration, Parameters, and Synchronize. Under Component Groups, there is a table with columns: Component Group, Alias, Number of Components, Enable state, and Description. Two rows are visible: 'MWC Real Time Sync' (RTSRemote) with alias 'RTSRemote', 4 components, Enabled state, and description 'Siebel Remote Real Time Sync Components'; and 'Siebel Remote' (Remote) with alias 'Remote', 8 components, Enabled state, and description 'Siebel Remote Components'. A yellow callout with the text 'Enable Siebel Remote' points to the 'Enable' button for the 'Siebel Remote' row. Below this is the Component Group Assignments screen, which lists components assigned to servers. One entry is shown: 'Database Extract' (DbExtract) assigned to 'SUsrvr'. The bottom of the page includes a copyright notice 'Copyright © 2007, Oracle. All rights reserved.' and a page number '5 of 20'.

2/8

ORACLE®

Create the Developer

- Add the developer as an employee
- Associate a position and responsibility
- Register the developer as a mobile client

The image displays two screenshots of the Siebel application interface.

Screenshot 1: Employee Record Creation

This screenshot shows the "Employees" list view. A new record for "Penguin" has been created with the following details:

Last Name	First Name	User ID	Responsibility	Position
Penguin	Pat	PPENGUIN	ABC Developer	ABC Developer

A yellow callout box labeled "Add employee record" points to the "New" button in the top navigation bar.

Screenshot 2: Mobile Client Registration

This screenshot shows the "Mobile Clients" list view. A new mobile client record for "PPENGUIN" has been created with the following details:

Mobile Client	User ID	Routing Model	User Last Name
PPENGUIN	PPENGUIN	MOBILE CLIENT - ST	Penguin

A yellow callout box labeled "Register mobile client" points to the "New" button in the top navigation bar.

Both screenshots include standard Siebel navigation bars at the top and bottom.

Copyright © 2007, Oracle. All rights reserved.

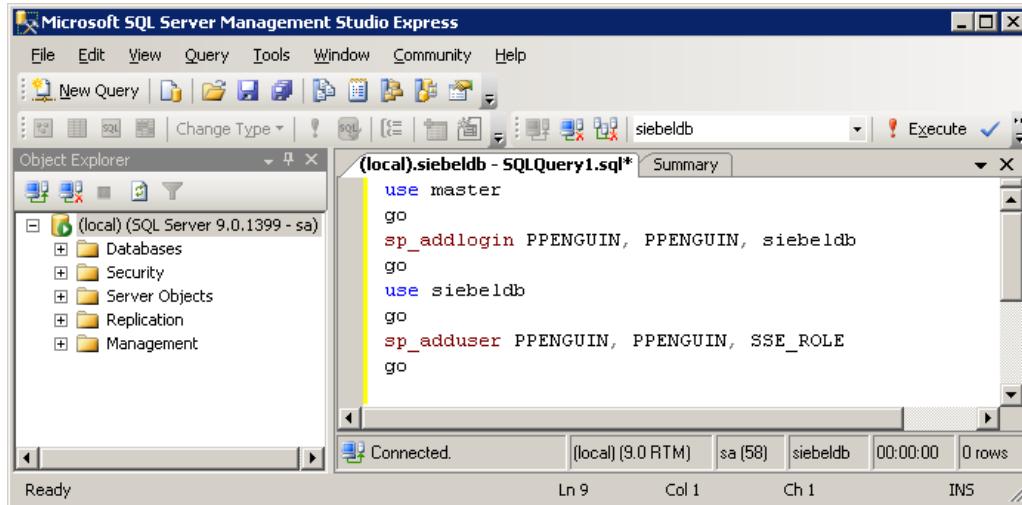
6 of 20

3/8

ORACLE®

Create a User Login

- If necessary, create a corresponding database login using RDBMS tools and the grantusr.sql script as a template
- Example:



The screenshot shows the Microsoft SQL Server Management Studio Express interface. The Object Explorer on the left shows a connection to '(local) (SQL Server 9.0.1399 - sa)' with nodes for Databases, Security, Server Objects, Replication, and Management. The central pane displays a query window titled '(local).siebeldb - SQLQuery1.sql*' containing the following T-SQL script:

```
use master
go
sp_addlogin PPENGUIN, PPENGUIN, siebeldb
go
use siebeldb
go
sp_adduser PPENGUIN, PPENGUIN, SSE_ROLE
go
```

The status bar at the bottom indicates 'Connected.', '(local) (9.0 RTM)', 'sa (58)', 'siebeldb', '00:00:00', '0 rows', and 'INS'.

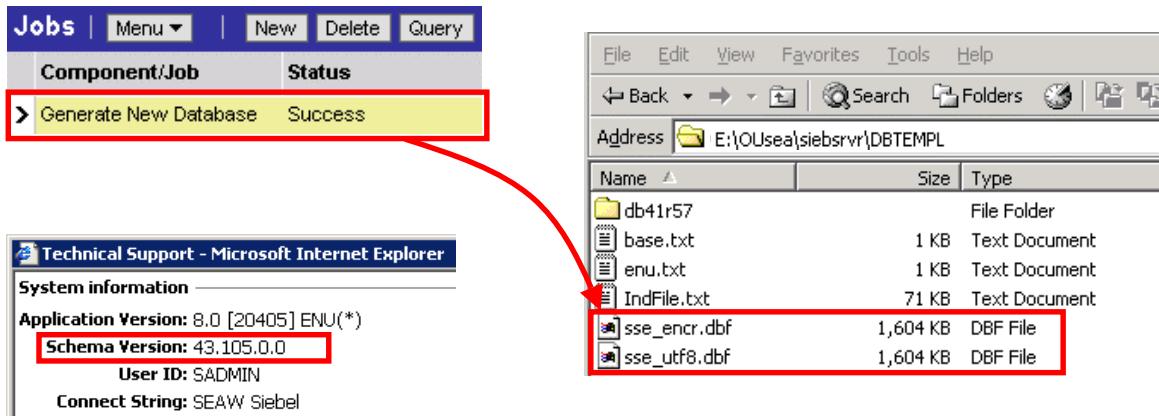
Screen shot from Microsoft SQL Server Management Studio Express copyright © Microsoft Corporation

Copyright © 2007, Oracle. All rights reserved.

7 of 20

Generate the Local Database Template

- Run the Generate New Database server task to generate a local database template
 - ▶ Snapshot of the current version of the database schema (tables, columns, indexes, and so forth)



Copyright © 2007, Oracle. All rights reserved.

8 of 20

Extract User Data

- Run the Database Extract server task to extract *user* data
 - ▶ Generates compressed data files
 - Stored on the server
 - ▶ User data is extracted for the user(s) specified in the job parameters

Jobs:

The screenshot shows the Siebel interface with the following details:

- Header:** Home, Accounts, Contacts, Opportunities, Quotes, Sales Orders, Service, Administration - Server Management.
- Sub-Header:** Servers, Components, Jobs, Tasks, Sessions.
- Job List:**

State (Icon)	ID	Component/Job	Execution Server Mode	Status
1-3PSP	Database Extract	OUsrvr	Asynchronous	Success
- Message Box:** A yellow box contains the text "Data is extracted to the specified mobile user's directory".
- File Explorer:**
 - Address:** E:\OUsea\siebsrvr\docking\PPENGUIN
 - Folders:** siebsrvr (expanded), ACTUATE, ADMIN, BIN.
 - Content:** A list of files in the PPENGUIN folder:

Name	Type	Date Modified
inbox	File Folder	2/5/2007 12:55 PM
outbox	File Folder	2/5/2007 1:02 PM
00000000.log	Text Document	2/5/2007 1:02 PM
dobjinst.dbf	DBF File	2/5/2007 1:02 PM

Copyright © 2007, Oracle. All rights reserved. 9 of 20

Modify Configuration Settings

- Modify configuration information for Siebel Tools in ..\tools\BIN\<Language Code>\tools.cfg
 - ▶ For example, ..\tools\BIN\ENU\tools.cfg
- Modify configuration information for Siebel Developer Web Client in ..\client\BIN\<Language Code>\<application>.cfg
 - ▶ For example, ..\client\BIN\ENU\uagent.cfg for the Siebel Call Center Developer Web client
- Both configuration files hold critical information to ensure proper behavior of local executables:
 - ▶ Native client connection information for the local database
 - ▶ ODBC connection for the server
 - ▶ ODBC connection for the local database
 - ▶ Location of the local Siebel file system

Example Configuration Settings

- Example: tools.cfg (edited for length)

```

tools.cfg - Notepad
File Edit Format View Help

[Siebel]
RepositoryFile      = siebel.srf
ApplicationName     = Siebel Tools
ClientRootDir        = c:\0usea\tools
LocalDbODBCDataSource= SSD Local db default instance
ServerDbODBCDataSource= SSD default instance

[Datasources]
Local                = Local
Sample               = Sample
ServerDataSrc        = Server

[Local]
Docked              = FALSE
ConnectionString    = c:\0usea\client\local\sse_data.dbf -q -m -x NONE -gp 4096 -c256m -ch256m
Tableowner           = SIEBEL
DockedDBFilename    = CHANGE_ME
DockConnstring      = CHANGE_ME
EnterpriseServer    = Siebel

[ServerDataSrc]
Docked              = TRUE
ConnectionString    = SSD default instance
Tableowner          = dbo
Filesystem          = c:\siebfile
GatewayAddress      = localhost
EnterpriseServer    = Siebel

```

Overall configuration: Name of SRF file, application name, root directory, and ODBC connections to local and server databases for testing connectivity

Available data sources. Add more if needed.

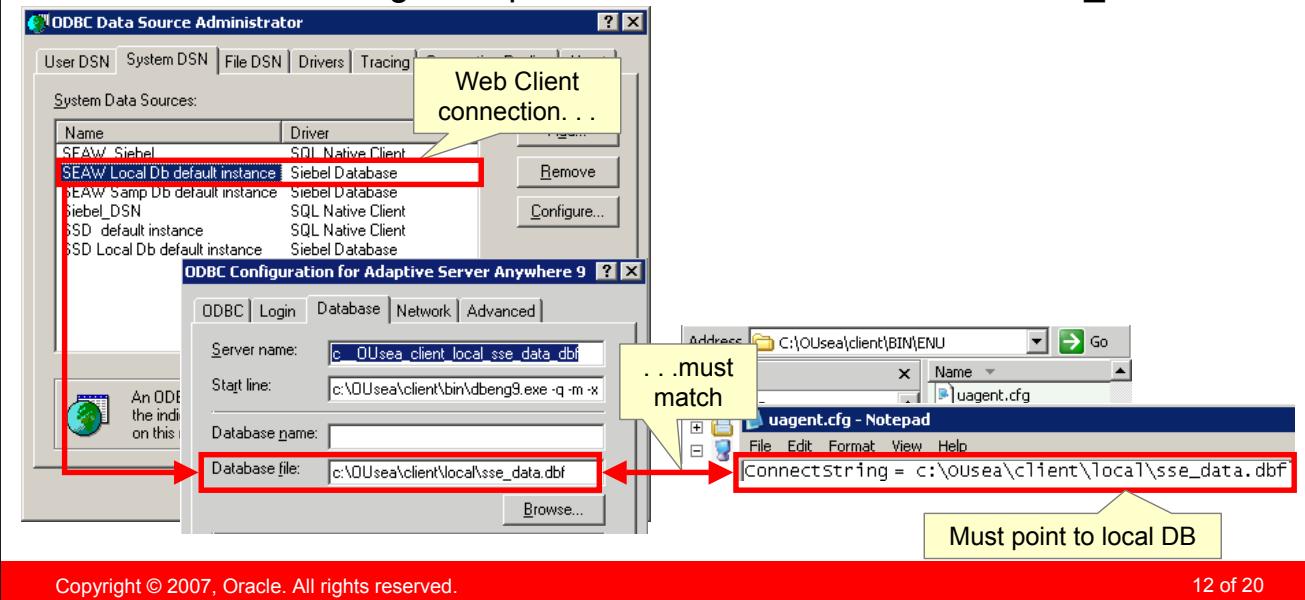
Server connection information

Copyright © 2007, Oracle. All rights reserved.

11 of 20

Verify Web Client ODBC Connections

- Ensure that the Web Client ODBC connection references the local database
 - ▶ The database file parameter of SEAW Local Db default instance must match ConnectString in [Local] section of uagent.cfg
 - ▶ ConnectString must point to local DB: \client\LOCAL\sse_data.dbf

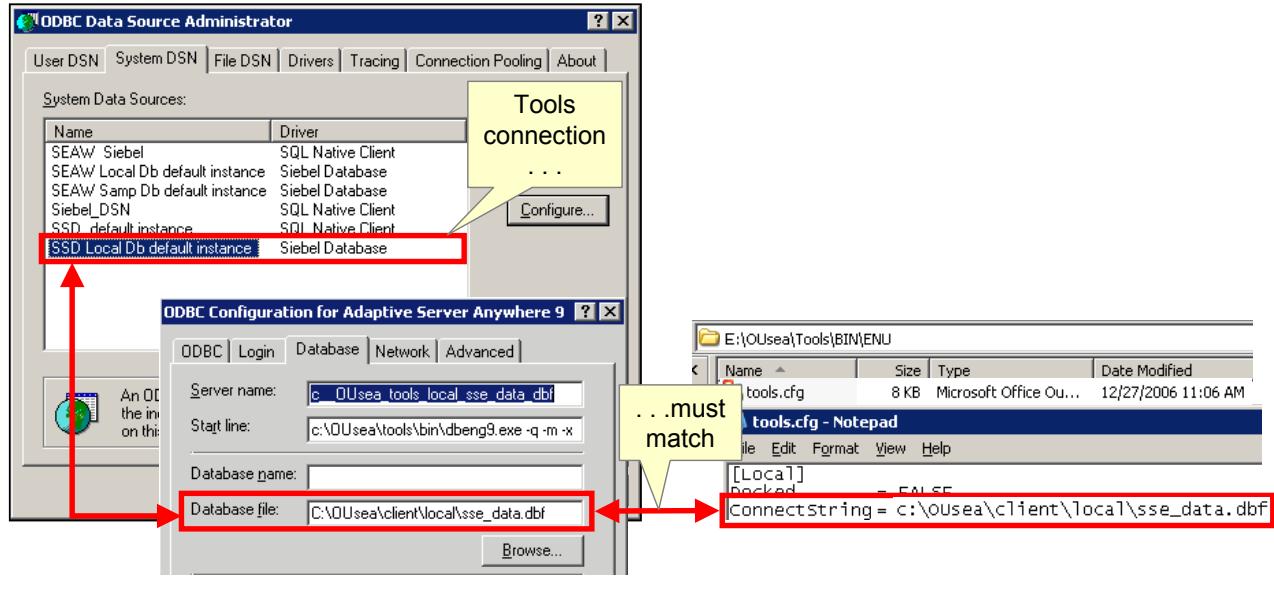


Copyright © 2007, Oracle. All rights reserved.

12 of 20

Verify Tools ODBC Connections

- Ensure that the Tools ODBC connection references the local DB
 - ▶ The database file parameter of SSD Local Db default instance must match ConnectString in [Local] section of tools.cfg

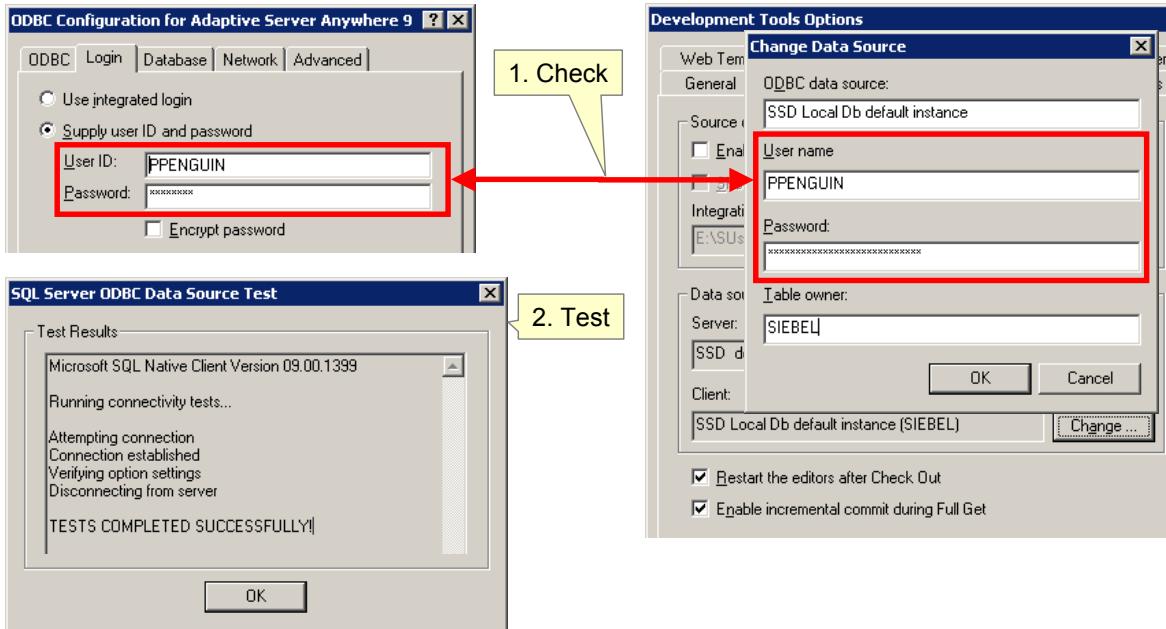


Copyright © 2007, Oracle. All rights reserved.

13 of 20

Test Tools ODBC Login

- Check user IDs and passwords, and then test the connection

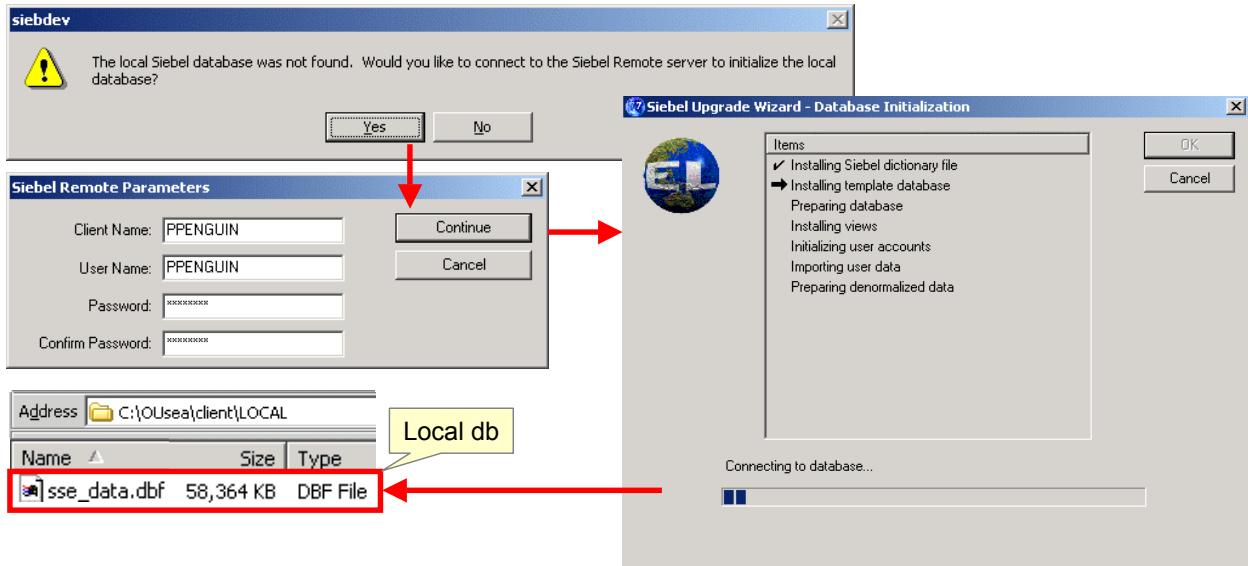


Copyright © 2007, Oracle. All rights reserved.

14 of 20

Initialize Local Database

- Local database is initialized upon first login to client
 - ▶ Creates local database (sse_data.dbf) from template
 - ▶ Populates local database with data from compressed files

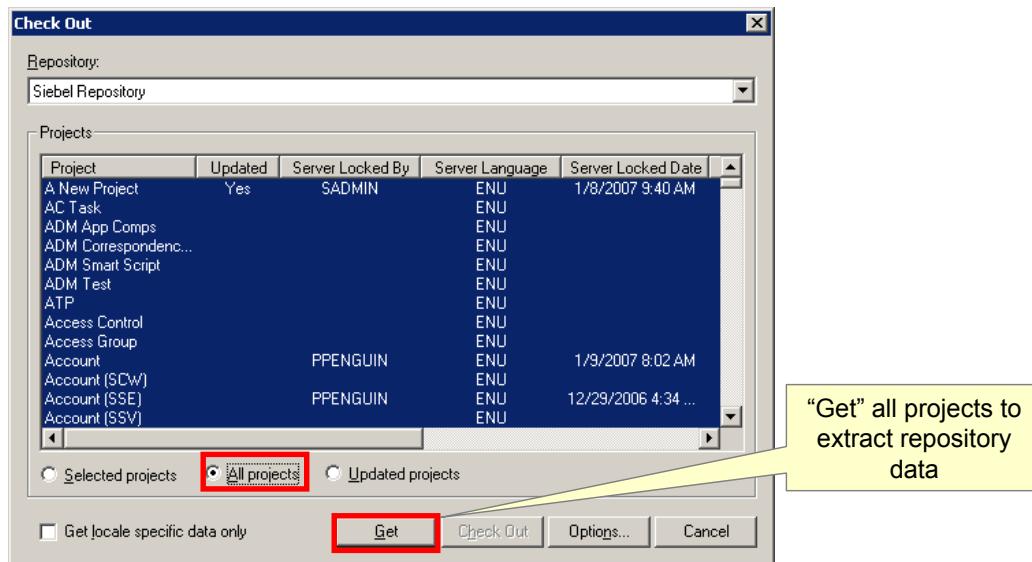


Copyright © 2007, Oracle. All rights reserved.

15 of 20

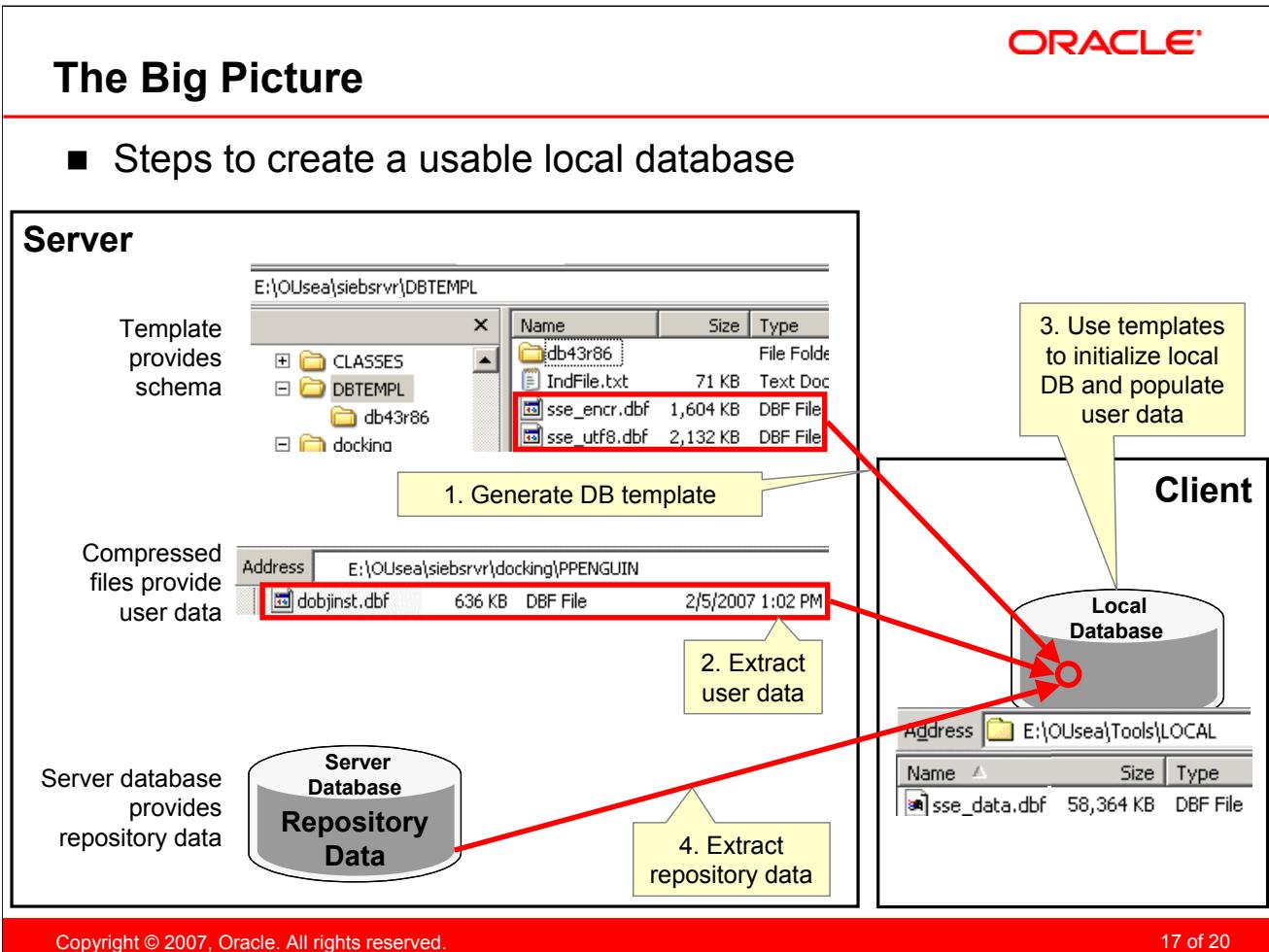
Extract Repository Data

- Perform a Get to extract repository data
 - ▶ Copies all objects from the server repository to the local repository
 - ▶ Must be performed before compiling
 - SRF must be based on the full set of Siebel objects



Copyright © 2007, Oracle. All rights reserved.

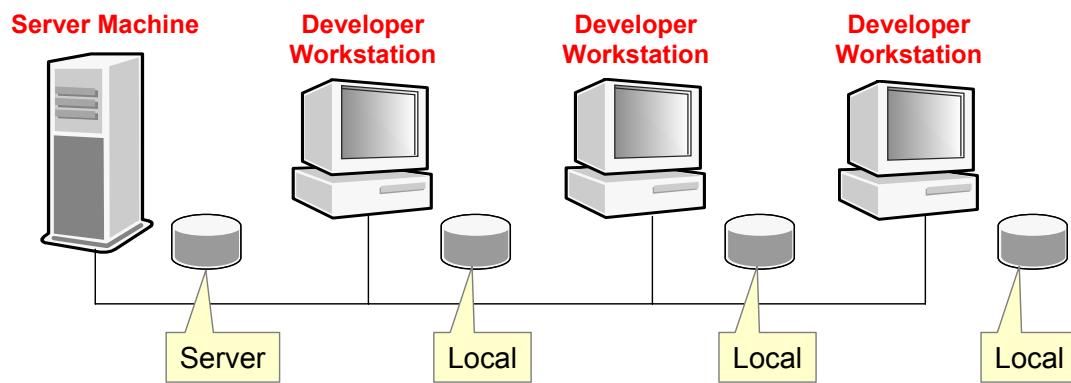
16 of 20



ORACLE®

Additional Developers

- For each additional developer:
 1. Create the developer
 2. Create a user login
 3. Extract the local database
 4. Initialize the local database
 5. Verify login and data sources



Copyright © 2007, Oracle. All rights reserved.

18 of 20



Module Highlights

- Siebel Remote is used to establish a local development environment
- An individual developer uses Siebel Developer Web Client to develop and test custom configurations
- Create the development environment by:
 - ▶ Installing Siebel Tools on the client
 - ▶ Enabling Siebel Remote on the server
 - ▶ Creating a developer, login, and remote user on the server
 - ▶ Generating, extracting, and initializing a local database
 - ▶ Verifying login and data sources



Lab

- In the lab you will:
 - ▶ Set up an employee as a developer
 - ▶ Extract a local database for the developer
 - ▶ Populate the developer's local database with user and repository data

ORACLE®

Siebel 8.0 Essentials

Module 22: Managing Object Definitions

22

Copyright © 2007, Oracle. All rights reserved.

Oracle Internal & Oracle Academy Use Only

22



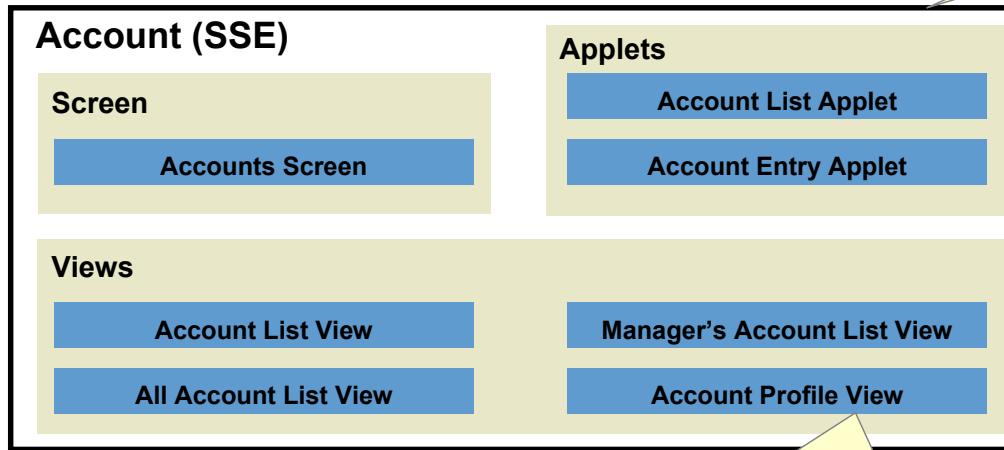
Module Objectives

- After completing this module you should be able to:
 - ▶ Explain the role of projects
 - ▶ Manage object definitions using Check Out and Check In
 - ▶ Lock projects locally
- Why you need to know:
 - ▶ Checking projects in and out are critical tasks that enable multiple developers to work together on a configuration effort

ORACLE®

Projects

- Are named sets of object definitions in a repository
 - ▶ Only one version of a project exists in a repository at a time
- Are a mechanism to organize object definitions so that a single developer can exclusively work on them as a group

Project**Object definition with Project property = Account (SSE)**

Copyright © 2007, Oracle. All rights reserved.

3 of 22

Reference

Using Siebel Tools

ORACLE®

Existing and New Projects

- The as-delivered Siebel application contains a large number of existing projects
- New projects are created by:
 - ▶ Selecting the Project object type, and
 - ▶ Creating a new record in the Object List Editor

The screenshot shows the Siebel Object Explorer and the Project List window side-by-side.

Object Explorer: Shows a tree view of Siebel Objects under the project "xx All Projects xx". The nodes include Applet, Application, Business Component, Business Object, Entity Relationship Diagram, Link, and Project.

Project List: Shows a grid of projects. The columns are Name, Changed, Inactive, Locked, and Locked By Name. The data is as follows:

Name	Changed	Inactive	Locked	Locked By Name
A New Project	✓			
ABC New Applets	✓		✓	PPENGUIN
ABC New Bus Comps	✓		✓	PPENGUIN
ABC New Views	✓		✓	PPENGUIN
ABC Strings	✓		✓	PPENGUIN
AC Task				

Copyright © 2007, Oracle. All rights reserved.

4 of 22

ORACLE®

Assigning Object Definitions to Projects

- Every object definition must belong to one, and only one, project
 - ▶ Either a Siebel-supplied project, or
 - ▶ A user-created project
- A top-level object definition has a Project property
 - ▶ Child object definitions belong to the parent's project

The screenshot shows the Siebel Control List interface. On the left is the Object Explorer with a tree view of Siebel Objects under a selected 'Applet' node. The main window displays two tables: 'Applets' and 'Controls'. The 'Applets' table has columns for Name, Project, and Business Component. The 'Controls' table has columns for Name, Changed, and Caption. A callout box points from the 'Controls' table to a note about child objects.

Name	Project	Business Component
Account Entry Applet	Account (SSE)	Account
Account Entry Applet - Child	Account (SSE)	Account
Account Entry Applet - Child - Admin	Account (SSE)	Account
Account Entry Applet - Toggle	Account (SSE)	Account

Name	Changed	Caption
Account Name Title		
AccountStatus		

Child belongs to parent's project. It does not have a project property.

Copyright © 2007, Oracle. All rights reserved.

5 of 22

ORACLE®

Viewing Object Definitions by Project

- View object definitions in a project using the Project drop-down list
 - ▶ Displays object types for which there are object definitions in the selected project
 - ▶ Select **All Projects** at the top of list to see all projects

The screenshot shows two windows: 'Object Explorer' on the left and 'Applets' on the right.

Object Explorer: The 'Project' dropdown is set to 'Account (SSE)'. A yellow callout box labeled '1. Select project' points to this dropdown. Below it, the 'Types' tab is selected in the ribbon. The tree view under 'Siebel Objects' shows 'Applet' is selected, indicated by a yellow callout box labeled '2. Select object type'.

Applets: This window lists various applets. A red box highlights the 'Project' column. The data is as follows:

Name	Project
Account Activity List Applet	Account (SSE)
Account Address Mvg Applet	Account (SSE)
Account Attachment Applet	Account (SSE)
Account Bill To Applet	Account (SSE)
Account Bill To/Ship To Entry Applet	Account (SSE)
Account Contact List Applet	Account (SSE)
Account Contact List Applet - Grandchild	Account (SSE)
Account DNB Tree Applet	Account (SSE)
Account Default Chart Applet	Account (SSE)
Account Entry Applet	Account (SSE)
Account Entry Applet - Child	Account (SSE)

A yellow callout box labeled '3. Object Type with project property = Account (SSE) displayed' points to the highlighted 'Project' column.

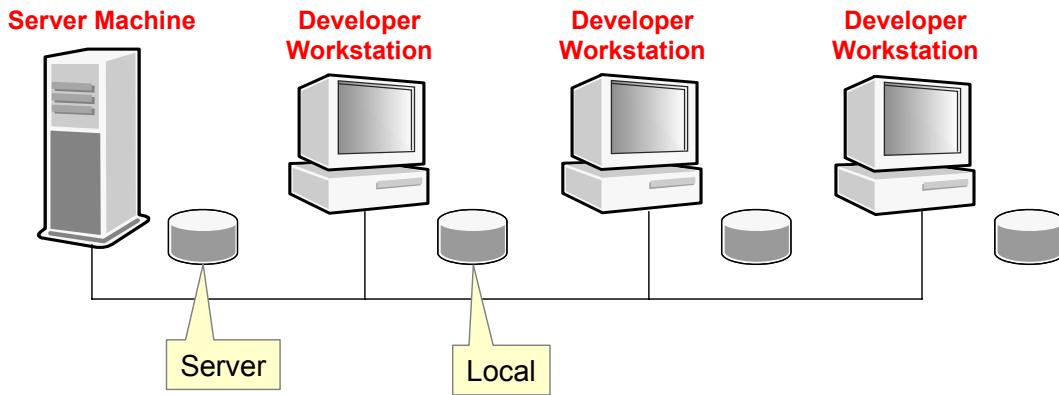
Copyright © 2007, Oracle. All rights reserved.

6 of 22

ORACLE®

Repositories

- Master copy of repository resides on server database
 - ▶ Stores tested object definitions for entire development team
- Each developer has repository copy on local developer (client) workstation
 - ▶ Developers perform all editing on their local repository



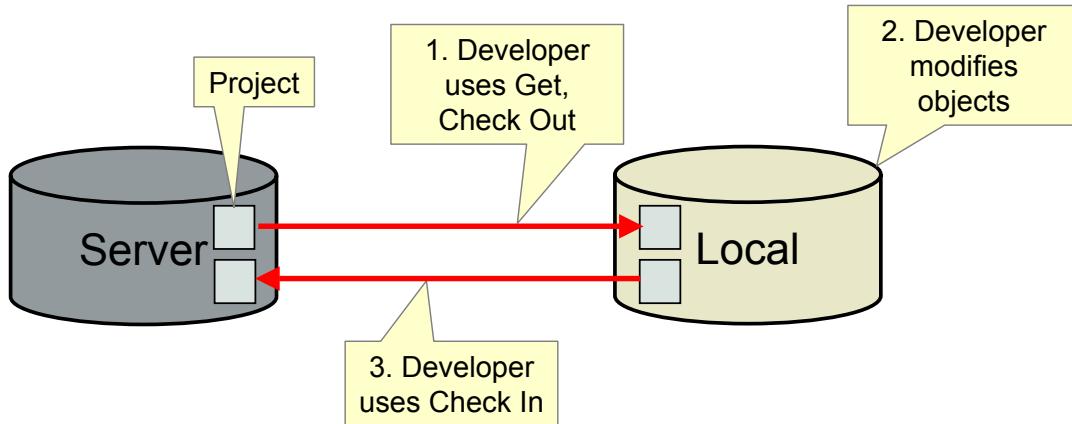
Copyright © 2007, Oracle. All rights reserved.

7 of 22

ORACLE®

Copying Projects

- Check In and Check Out safely manage concurrent use by multiple developers
- Object definitions in one or more projects can be copied:
 - ▶ From the server to the local developer repository (Use Get or Check Out to retrieve definitions)
 - ▶ From the local developer repository to the server (Check In)
- Overwrites the version in the destination repository



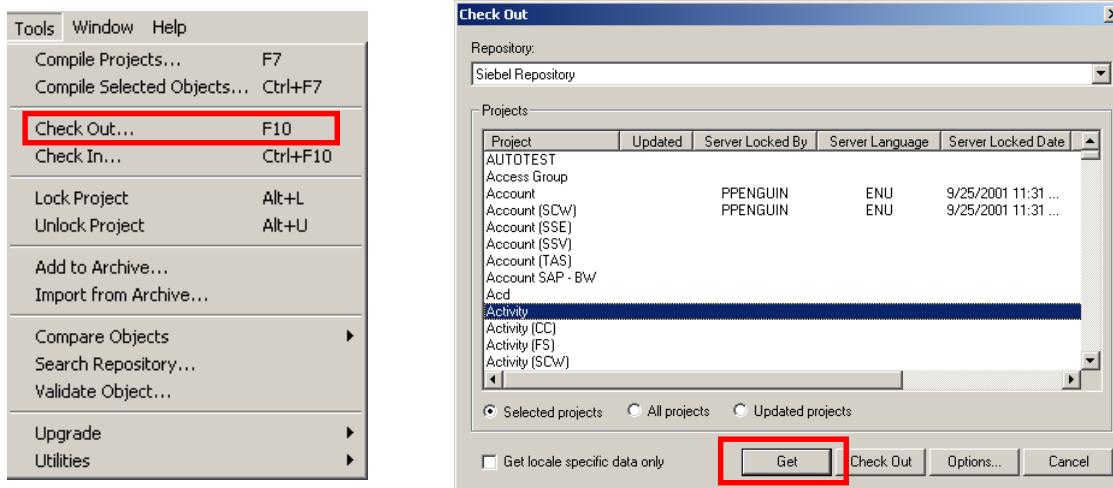
Copyright © 2007, Oracle. All rights reserved.

8 of 22

ORACLE®

Getting Projects

- Get populates a local database with a copy of projects from the server repository
- Use Get to bring object definitions modified by other developers to your local database
- Use Get with All Projects selected to populate a newly-initialized local database



Copyright © 2007, Oracle. All rights reserved.

9 of 22

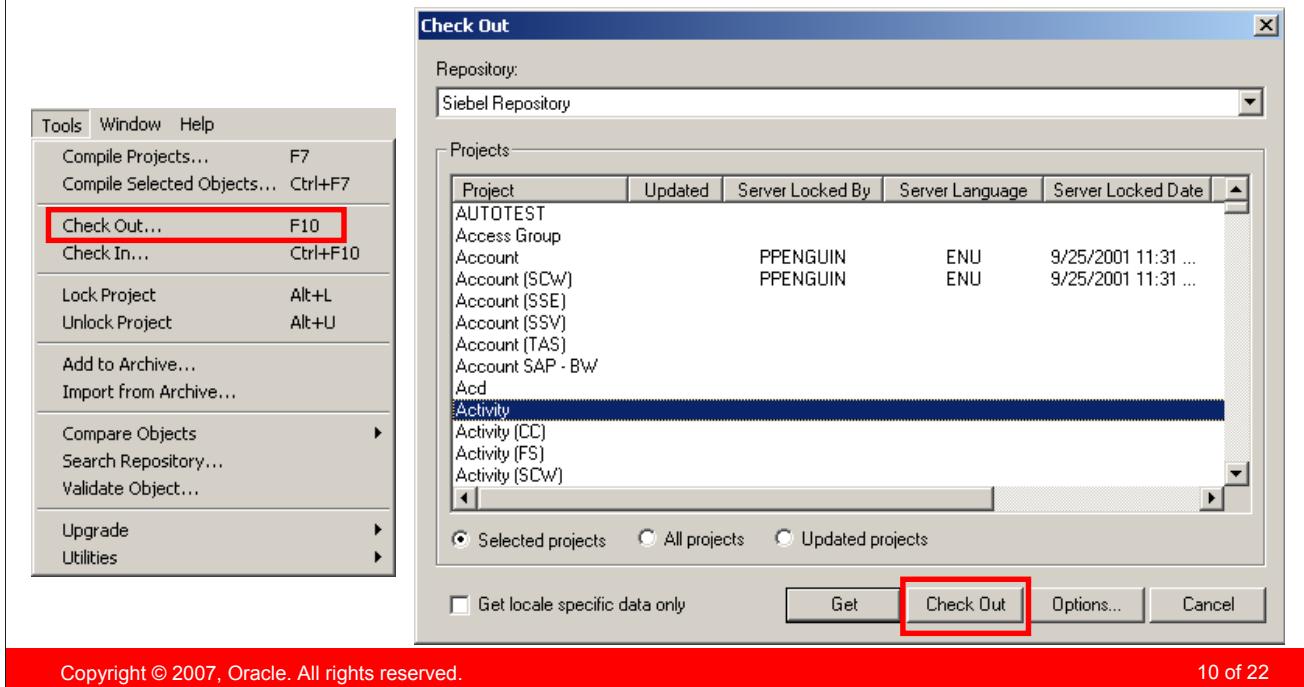
Note

First select Check Out from the Tools menu, and then click Get in the Check Out dialog box.



Check Out

- Click Check Out to copy projects from the server repository to the local repository
 - ▶ The project's Allow Object Locking property must be disabled



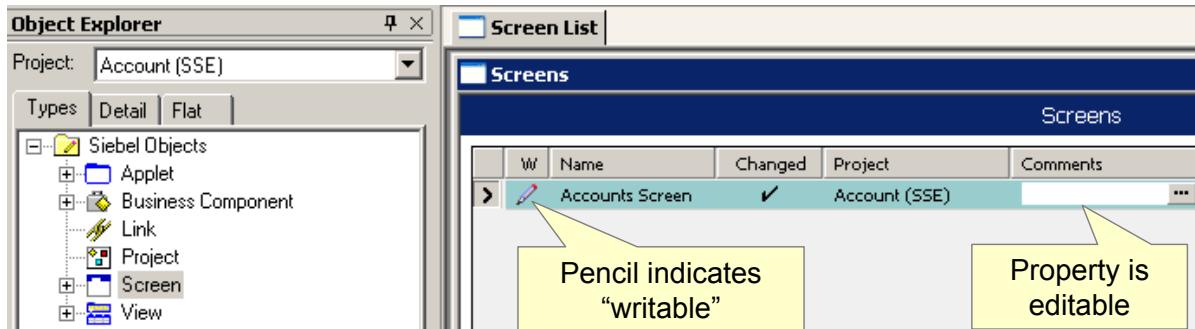
Copyright © 2007, Oracle. All rights reserved.

10 of 22

ORACLE®

Check Out Continued

- Use Check Out in order to edit object definitions in a project
 - ▶ Copies all object definitions in project on server to local database
 - Local copy of project is overwritten by server version
 - ▶ Locks project on server repository
 - Prevents other developers from modifying that project
 - ▶ Locks the project in the local repository
 - Permits developer to make changes to object definitions in that project



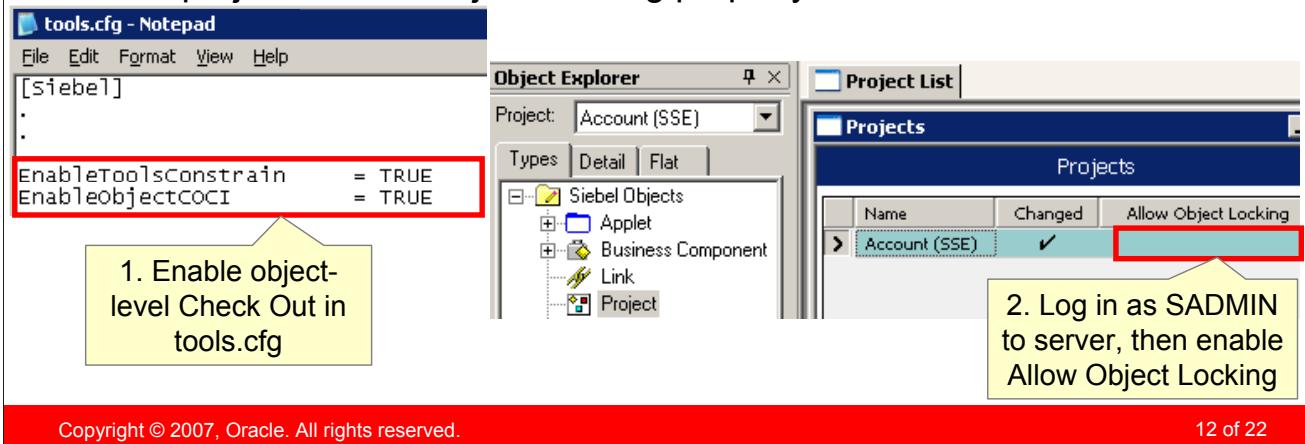
Copyright © 2007, Oracle. All rights reserved.

11 of 22

ORACLE®

Check Out Objects

- Allows for checking out individual objects
 - ▶ Quicker than checking out entire projects
- To enable object-level Check Out
 - ▶ Enable object-level Check Out in the [Siebel] section of tools.cfg
 - Ensure ToolsConstrain parameters = TRUE
 - Add EnableObjectCOCI=TRUE
 - ▶ Log in to Tools as SADMIN to the server database and enable the project's Allow Object Locking property



Copyright © 2007, Oracle. All rights reserved.

12 of 22



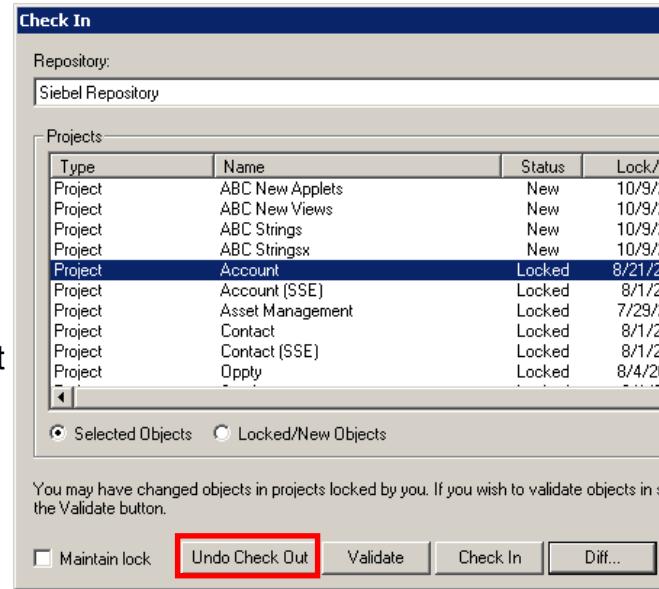
Canceling Check Out

- Discards changes made to checked-out project if:
 - ▶ Changes are no longer required
 - ▶ Local project has become corrupted
- To cancel a Check Out of a project:
 - ▶ Check out the project from the server again
 - Replaces modified object definitions in local repository with original version from server repository
 - Repeating Check Out is permitted for person who last checked out project
 - ▶ Then check in the project to unlock the project on both the local and server repositories



Undo Check Out

- Use Undo Check Out to release lock on a project on the server without checking in edited object definitions
 - ▶ Makes the original project available to another developer
 - ▶ Can be followed by Get to restore original definitions in local DB
 - Alternate way to cancel Check Out
- Undo Check Out:
 - ▶ Releases lock on server
 - ▶ Retains lock on local database
 - Developer can continue to modify object definitions but will not be able to check them in to the server



Copyright © 2007, Oracle. All rights reserved.

14 of 22



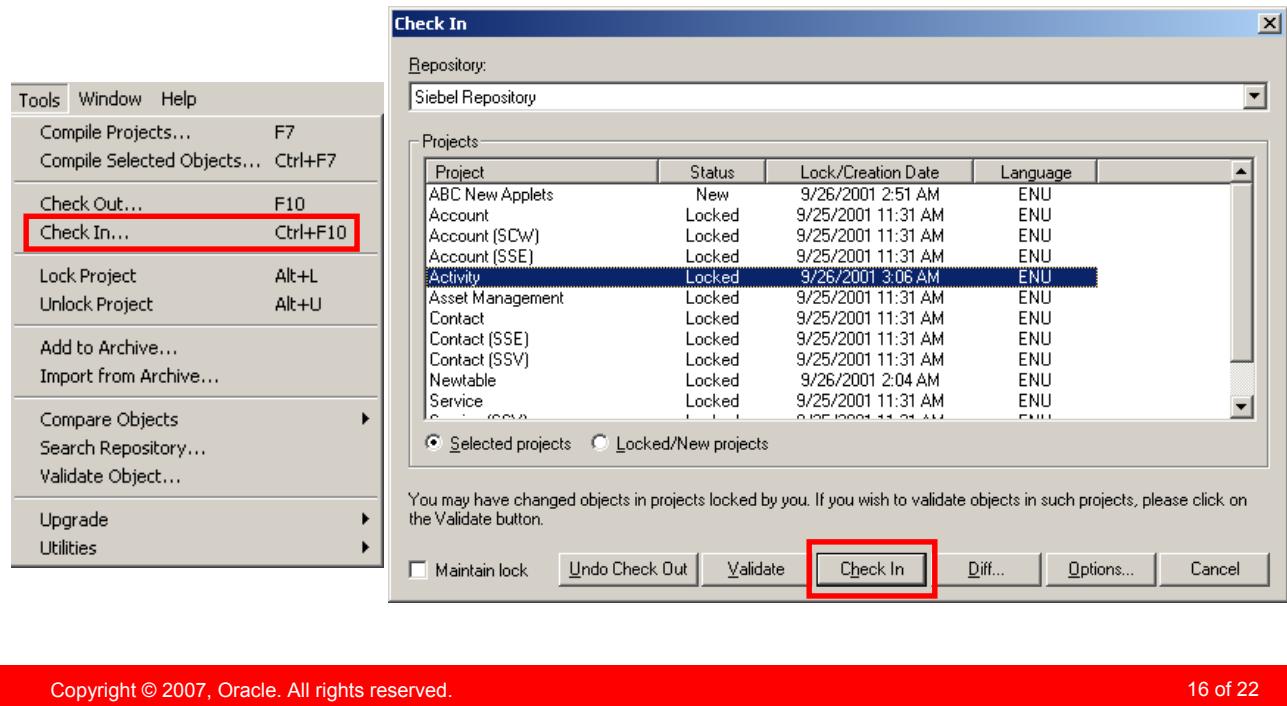
Check In

- Is used when object definitions in the project have been edited and tested
 - ▶ Copies object definitions from local repository to server repository
 - ▶ Replaces versions of checked out object definitions with new versions and unlocks projects
 - Can also check in object definitions in newly created projects
 - ▶ Releases locks on both server and local copies of project
 - Maintain Lock leaves project locked for further local modification by current developer

ORACLE®

Check In continued

- Click Check In to copy projects and objects from the local repository to the server repository



Copyright © 2007, Oracle. All rights reserved.

16 of 22



Locking Projects

- You can directly lock and unlock projects in the current repository (local or server)

The screenshot shows the Siebel Object Explorer interface. On the left is the Object Explorer with a tree view of Siebel Objects, Applet, Application, Business Component, Business Object, Entity Relationship Diagram, and Link. On the right is the Project List window titled 'Projects'. It displays a table with columns: Name, Changed, Inactive, Locked, Locked By Name, and Locked Date. A row for 'A New Project' is selected, showing checked marks in the Changed and Locked columns, and 'SADMIN' in the Locked By Name column. A yellow callout box points to the 'Locked' column header with the text 'Click to set Locked property directly...'. Below the main windows, a context menu is open with options: Compile Projects... (F7), Compile Selected Objects... (Ctrl+F7), Check Out... (F10), Check In... (Ctrl+F10), Lock Project (Alt+L, highlighted with a red box), and Unlock Project (Alt+U). A yellow callout box points to the 'Lock Project' option.

Copyright © 2007, Oracle. All rights reserved.

17 of 22

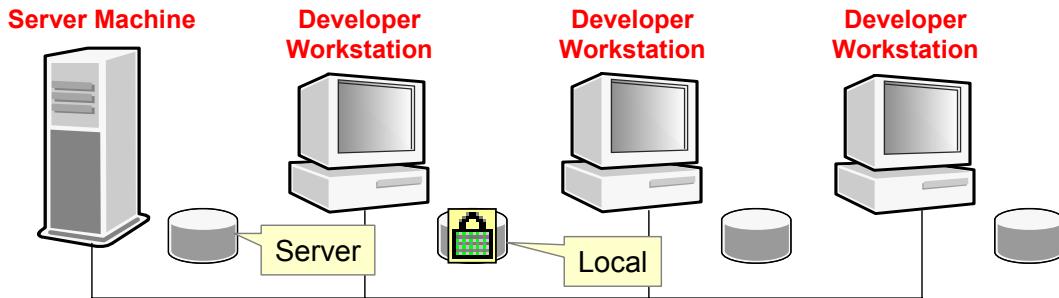
Project Drop-Down

You can select **My Locked Projects** in the Project drop-down list to display only those projects you have locked (either by checking them out or locking them directly).



Locking Projects Locally

- Allows developer to make and test modifications locally
- Prevents developer from checking changes into server repository
 - ▶ However, locking does not prevent other developers from checking out and modifying a project
- Consider when:
 - ▶ Prototyping your ideas without preventing other developers from checking out the project
 - ▶ Intending to discard (rather than save) your work



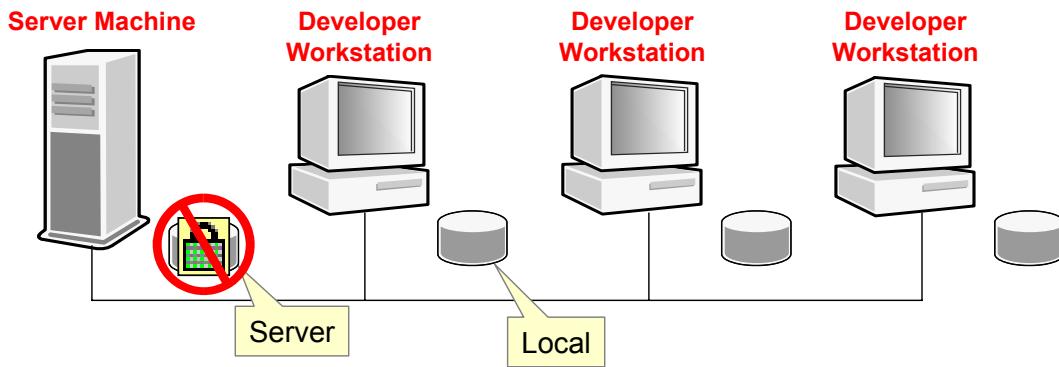
Copyright © 2007, Oracle. All rights reserved.

18 of 22

ORACLE®

Do Not Lock Projects on the Server

- Oracle recommends that developers never directly lock projects on the server
 - ▶ Prevents the developer from undoing changes and restoring original definitions
 - ▶ Allows other developers to get object definitions in an incomplete, inconsistent, and untested state
- Always use Check Out and modify the object definitions locally



Copyright © 2007, Oracle. All rights reserved.

19 of 22



Module Highlights

- Projects organize object definitions so a developer can work on them exclusively
- The as-delivered application contains many projects
- Every object definition must belong to one, and only one, project
- Developer edits a copy of master repository residing on local client
- Check Out and Check In are used to modify object definitions
 - ▶ Project locking ensures only one developer can modify a given object



Lab

- In the lab you will:
 - ▶ Explore how projects work in Siebel Tools
 - ▶ Check out a project from the server

Copyright © 2007, Oracle. All rights reserved.

21 of 22

ORACLE®

Siebel 8.0 Essentials

Module 23: Editing and Compiling Object Definitions

23

Copyright © 2007, Oracle. All rights reserved.

23



Module Objectives

- After completing this module you should be able to:
 - ▶ Create new and edit existing object definitions
 - ▶ Validate edited object definitions
 - ▶ Archive object definitions and projects
 - ▶ Back up the local database
 - ▶ Compile object definitions into a repository file
- Why you need to know:
 - ▶ These are fundamental activities that occur during all configuration efforts

ORACLE®

Editing Object Definitions

- Use the Object Explorer and Object List Editor to edit object definitions
 - ▶ Ensure that the project has been locked

The screenshot shows the Siebel Object Explorer and Applet List windows. The Object Explorer window on the left displays a tree view of Siebel Objects under the Project 'Account (SSE)'. The Applet List window on the right shows a list of applets with columns for Name, Project, Changed, Comments, and Type. A context menu is open over the 'Account List Applet' row, with options like 'Type in a value, or...', '...select a value', and a dropdown menu for 'Order Type'.

Pencil indicates object is locked and editable

W	Name	Project	Changed	Comments	Type
	Account Contact List Applet	Account (SSE)			Standard
	Account Contact List Applet - Grandchild	Account (SSE)			Standard
	Account DNB Tree Applet	Account (SSE)			Standard
	Account Default Chart Applet	Account (SSE)			Standard
	Account Entry Applet	Account (SSE)			Standard
	Account Entry Applet - Child	Account (SSE)			Standard
	Account Entry Applet - Child - Admin	Account (SSE)			Standard
	Account Entry Applet - Toggle	Account (SSE)			Standard
	Account Form Applet	Account (SSE)			Standard
	Account Form Applet - Short	Account (SSE)			Standard
	Account Form ReadOnly Applet	Account (SSE)			Standard
	Account Issue Action Applet	Account (SSE)			Standard
	Account Issue List Applet	Account (SSE)			Standard
	Account List Applet	Account (SSE)			Standard
	Account List Without Navigation Applet	Account (SSE)			Standard
	Account Mvg Applet	Account (SSE)			Standard
	Account Note Applet	Account (SSE)			Standard
	Account Order List Applet (Sales)	Account (SSE)	[Order Type] = LookupValue	Create for Search Preview	Standard
	Account Preview Applet	Account (SSE)			Standard
	Account Private Note Applet	Account (SSE)			Standard

Copyright © 2007, Oracle. All rights reserved.

3 of 24

Reference

Using Siebel Tools

ORACLE®

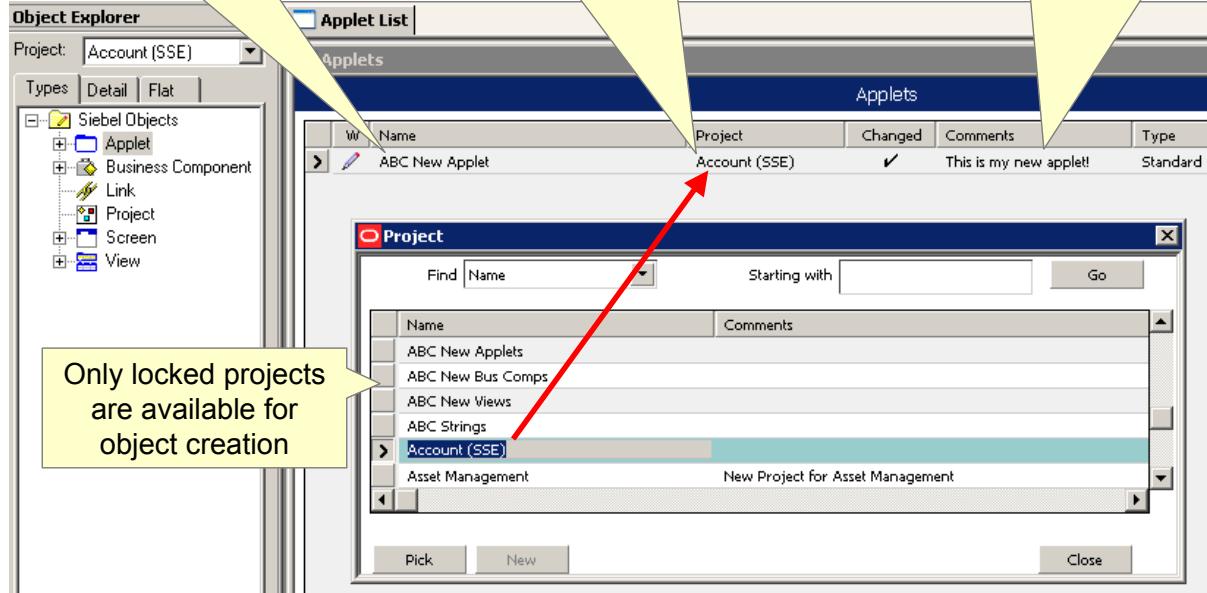
Creating a New Object Definition

- Select the desired object type in the Object Explorer
- Create a new record in the Object List Editor

1. Enter a unique name

2. Select a project from the picklist

3. Assign values to the remaining properties, as necessary



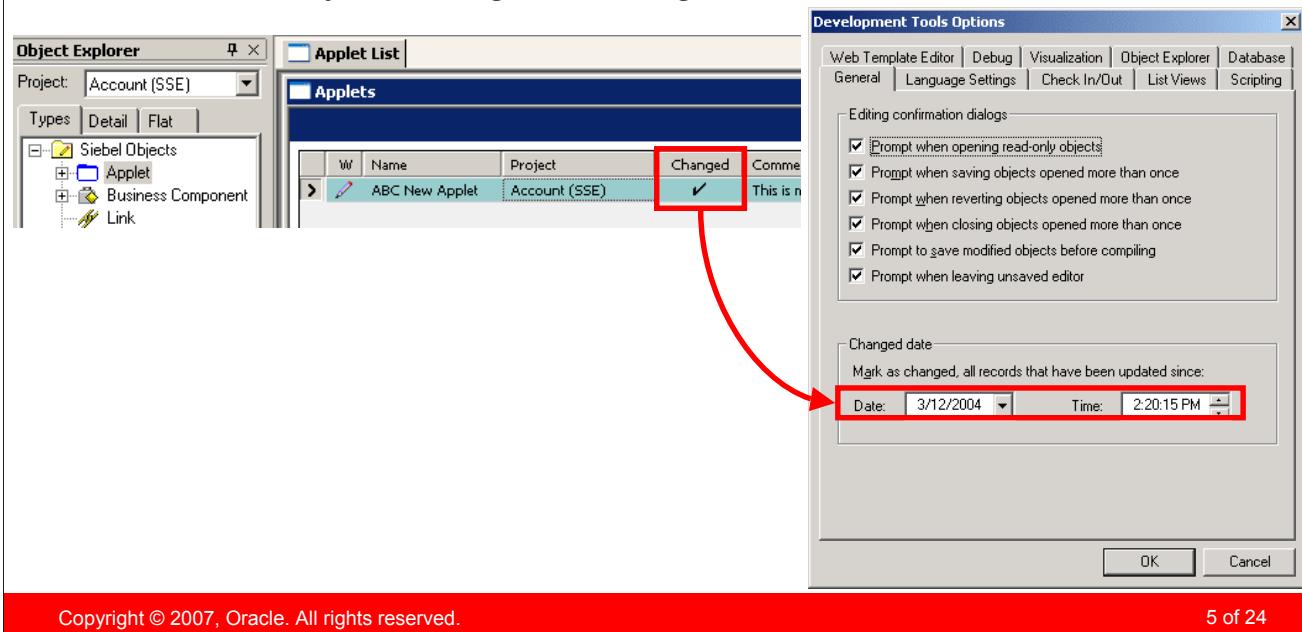
Copyright © 2007, Oracle. All rights reserved.

4 of 24

ORACLE®

Changed Flag

- Indicates records that have been modified since the changed date
- Is set whenever a record is edited
- Is cleared by resetting the changed date value



Copyright © 2007, Oracle. All rights reserved.

5 of 24

Changed Flag

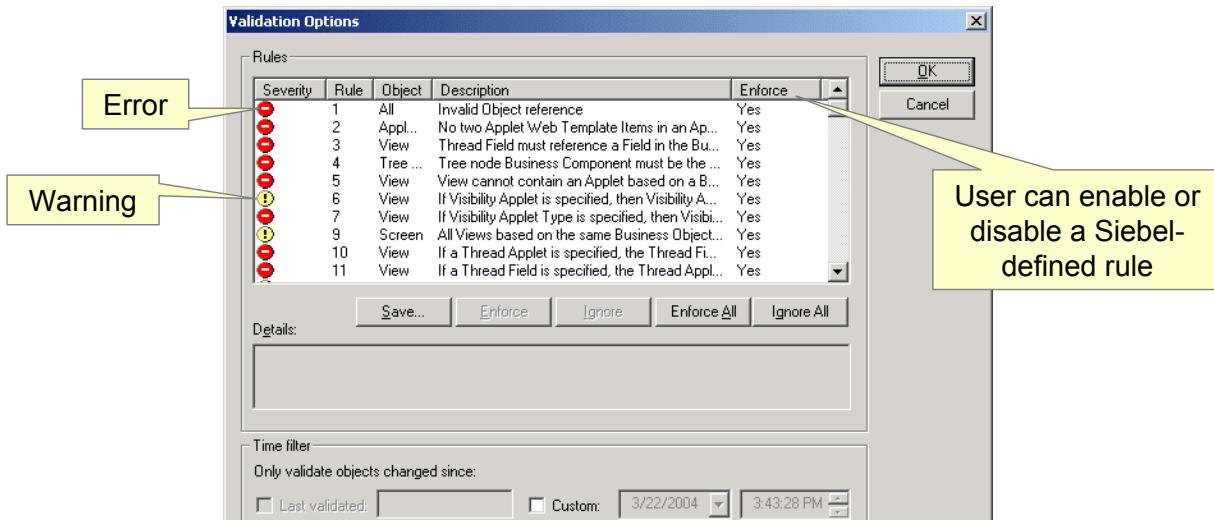
The Changed flag indicates only that the record has been touched. A developer may change the value of a property and then undo the change. The Changed flag will still be set.

Some developers will select an object type and then query for all object definitions with the Changed flag set. Only those definitions that have been edited will display in the OBLE.

ORACLE®

Validating Object Definitions

- Determines the correctness of selected object definitions in the repository
 - ▶ Includes all child object definitions for the selected parent definitions
- Is based on a set of Siebel-defined rules



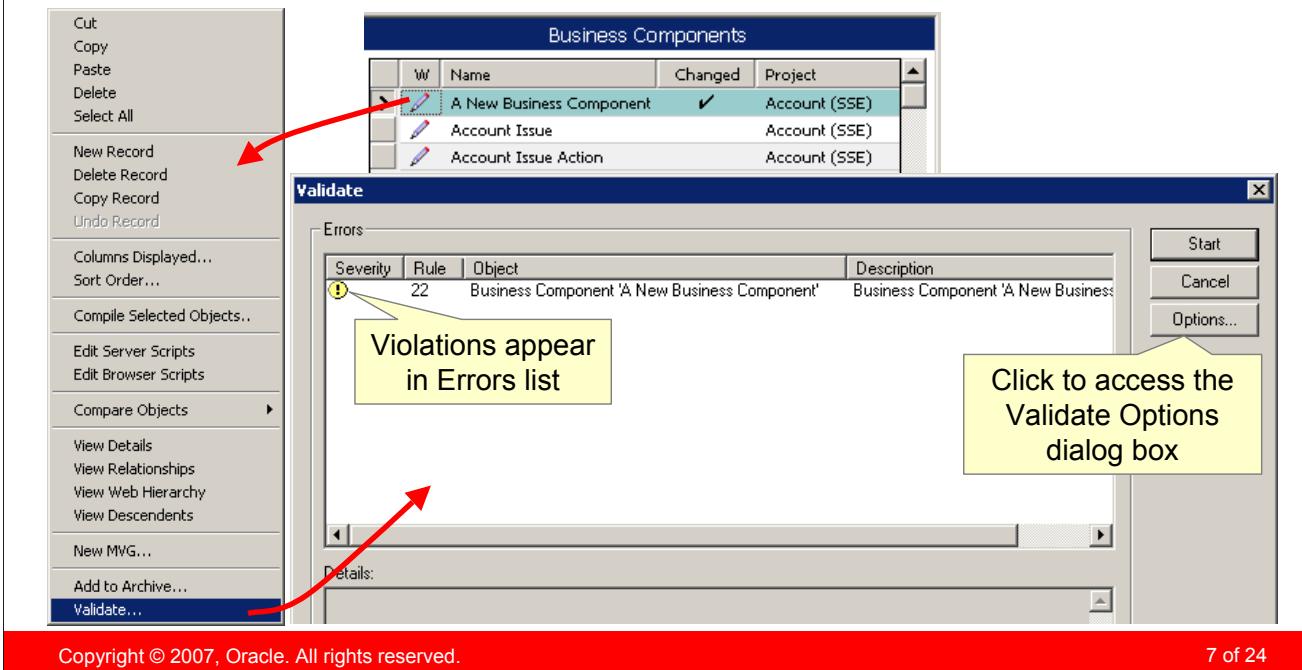
Copyright © 2007, Oracle. All rights reserved.

6 of 24

ORACLE®

Validating Object Definitions Continued

- Select the object definition to validate
- Right-click and select Validate
- Click Start in the Validate dialog box



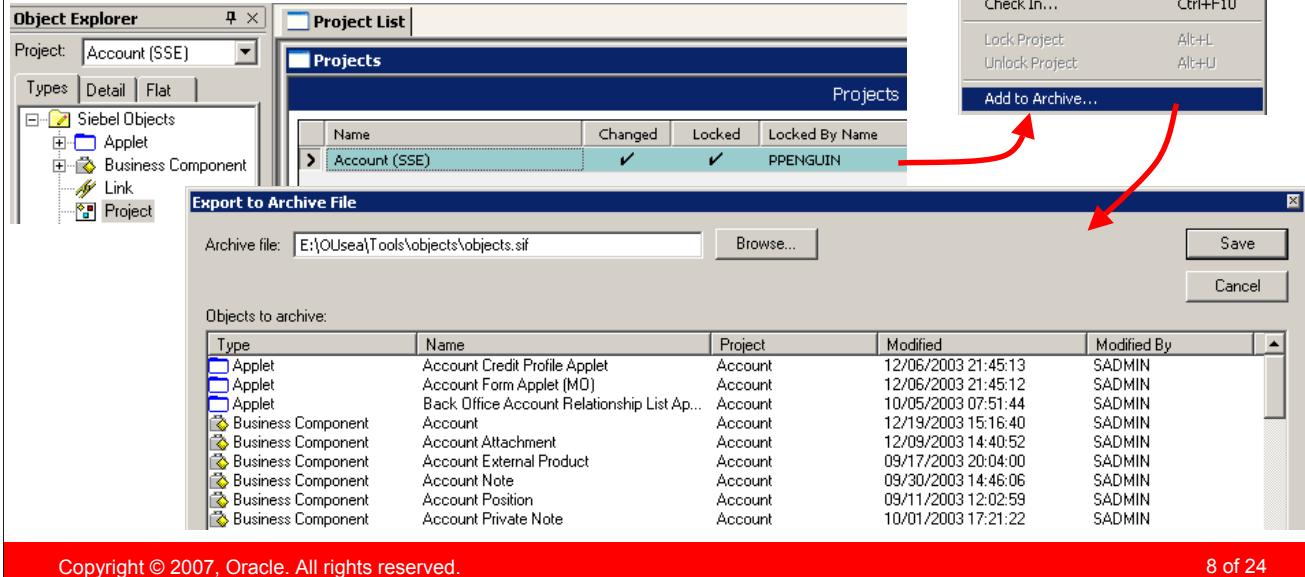
Copyright © 2007, Oracle. All rights reserved.

7 of 24

Archiving Projects

ORACLE®

- Projects can be archived to .sif files
 - ▶ Select the project (or projects) to be archived
 - ▶ Select Tools > Add to Archive
 - ▶ Select an existing (or create a new) archive file
 - ▶ Click Save in the Export to Archive File dialog box



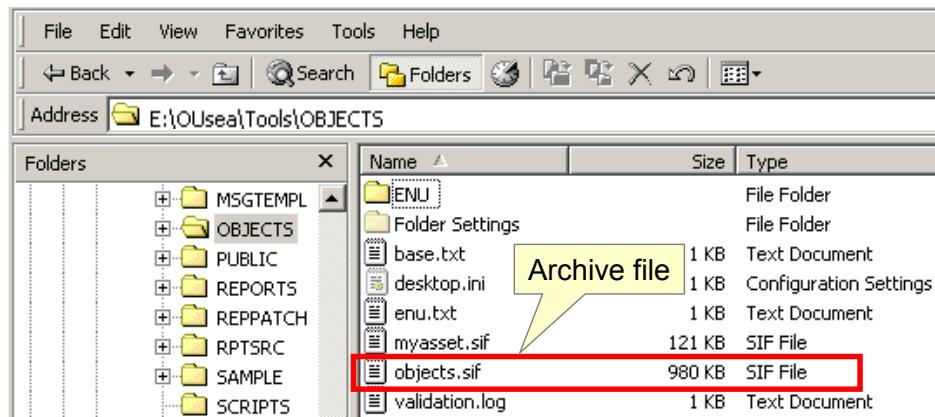
Copyright © 2007, Oracle. All rights reserved.

8 of 24

ORACLE®

Archive Files

- Contain projects exported from a repository
 - ▶ Available for importing into other repositories
- Have .sif extension by default
- Are used to share projects among developers
- Are used to back up projects
- Can be version-controlled using source code control systems



Copyright © 2007, Oracle. All rights reserved.

9 of 24



Archive Recommendations

- Archive the entire project
- Do not archive individual objects
 - ▶ Could result with inconsistent object definitions
 - ▶ Difficult to recover from a partial or incomplete archive
 - ▶ Will only save a few minutes per day compared to archiving the project
- Archive at appropriate intervals
 - ▶ Prior to implementing a large amount of work
 - ▶ After testing a section of work
 - ▶ Several times per day during heavy development

ORACLE®

Importing Object Definitions

- Object definitions in an archive file can be imported into the local repository on the developer workstation
 - ▶ Appear in the current active repository
- To import object definitions
 - ▶ Select Tools > Import from Archive
 - ▶ Select the archive file
 - ▶ Indicate how to resolve conflicting object definitions

Type	Name	Project	Modified	Modified By
Applet	ABC New Applet	Account	03/22/2004 07:03:00	SADMIN
Applet	Account Credit Profile Applet	Account	12/06/2003 21:45:13	SADMIN
Applet	Account Form Applet (MO)	Account	12/06/2003 21:45:12	SADMIN
Applet	Back Office Account Relationship List A...	Account	10/05/2003 07:51:44	SADMIN
Business Component	A New Business Component	Account	03/22/2004 07:47:24	SADMIN
Business Component	Account	Account	12/19/2003 15:16:40	SADMIN
Business Component	Account Attachment	Account	12/09/2003 14:40:52	SADMIN
Business Component	Account External Product	Account	09/17/2003 20:04:00	SADMIN

Copyright © 2007, Oracle. All rights reserved.

11 of 24



Resolve Import Conflicts

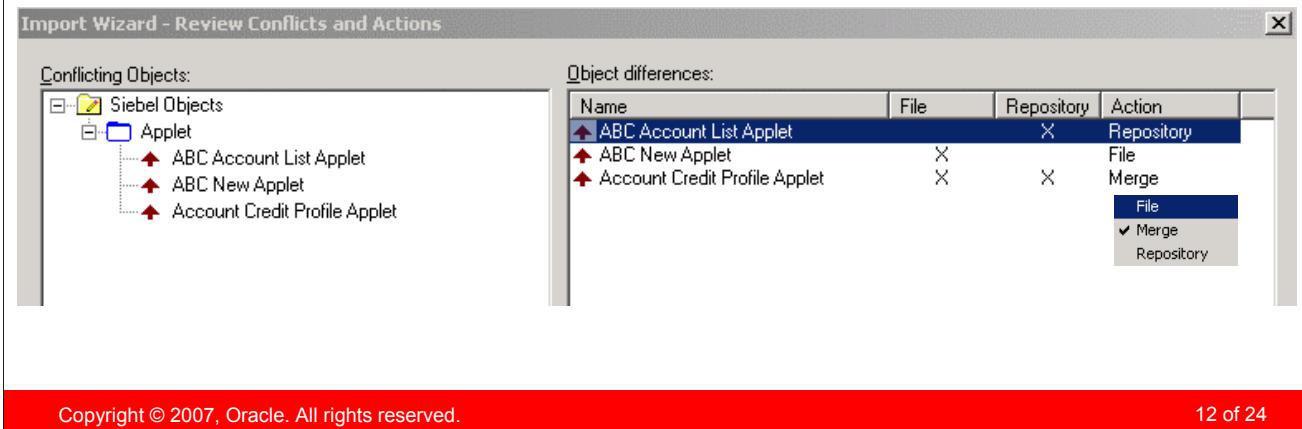
- Set the default conflict resolution

Conflict resolution

How would you like to resolve conflicts when an object definition exists in both the archive file and the repository?

Overwrite the object definition in the repository
 Merge the object definition from the archive file with the definition in the repository
 Do not import the object definition from the archive file

- Adjust individual resolutions by right-clicking them in the “Object differences” list



Copyright © 2007, Oracle. All rights reserved.

12 of 24



Compare Objects

- Displays differences between the selected object definition and another one:
 - ▶ In the current local repository
 - ▶ In an archive file
 - ▶ In the repository on the server
- Allows developers to identify differences while comparing objects
 - ▶ Delete individual child object definitions
 - ▶ Copy individual child object definitions in either direction

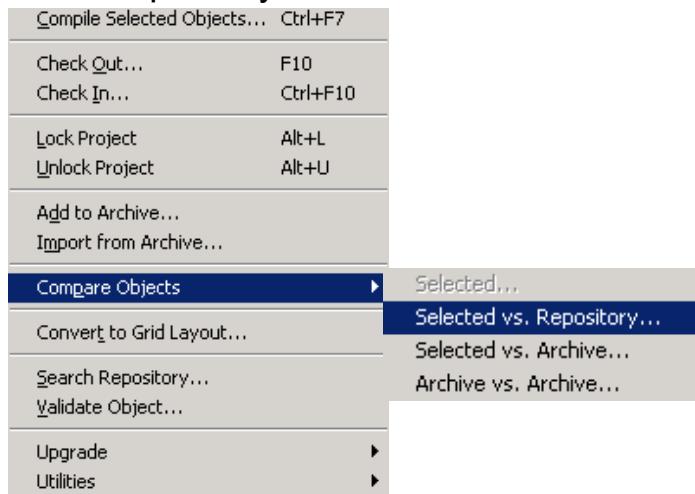
Copyright © 2007, Oracle. All rights reserved.

13 of 24



Compare Objects Continued

- To compare an object definition to one in an archive or repository:
 - ▶ Select the object definition to be compared
 - ▶ Select Tools > Compare Objects
 - ▶ Select type of comparison
 - ▶ Select the archive or repository



Copyright © 2007, Oracle. All rights reserved.

14 of 24

Note

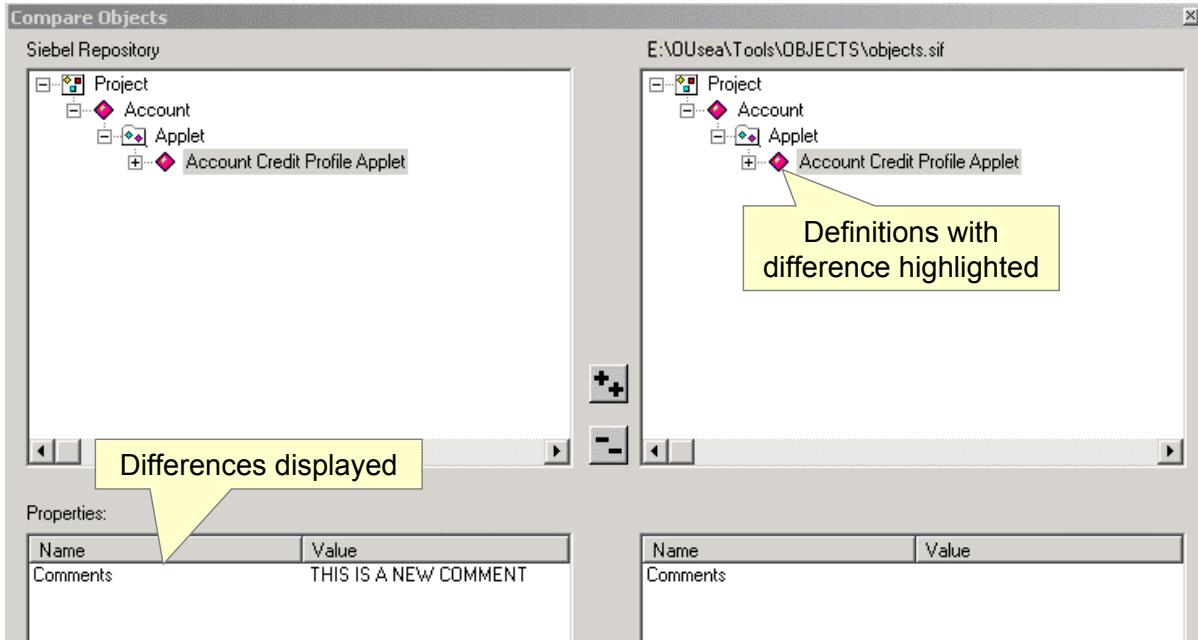
Typically there will only be one repository available, and this repository defines an application. This repository is the Siebel repository and it will be available on a local database and on a server database. However, there are circumstances when multiple repositories will be available: when a group is upgrading to a new version of Siebel Applications, or working with multiple applications.

There can not be multiple repositories in a production environment because some run-time components read repository objects directly from repository tables, not the SRF. There can be multiple SRF files, compiled from different repositories, provided that they do not compromise data integrity. Refer to *Bookshelf* for more information.

ORACLE®

Compare Objects Continued

- Examine the differences in the Compare Objects window



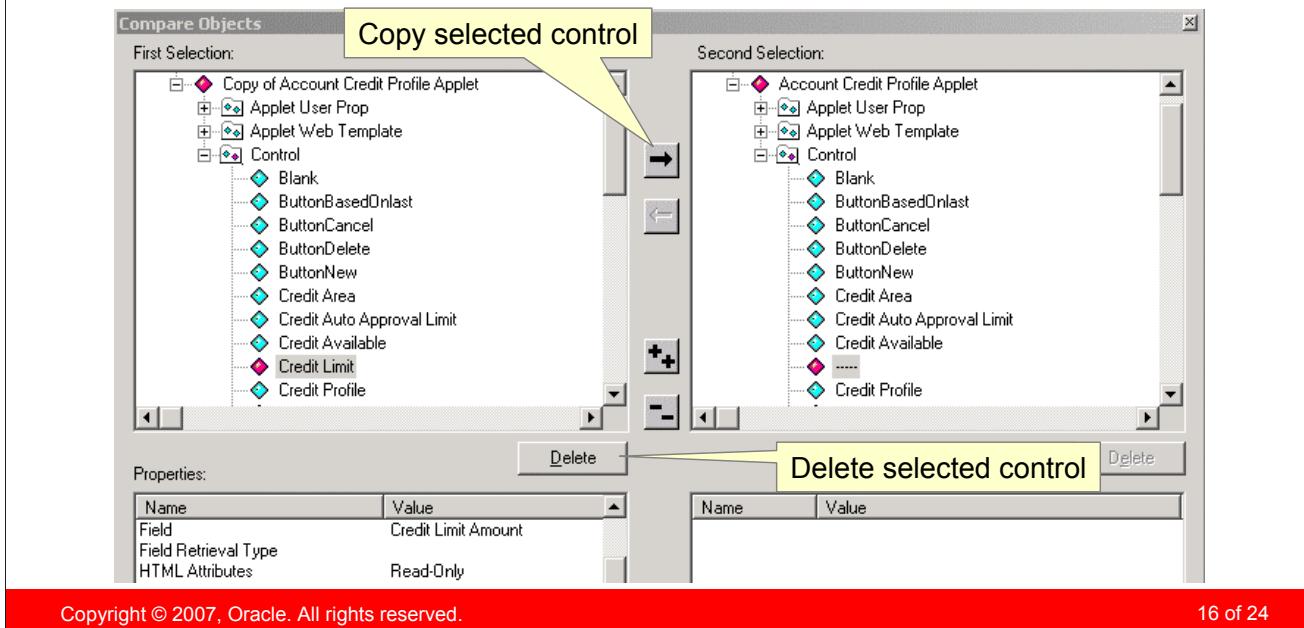
Copyright © 2007, Oracle. All rights reserved.

15 of 24

ORACLE®

Compare Objects Continued

- To compare two object definitions in the developer repository
 - ▶ Select two object definitions to be compared (by SHIFT-clicking)
 - ▶ Select Tools > Compare Objects > Selected
- Modifying definitions is supported



Modifying Definitions

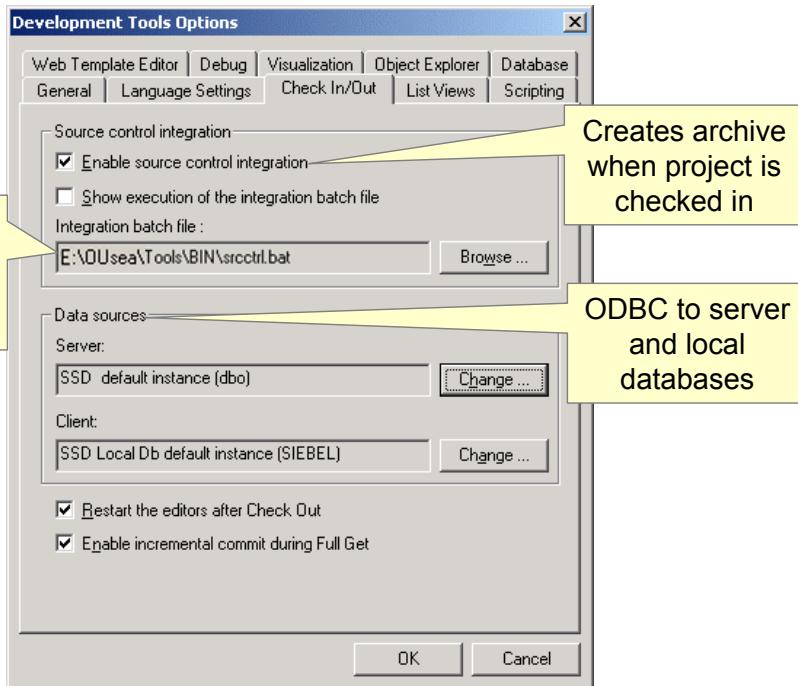
A child object definition in either parent object definition can be deleted or copied to the other definition.



Source Code Control Integration

- Automate integration with source code control for repository data

Check archive
into code
management
system



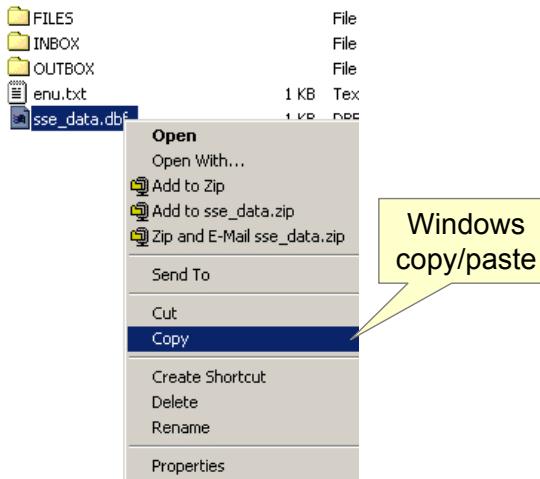
Copyright © 2007, Oracle. All rights reserved.

17 of 24



Backing Up the Development Database

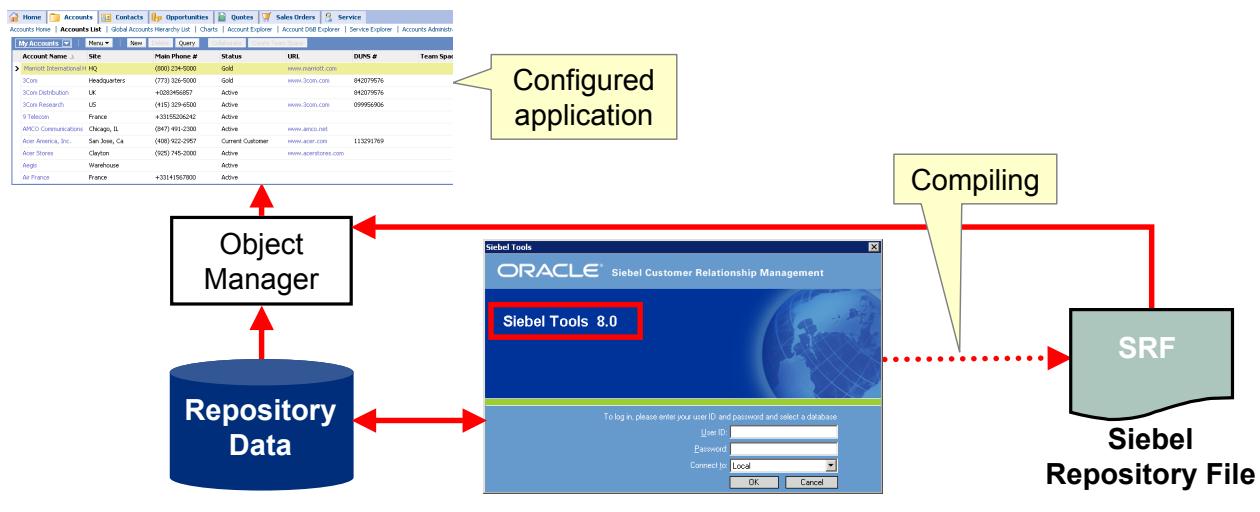
- Is achieved by copying the local database file (.dbf)
 - ▶ Backs up the entire database



ORACLE®

Compiling

- Compile to produce an .srf file to test the configuration
 - ▶ Reads object definitions from repository and creates a flat file representation to be accessed by siebel.exe at run time
- Compile into .srf file located in the objects directory of the client application



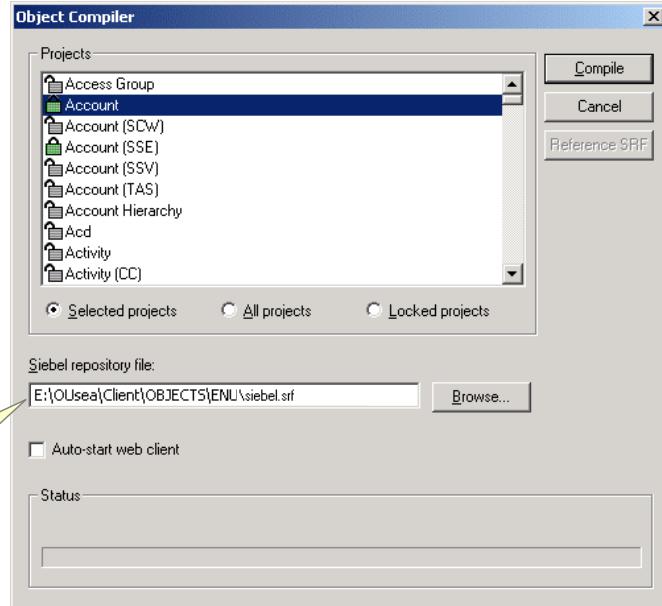
Copyright © 2007, Oracle. All rights reserved.

19 of 24

ORACLE®

Compile Projects

- When starting and at milestones
 - ▶ Use Compile All Projects to generate a new .srf file
- Compile only selected projects or locked projects for incremental compile
- Back up the .srf file prior to full compile
- To compile projects:
 - ▶ Select Tools > Compile Projects
 - ▶ Select project or projects



Copyright © 2007, Oracle. All rights reserved.

20 of 24

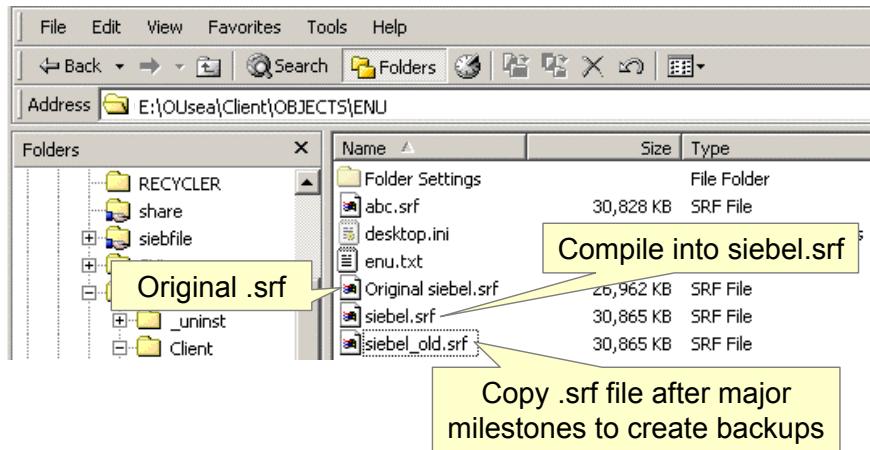
Get Projects

Remember to compile any projects after using Get; this ensures that the object definitions edited by your fellow developers are compiled into your .srf file.

ORACLE®

Backing Up the Configuration File

- Copy the .srf file to create backups
 - ▶ Create an initial backup copy before starting development
 - ▶ Create additional backups after major milestones
- Rename the copies and compile into siebel.srf
 - ▶ Using the original siebel.srf name for development eliminates the need to edit .cfg files to point to a renamed .srf



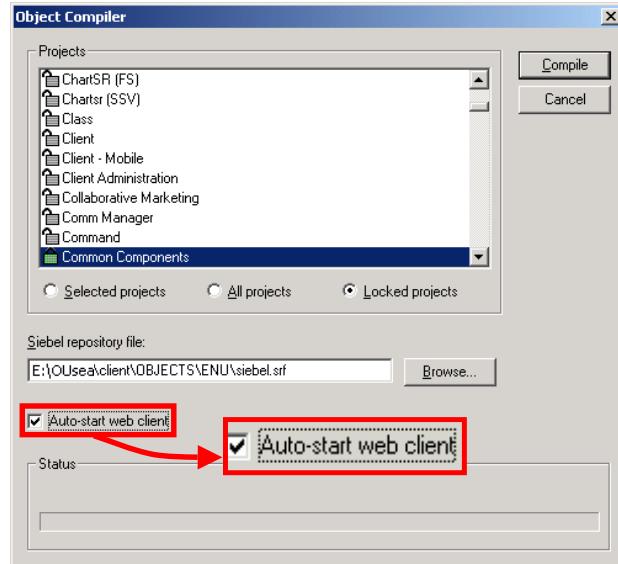
Copyright © 2007, Oracle. All rights reserved.

21 of 24



Rapid Unit Testing

- Allows developers to test changes in a local instance of the Developer Web Client
- Run-time behavior when user compiles SRF
 - ▶ If Web client is running:
 - At end of the compile, the browser window refreshes
 - The user is brought back to the view that was displayed before the compile
 - ▶ If Developer Web Client is not running:
 - If user checked auto-start, Web client is started using debug setting



Copyright © 2007, Oracle. All rights reserved.

22 of 24



Module Highlights

- Object definitions can be created and edited once the project is locked
- Validation determines correctness of object definitions
- Developer can generate archives that contain exported projects
 - ▶ Share projects for back-up and version-control
 - ▶ Export projects to and import from archive
 - ▶ Highlight import conflicts and project differences
- Compile to produce the .srf used to test a configuration
- Compiling:
 - ▶ Reads object definitions from repository tables
 - ▶ Creates a flat file (.srf) accessed by the Siebel application at run time
- Rapid Unit Testing allows developers to test changes using the Developer Web Client

Copyright © 2007, Oracle. All rights reserved.

23 of 24



Lab

- In the lab you will:
 - ▶ Practice editing and validating object definitions in Siebel Tools
 - ▶ Explore the use of diff and archive options in Siebel Tools
 - ▶ Compile an .srf file



24

Siebel 8.0 Essentials

Module 24: UI Layer Configuration: Web Templates

24

Copyright © 2007, Oracle. All rights reserved.



Module Objectives

- After completing this module you will be able to:
 - ▶ Describe the role of Siebel Web template files
 - ▶ Describe the role of Siebel tags in template files
 - ▶ List the types of templates
 - ▶ Describe how to register, associate, and bind a template file
- Why you need to know
 - ▶ Understanding template structure and how to register, associate, and bind template files is necessary in order to customize the look and feel of your Siebel application

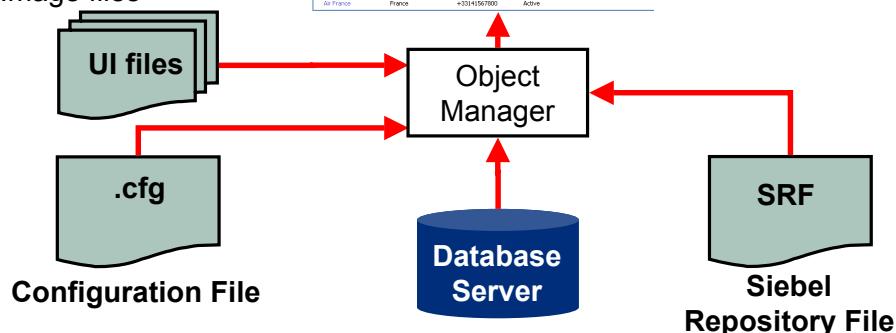
Tailoring the Physical User Interface (UI)

- Is performed by editing template files and cascading style sheets

Use an HTML or a text editor to modify layouts

Use a text editor to modify colors, fonts, etc.

- Siebel template files
- Cascading style sheets
- Image files



Copyright © 2007, Oracle. All rights reserved.

3 of 28

Reference

Siebel Developer's Reference



Siebel Web Template Files

- Are HTML files that specify how to render the views for a Siebel application in the user's browser
- Specify the formatting and layout of:
 - ▶ Graphic and text elements
 - ▶ User data
- Consist of standard HTML tags and Siebel-specific swe: tags

```
<table width="100%" class="banner" cellpadding='0'  
       cellspacing='0' border='0'>  
  
    <tr>  
  
        <td></td>  
  
        <td><swe:menu type="Default" width="275" height="29"  
bgcolor="#ccccff" fgcolor="#000000" /></td>
```

Standard
HTML tag

Siebel tag

Copyright © 2007, Oracle. All rights reserved.

4 of 28

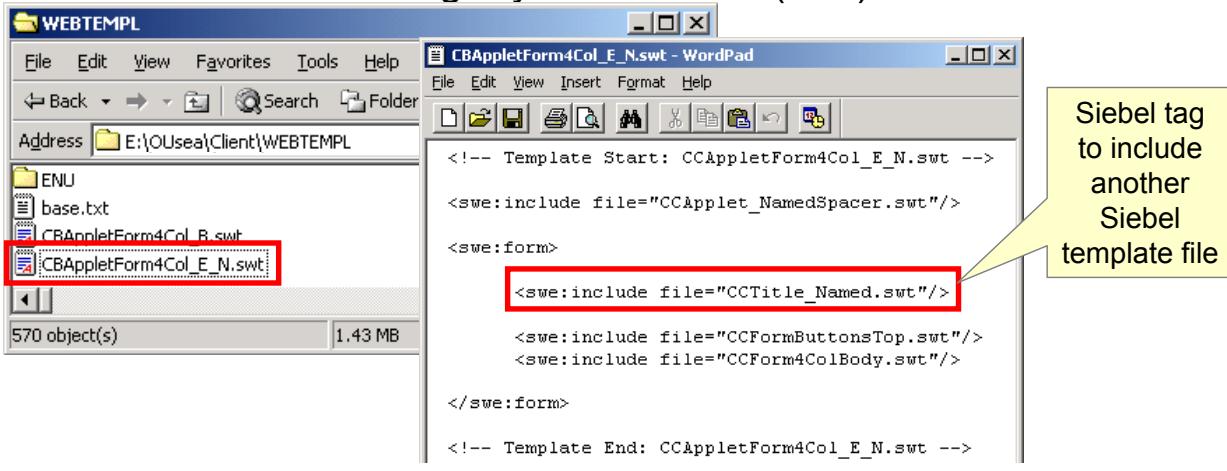
Sample HTML

The sample HTML tags shown in the slide are excerpted from the CCFrameBanner.swt template file.

Siebel Web Template Files Continued

24

- Are stored as separate files outside the Siebel repository
 - ▶ Located in the WEBTEMPL directory in a Siebel installation
 - ▶ Have an .swt extension
- Are edited using text or HTML editor of your choice
- Can include other template files via swe:include tag
- Reference Cascading Style Sheet files (.css)



Copyright © 2007, Oracle. All rights reserved.

5 of 28



Siebel Tags

- Are special tags inserted into template files
 - ▶ They specify how objects defined in the repository should be laid out and formatted in HTML for display in the user's Web browser
- Control layout repetitively through iteration
- Serve as placeholders for Siebel-specific content

```
<swe:for-each-screen>           Siebel tag to iterate over all screens
    <swe:screenlink state="Active"><td id="swe:this.Id" .....
    
        <swe:this property= FormattedHtml">&nbsp;<swe:screenname/>
        &nbsp;</swe:this></td>
    </swe:screenlink>
...
</swe:for-each-screen>
```

Siebel tag to insert the name of the screen

Copyright © 2007, Oracle. All rights reserved.

6 of 28

Sample HTML

The sample HTML tags shown in the slide are excerpted from the [CCScreenbar_Tabs.swt](#) template file.

Cascading Style Sheets

- Can be modified to define global parameters
 - ▶ Fonts, foreground and background colors, and so forth
- Style sheets:
 - ▶ Are located in PUBLIC\enu\FILES directory in a Siebel installation
 - ▶ Can be modified by customers
- Edit style sheet files to achieve the desired look and style for your implementation
 - ▶ Backup the current .css files
 - ▶ Edit .css files with your preferred HTML editor

From main.css

```
/*globalMenu Definitions*/
/*
TR.globalMenu, TD.globalMenu,
TD.globalMenu A,
TD.globalMenu A:visited,
TD.globalMenu A:hover
{
    font-size: 8pt;
    color:#000000;
    background-color: #ccccff;
    font-weight:normal;
}
```

Adjust fonts

Adjust colors

Copyright © 2007, Oracle. All rights reserved.

7 of 28

File Locations

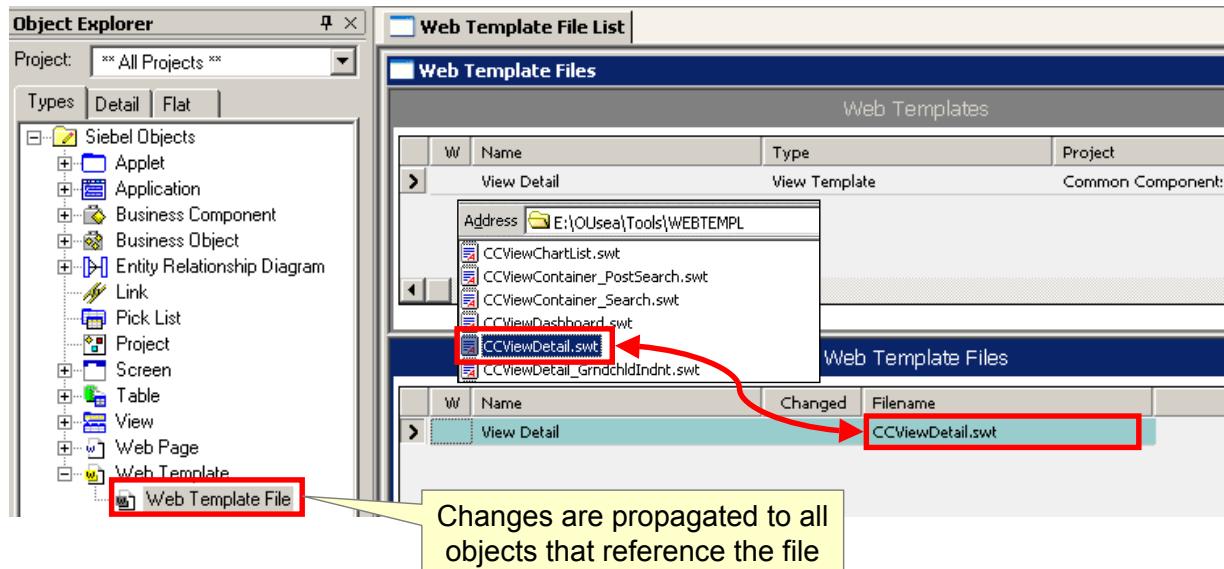
Cascading style sheet files (.css files) are located in:

- The Siebel Server installation directory
[siebsrvr_root]\WEBMASTER\files\[language_code]
- The Mobile or Dedicated Web Client installation directory
[client_root]\PUBLIC\[language_code]\FILES
- The Tools installation directory
[tools_root]\PUBLIC\[language_code]\FILES

ORACLE®

Web Template Object Definition

- Is a layer of abstraction between a UI object definition (View, Applet, or Web Page) and the file itself
- UI object definitions reference Web Template object definitions
- References an HTML Web template file



Copyright © 2007, Oracle. All rights reserved.

8 of 28

Web Template Placeholders

- Contain placeholders for Siebel logical UI definitions
- Definitions map to Siebel Web templates

The screenshot illustrates the Siebel UI configuration process. On the left, the 'Object Explorer' shows a tree view of Siebel Objects, including Applet, Application, Business Component, Business Object, Entity Relationship Diagram, Link, Pick List, Project, Screen, Table, and View. The 'View (ABC New View) - View Web Template...' tab is selected. The main area displays three placeholder components:

- Parent Applet**: A large rectangular placeholder.
- Child Applet**: A smaller rectangular placeholder below the Parent Applet.
- Child Applet (HI Display Only)**: A smaller rectangular placeholder at the bottom.

Two red arrows point from the Parent Applet and Child Applet to their corresponding representations in a Siebel application window on the right. The application window is titled '3Com' and shows a form for creating or editing account information. Below it is a grid view of employee data:

Mr/Ms	First Name	Last Name	Job Title	Work Phone #
Mr.	Shashi	Aamot	IT Manager	(614) 343-8700
Mr.	Avram	Ahl	Vice President	(773) 555-9870
Mr.	Alan	Genever	Manager, Account Relat	(773) 555-6500
Ms.	Kristy	Helena	Supervisor, Logistics	(773) 555-2223
Ms.	Jill	Ishi	Account Manager	(773) 555-5312
Mr.	Bob	Kreisberg	Programmer	(650) 786-8767

A yellow callout box labeled 'Placeholder' points to the bottom-most placeholder component.

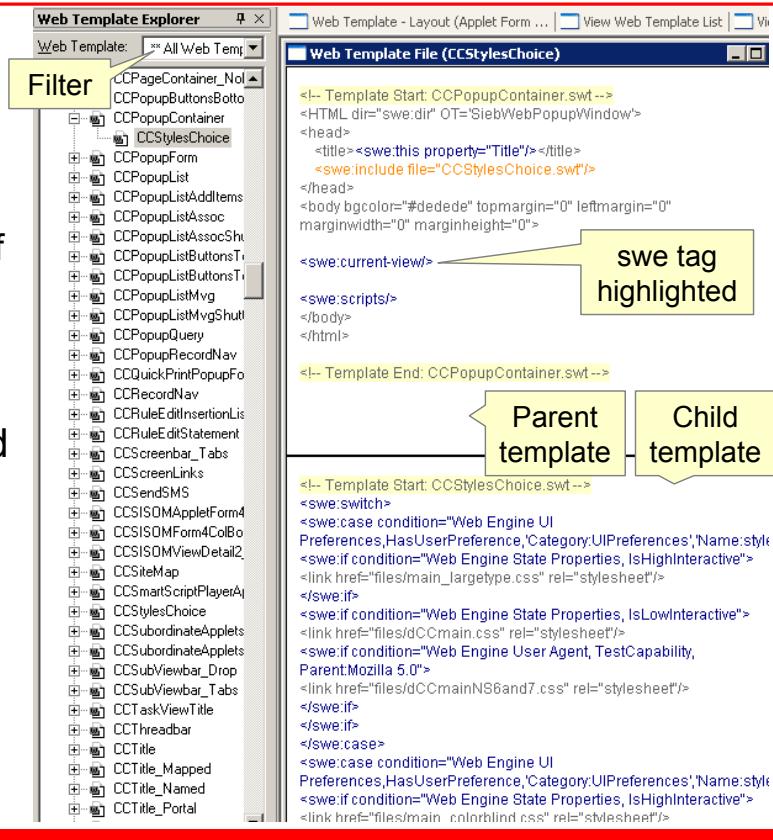
Copyright © 2007, Oracle. All rights reserved.

9 of 28

ORACLE®

Web Template Explorer

- Is used in Tools to:
 - ▶ Identify template files included in a template file
 - ▶ Examine the contents of a template file
 - ▶ Invoke an editor to modify the template file
- Highlights Siebel-supplied tags for easy viewing
- Navigate to View > Windows > Web Templates Window



Copyright © 2007, Oracle. All rights reserved.

10 of 28

Types of Web Templates

24

Form Applet Web Templates

List Applet Web Templates

View Web Templates

Web Page Web Templates

Copyright © 2007, Oracle. All rights reserved.

11 of 28

Note

This list of Web Templates is not comprehensive; there are also Tree Applet Web Templates and Specialized Web Templates. Refer to Siebel Bookshelf for more information.

Form Applet Web Templates

- Define the layout of form applets
 - ▶ Placeholder provides starting point for building applet
 - ▶ Tools' Web Layout Editor assists you in setting template file parameters

The diagram illustrates the process of creating a form applet. It is divided into two main sections: **Tools** and **UI**.

Tools: This section shows the **Web Layout Editor**, which is used to define the layout of form applets. A red box highlights the editor's interface, which includes a toolbar with buttons like New, Edit, Delete, Save, and Query, and a main area for defining fields. A callout box states: "Account Entry applet uses Applet Form Grid Layout". A red arrow points from the editor's interface down to the UI section.

UI: This section shows the resulting user interface for a 3Com account entry. The UI is built using the template defined in the Web Layout Editor. It features a grid layout with various input fields for account information such as Account Name, Address, City, Zip Code, Site, State, Country, Account Team, Main Phone #, Main Fax #, and URL. A callout box states: "Object Manager uses template to build applet".

Copyright © 2007, Oracle. All rights reserved.

12 of 28

List Applet Web Templates

24

- Define the layout of a list applet
 - ▶ Standard Interactivity list applets use Base and Edit modes

Base Mode

My Service Requests			
New	Query		
SR #	Status	Created	Summary
1-18605	Open	3/22/2004 12:43:46 PM	Ethernet card malfunctioning
1-18601	Open	3/22/2004 12:41:42 PM	Trouble distributing packaged

A read-only list of data with buttons for editing

Edit Mode

My Service Requests	
SR #:	<input type="text"/>
Status:	<input type="button" value=""/>
Summary:	<input type="text"/>
Product:	<input type="text"/>
Serial #:	<input type="text"/>
Created:	<input type="text"/> 
<input type="button" value="Go"/> <input type="button" value="Cancel"/>	

An editable form for querying and data entry

List Applet Web Templates Continued

- High Interactivity clients use EditList mode

EditList Mode

The screenshot illustrates the EditList Mode interface for managing account data. At the top, there is a toolbar with buttons for 'My Accounts' (dropdown), 'Menu' (dropdown), 'New', 'Delete', 'Query' (highlighted with a red box), 'Collaborate', and 'Create Team Space'. Below the toolbar is a table listing account information:

Account Name	Site	Parent Account Name	Parent Account Site
Marriott International H HQ			
AT&T	Edison, NJ		
Akamai Technologies, In	Cambridge, MA		
Art.net	Sterling, VA		
Broadband e2e	Las Angelas, CA		
Chase Manhattan Bank	Manhattan, Ny		
Digital River, Inc.	San Francisco, Ca		

An annotation with a yellow box and arrow points to the 'Query' button in the toolbar, with the text 'An editable list of data'.

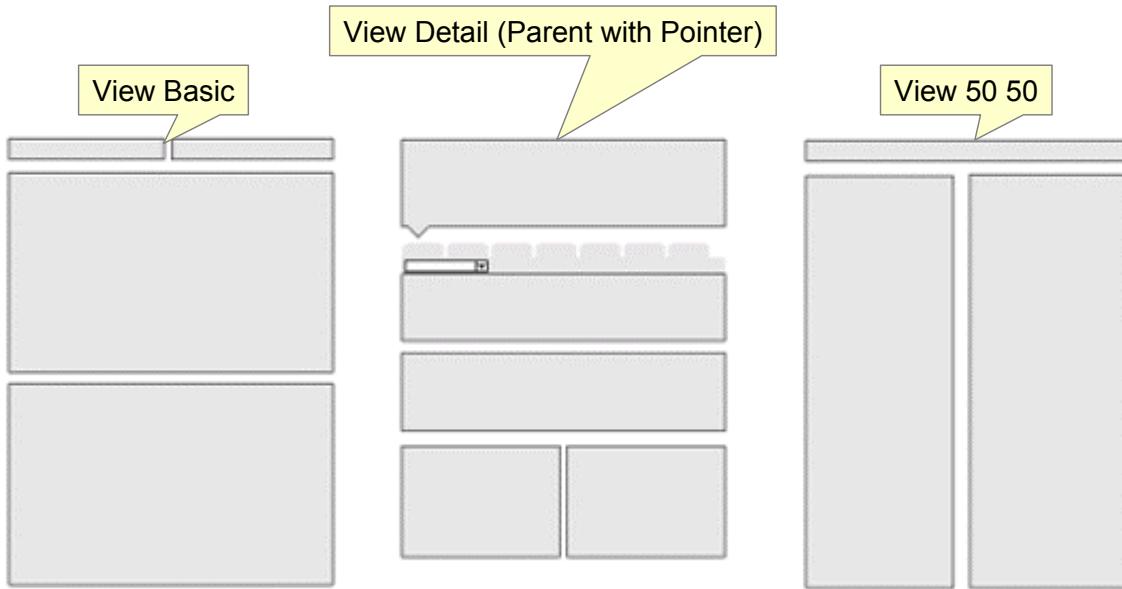
Below the table is another toolbar with buttons for 'My Accounts' (dropdown), 'Menu' (dropdown), 'Go', 'Cancel', 'Query Assistant', and 'Enter Query' (highlighted with a red box). A new row is being added to the table, indicated by a cursor in the 'Account Name' field. The new row has three columns labeled '<Case Required>' and one empty column for 'Parent Account Site'.

An annotation with a yellow box and arrow points to the 'Enter Query' button in the second toolbar, with the text 'EditList mode supports querying and data entry directly in the list'.

View Web Templates

24

- Define the layout of a view
 - ▶ Common view templates include:



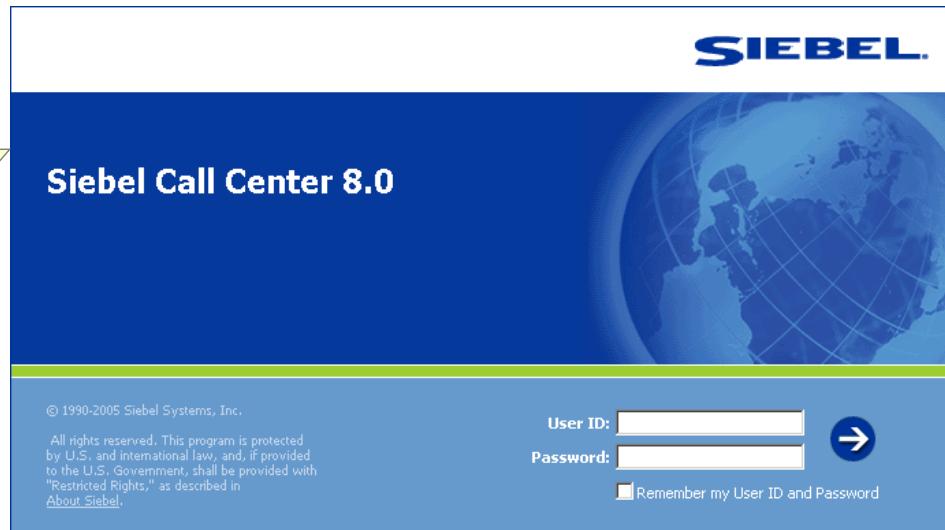
Copyright © 2007, Oracle. All rights reserved.

15 of 28

Web Page Web Templates

- Define the layout of Web pages such as:
 - ▶ Login Page
 - ▶ Error Page
 - ▶ Container Page

The Login Web template defines the layout of the Login page

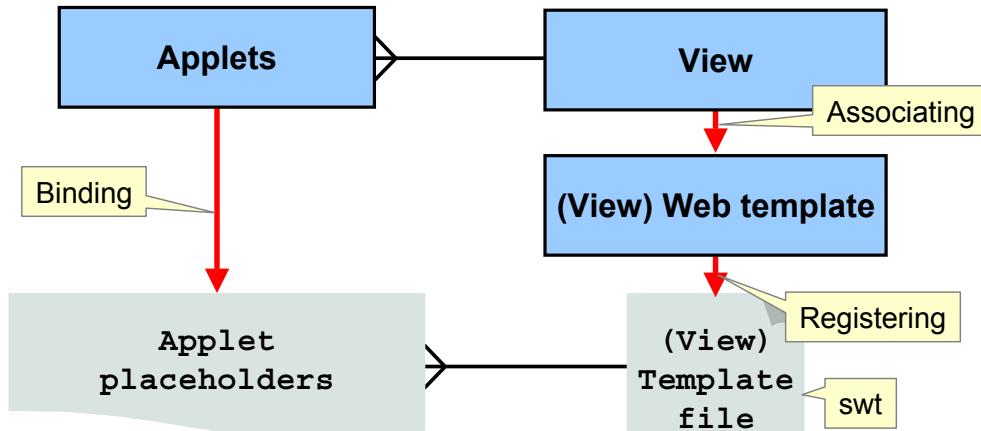


Using Template Files

24

- Requires three types of relationships:
 - ▶ Registering creates an object definition in the repository
 - ▶ Associating identifies the Web template used to render an applet or view
 - ▶ Binding fastens a control to a specific position on the page or an applet to a view

View Example



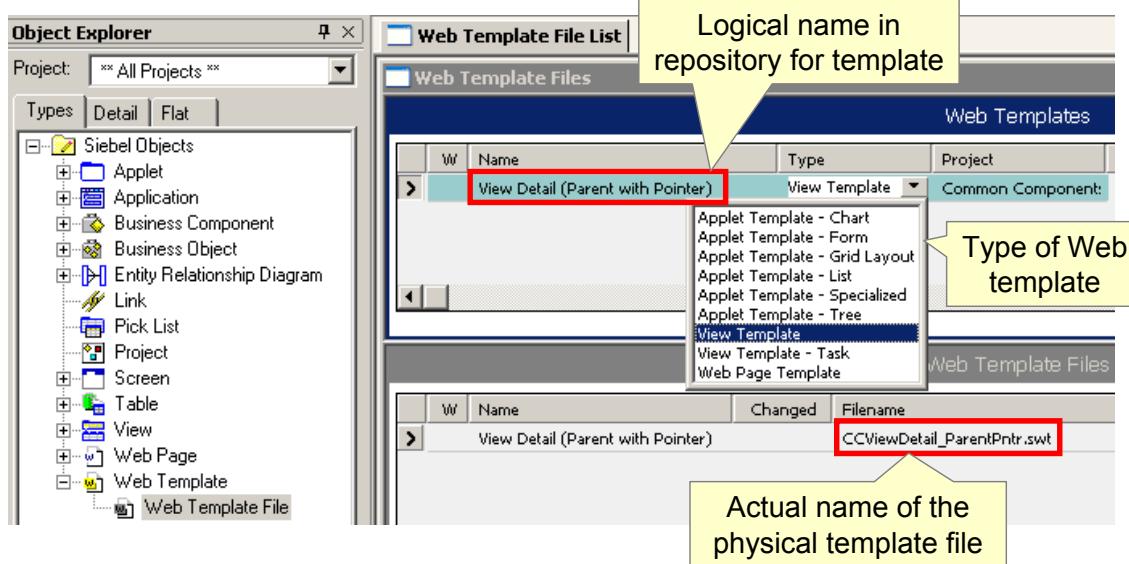
Copyright © 2007, Oracle. All rights reserved.

17 of 28

ORACLE®

Registering a Template File

- Creates a Web Template object definition to define the logical name for the template
- Creates a Web Template File object definition to reference the actual physical file



Copyright © 2007, Oracle. All rights reserved.

18 of 28

Registering

Siebel applications are delivered with hundreds of template files, all of which are registered. You will only need to register a template file in the unlikely event that you need to create a new file.

Associating an Applet with Its Template

- Creates an Applet Web Template object definition to specify the associated Web template
 - Different applet modes can reference different Web templates

The screenshot shows the Siebel Object Explorer on the left and the Applet Web Template List window on the right.

Object Explorer: Shows the project as "All Projects" and the types as "Types". The tree view under Siebel Objects shows the structure: Applet > Applet Method Menu > Applet Web Template > Applet Web Temp > Control > Drilldown Object > List > Tree.

Applet Web Template List: The title bar says "Applet Web Template List". Inside, there's a sub-window titled "Applet Web Templates" showing the "Applets" section. It lists one item: "Account List Applet" with "Business Component" set to "Account". Below this is the "Applet Web Templates" section, which contains a table:

Name	Type	Web Template
Base	Base	Applet List (Base/EditList)
Edit	Edit	Applet Form Grid Layout
Edit List	Edit List	Applet List (Base/EditList)

Annotations with yellow callouts explain the columns:

- A callout points to the "Type" column with the text "Mode for the applet".
- A callout points to the "Web Template" column with the text "Applet template used to render applet for mode".

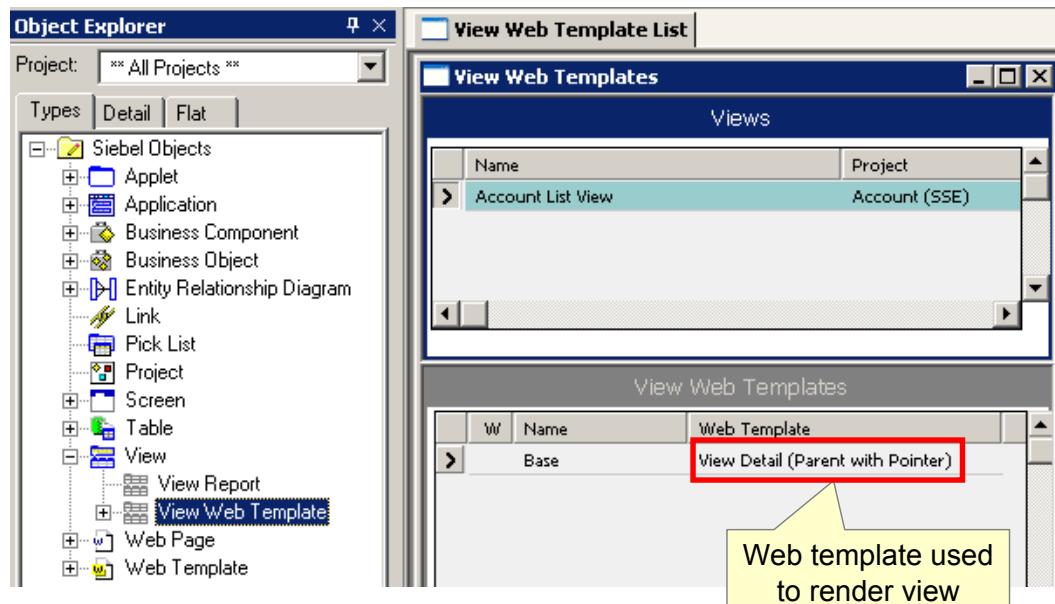
At the bottom of the main window, there are copyright and page navigation details:

Copyright © 2007, Oracle. All rights reserved. 19 of 28

ORACLE®

Associating a View with Its Template

- Creates a View Web Template object definition to specify the associated Web template
 - ▶ Each view references a single view Web template

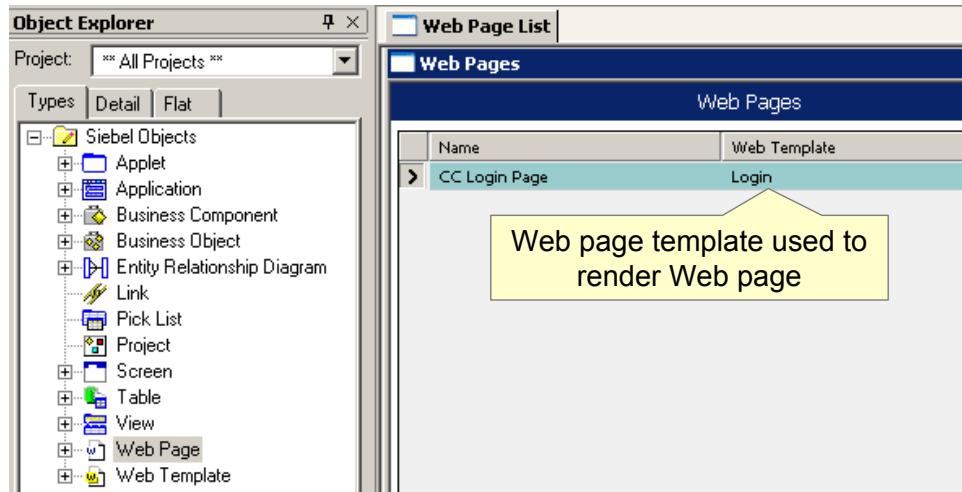


Copyright © 2007, Oracle. All rights reserved.

20 of 28

Associating a Web Page with Its Template

- Creates a Web Page object definition and sets the Web Template property



Copyright © 2007, Oracle. All rights reserved.

21 of 28

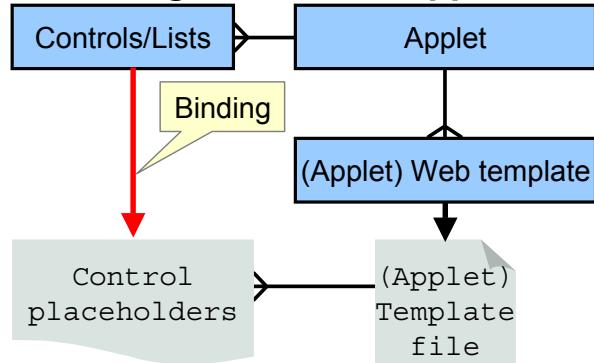
Web Page Template Web page templates are templates that are not view or applet Web templates. Examples of common Web page templates are login page, error page, and container page.

ORACLE®

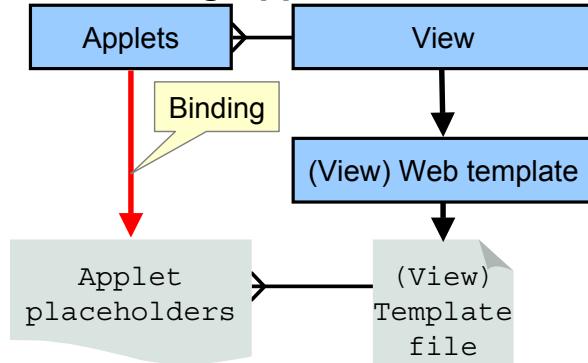
Binding

- Maps the components of an applet or view to placeholders in the corresponding template file
- Information is specified in the repository
 - ▶ View Web Template Item
 - ▶ Applet Web Template Item
 - ▶ Web Page Item
 - ▶ Results in the corresponding element being displayed at run time

Binding Controls to Applets



Binding Applets to Views



Copyright © 2007, Oracle. All rights reserved.

22 of 28

Applet Web Template Items

- Are created by the binding process
 - ▶ Item Identifier is the “id” attribute value in the swe:control tag
- Are used at run time by the Siebel Web Engine to identify the controls that make up an applet

The screenshot shows the Siebel Object Explorer window on the left and the Applet Web Template Item List window on the right.

Object Explorer:

- Project: ** All Projects **
- Types: Siebel Objects
 - Applet
 - Applet Method MenuItem
 - Applet Web Template
 - Applet Web Template Item
 - Control
 - Drilldown Object
 - List
 - Tree
 - Application
 - Business Component
 - Business Object
 - Entity Relationship Diagram
 - Link

Applet Web Template Item List:

W	Name
>	Query

Applet Web Template Items:

W	Name	Control	Item Identifier
	SR Number	SR Number	1,300
	Status	Status	1,301
	SummaryLong	SummaryLong	1,302
	ProductText	ProductText	1,303
	Serial Number	Serial Number	1,304
	Created	Created	1,305

A red arrow points from the highlighted 'Item Identifier' column in the table above to the 'Item Identifier' attribute in the code below.

```
<swe:for-each count="6" startValue="1300" iteratorName="currentId">
  <swe:control id="swe:currentId" hintMapType="FormItem">
```

Copyright © 2007, Oracle. All rights reserved.

23 of 28

swe:control Tag

The swe:control tag shown in the slide is in the dCCForm1Col.swt template file. This file is included in the dCCAppletForm1Col.swt template file which is registered as the Web Template File for the DotCom Applet Form 1-Column Web Template.

This tag illustrates the use of the swe:for-each tag. In this case the swe:control tag is executed repeatedly with values from 1300 to 1305.

ORACLE®

View Web Template Items

- Are created by the binding process
 - ▶ Item Identifier is the “id” attribute value in the swe:applet tag
- Are used at run time by the Siebel Web Engine to identify the applets that make up a view

The screenshot shows the Siebel Object Explorer and the View Web Template Items window. In the Object Explorer, under the Siebel Objects category, the View node is expanded, showing View Report, View Web Template, and View Web Template Item. The View Web Template Item node is selected. In the View Web Template Items window, there are two tables. The top table, 'View Web Templates', has one row named 'Base'. The bottom table, 'View Web Template Items', has two rows: 'Account Entry Applet' (Item Identifier 2) and 'Account List Applet' (Item Identifier 1). A red arrow points from the 'Item Identifier' column of the bottom table to the 'id="1"' attribute in the XML code below.

```
<swe:applet hintMapType="Applet" id="1" hintText="Parent Applet">
```

Copyright © 2007, Oracle. All rights reserved. 24 of 28

swe:applet Tag

The swe:applet tag shown in the slide is in the CCViewDetail_ParentPntr.swt template file. This file is registered as the Web Template File for the View Detail (Parent with Pointer) Web Template.

Web Page Items

- Identify the links and buttons appearing on a Siebel Web page
 - ▶ Item Identifier is the “id” attribute value in the swe:pageitem tag
- Are used at run time by the Siebel Web Engine to position the links and buttons

The screenshot shows two tables side-by-side. The left table is titled 'Web Pages' and lists a single entry: 'CC Container Page (eSales)' under the 'Name' column, with 'eSales' under 'Project'. The right table is titled 'Web Page Items' and lists several items:

	W	Name	Item Identifier	Type
>		Shopping Cart	12	Link
		ProfileButton	13	Link
		HelpButton	14	Link
		ContactButton	15	Link
		LogoutButton	17	Link

A red box highlights the 'Item Identifier' column in the 'Web Page Items' table. A red arrow points from the top of this box down to the 'id' attribute values in the corresponding row of the HTML code below.

```

<td class="banner">
  <table border="0" cellspacing="0"
    <swe:pageitem id="11"><td nowrap>
    <swe:pageitem id="12"><td nowrap>
    <swe:pageitem id="13"><td nowrap>
    <swe:pageitem id="14"><td nowrap>
    <swe:pageitem id="15"><td nowrap>
    <swe:pageitem id="16"><td nowrap>
    <swe:pageitem id="17"><td nowrap>
    <swe:pageitem id="18"><td nowrap>
    <swe:pageitem id="19"><td nowrap>
  </tr></table>
</td>

```

Copyright © 2007, Oracle. All rights reserved.

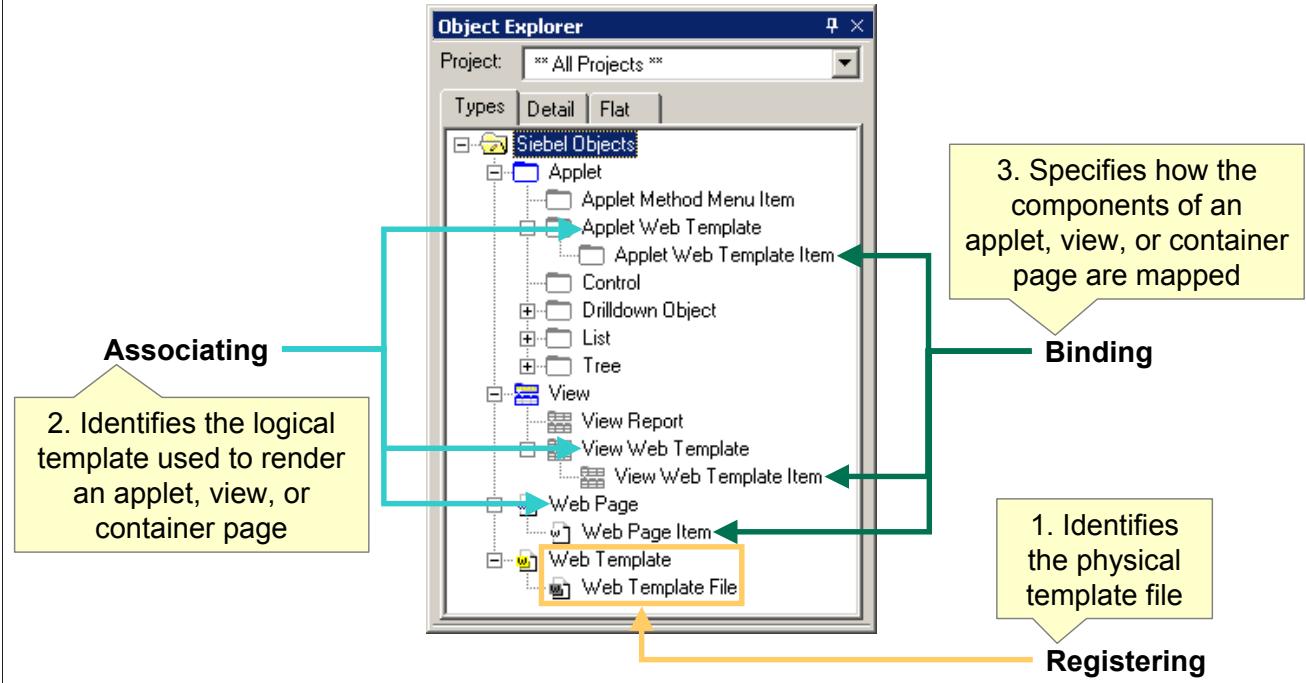
25 of 28

swe:pageitem Tags The HTML and SWE tags shown in the slide are excerpted from the dCCFrameBanner.swt template file.

ORACLE®

Summary of Object Definitions

- Registering, associating, and binding adds object definitions to the repository



Module Highlights

- Siebel Web Template Files are HTML files that specify how to render views for a Siebel application in a user's browser
 - ▶ Can include other template files
 - ▶ Can reference CSS files
- Siebel Tags are a Siebel-developed library of tags in Web template files processed at run time
- CSSs can be modified to define global parameters such as fonts and colors
- The four Web Template types are Form Applet, List Applet, View, and Web Page
- Template files require three types of relationships: Associate, Register, and Bind



Lab

- In the lab you will:
 - ▶ Examine Web templates

Copyright © 2007, Oracle. All rights reserved.

28 of 28

ORACLE®

Siebel 8.0 Essentials

25

Module 25: UI Layer Configuration: Applets

25

Copyright © 2007, Oracle. All rights reserved.



Module Objectives

- After completing this module you should be able to:
 - ▶ Create and modify a list applet
 - ▶ Create and modify a form applet
- Why you need to know:
 - ▶ You may want to display new data in the UI

ORACLE®

Primary Applet Types: List Applet

- Displays one or more records simultaneously in a list
- Fields for one record are displayed in a single row
- Typically displays fewer fields than a form applet due to width of monitor
- Supports drilldown to other views through hyperlinked fields

Account Name	Site	Industries
Art.net	Sterling, VA	prepackaged software
Broadband e2e	Las Angelas, CA	communications equipment
Chase Manhattan Bank	Manhattan, Ny	bank holding companies
Digital River, Inc.	San Francisco, Ca	computer related services
First Record, Inc	HQ	
Honeywell Intl (Allied Signal Aero)	Hq-Morristown, NJ	aircraft engines & engine parts
IBM Corporation	Poughkeepsie, NY	computer peripheral equipment
Imperial Tobacco	Theobald Road, London, tobacco & tobacco products	
Lexis-Nexis	Dayton, OH	information retrieval services

Copyright © 2007, Oracle. All rights reserved.

3 of 24

25

Reference

Configuring Siebel Business Applications: Configuring Applets

ORACLE®

Primary Applet Types: Form Applet

- Displays a single record
- Fields for the record are displayed in a grid or form
- Can show more fields at once than a list applet

The screenshot shows a Siebel Form Applet titled "Art.net". The top navigation bar includes "Menu", "New", "Delete", and "Query" buttons. The main area contains several input fields grouped into two columns:

Account Name: * Art.net	Site: Sterling, VA	Account Team: JRUBIN
Address: 22110 Pacific Blvd	Address Line 2:	Main Phone #: (703) 205-3500
City: Sterling	State: VA	Main Fax #:
Zip Code: 20166-6913	Country: USA	URL: http://www.artnet.com

Copyright © 2007, Oracle. All rights reserved.

4 of 24

Additional Applet Types

The following are some additional applet types (see Siebel Bookshelf for a complete list):

- Catalog list
- Chart
- Explorer view
- Hierarchical list
- Message
- Multi-value group applet
- Pick applet
- Rich list
- Salutation

Creating a List Applet

- Follow these steps to create a new list applet ...

1. Create new applet object definition

2. Associate applet template with object definition

3. Bind controls and list columns to template

Highly recommended

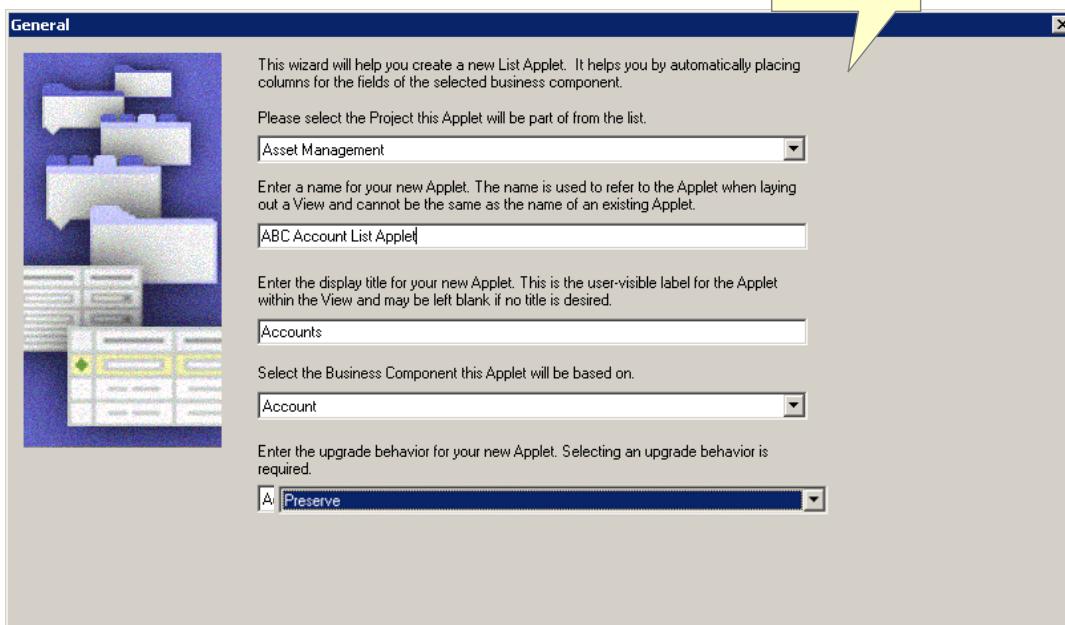
- ...Or use the List Applet Wizard

- Automates these steps
- Generally the preferred method
- Ensures that object definitions and properties are created
- Ensures that steps are not omitted
- Increases developer productivity
- Speeds project completion

ORACLE®

List Applet Wizard

- Use the List Applet Wizard to create a new applet
 - ▶ Select File > New Object, select the Applets tab, select List Applet, and then click OK



Copyright © 2007, Oracle. All rights reserved.

6 of 24



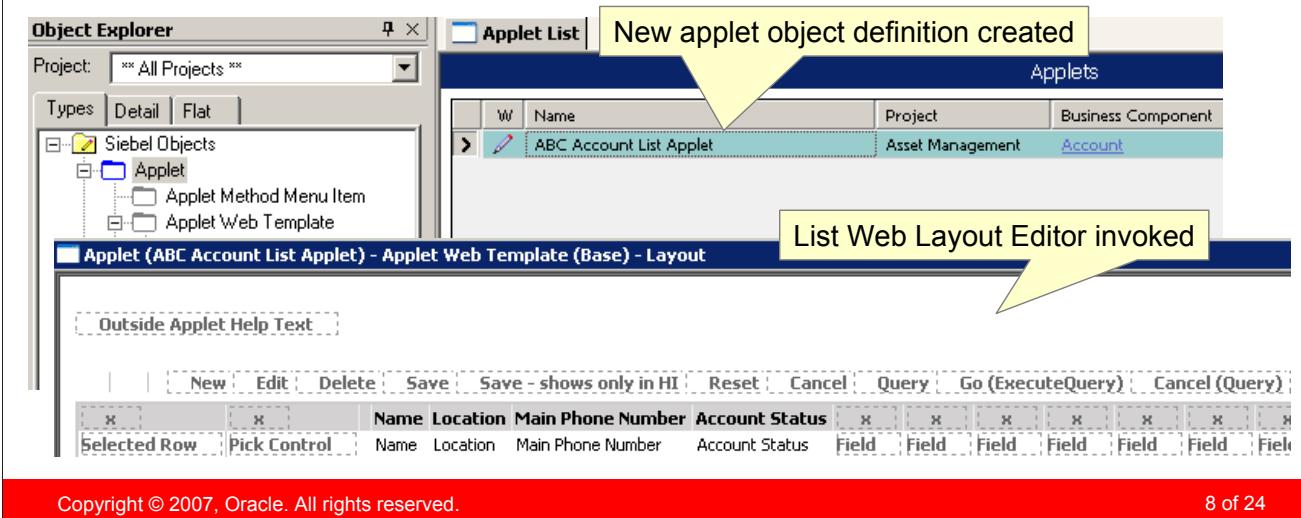
List Applet Wizard Inputs

- The List Applet Wizard requires the following inputs:
 - ▶ Project the new applet will be part of
 - ▶ Applet name and display title
 - ▶ Business component the applet will reference
 - ▶ The upgrade behavior
 - Admin, Non-preserved, Preserved
 - ▶ The Web templates that will be used for each mode
 - Base and EditList mode are commonly built using Applet List (Base/EditList)
 - Edit mode is commonly built using Applet List Edit (Edit/New/Query)
 - ▶ The business component fields that will appear in the Web layout
 - ▶ Additional controls that will be added to the applet
 - By default, all of the standard buttons are selected

ORACLE®

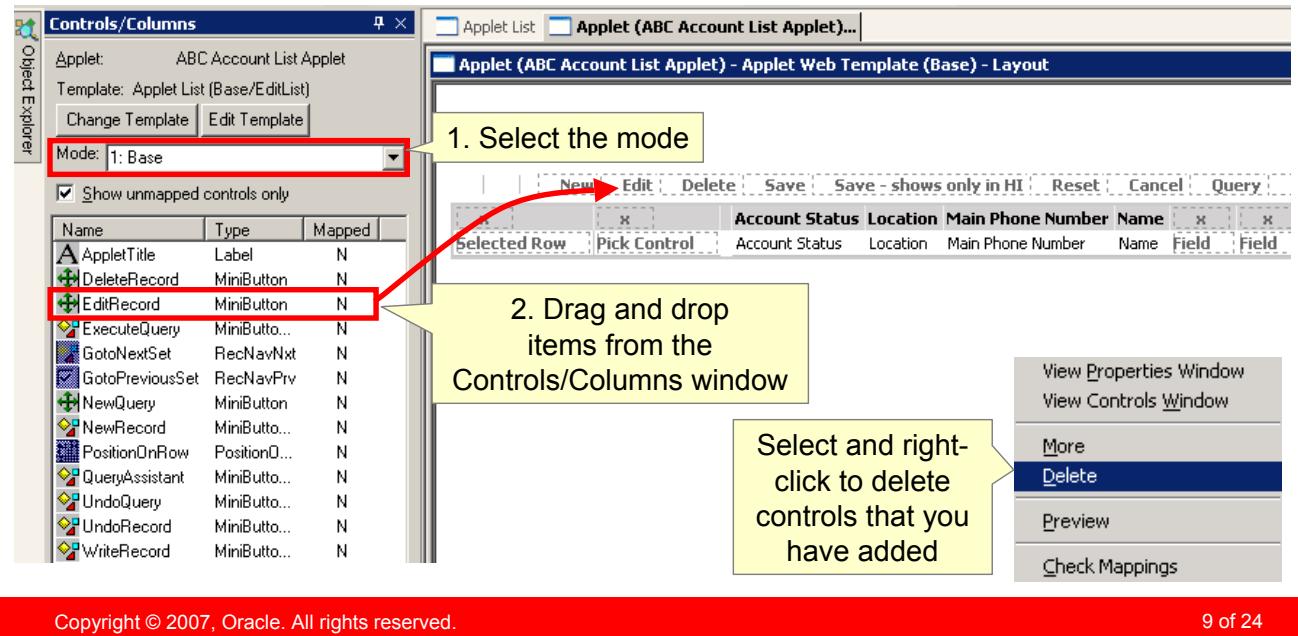
List Applet Wizard Outputs

- Creates required object definitions for controls, lists, and list columns
- Associates applet Web templates to the applet
- Binds list columns to templates
 - ▶ Appear as applet Web template items
- Invokes the Web Layout Editor



Web Layout Editor

- Is used to add, remove, and reorder applet Web template items
 - ▶ Placeholders correspond to item identifiers of applet Web template items
- Can be invoked from the right-click menu



ORACLE®

Applet Web Templates

- Specify which template is used when the applet is displayed in a given mode
- Contains Applet Web Template items
 - ▶ The controls or list columns that have been bound to the template
 - ▶ The item identifier that describes where in the applet the controls or list columns should appear

The screenshot shows the Siebel Object Explorer interface. On the left, the Object Explorer tree view is open, showing categories like Siebel Objects, Application, Business Component, etc., with Applet expanded. Under Applet, Applet Method Menu Item, Applet Web Template, Applet Web Template Item, Control, Drilldown Object, List, and Tree are listed. The Applet Web Template Item node is selected.

The main workspace displays two tables:

- Applet Web Template Item List:** A table with columns W, Name, and Web Template. It contains three rows: Base (Applet List (Base/EditList)), Edit (Applet List Edit (Edit/New/Query)), and Edit List (Applet List (Base/EditList)). The 'Web Template' column is highlighted with a red box and a callout bubble stating "Specifies which template to use".
- Applet Web Template Items:** A table with columns W, Name, Control, Type, and Item Identifier. It contains five rows: Account Status (Account Status, List Item, 501), Location (Location, List Item, 502), Main Phone Number (Main Phone Number, List Item, 503), and Name (Name, List Item, 504). The 'Item Identifier' column is highlighted with a red box and a callout bubble stating "Specifies where list column will appear".

Copyright © 2007, Oracle. All rights reserved.

10 of 24

List Column Definitions

- Specify:
 - ▶ The field that is displayed
 - ▶ The display name that appears at the top of the list column
 - May reference a symbolic string

The screenshot shows the Siebel Object Explorer on the left and the List Column List interface on the right.

Object Explorer: Shows a tree view of Siebel Objects under Project: ** All Projects **. The 'Applet' node is expanded, showing 'Applet Method Menu Item', 'Applet Web Template', 'Applet Web Template Item', 'Control', 'Drilldown Object', 'List', and 'Tree'. The 'List' node is also expanded, showing 'List Column'.

List Column List: This tab is selected in the top navigation bar. It contains two tables:

- Lists:** A table with columns W, Name, and a toolbar with edit and delete icons. One row is listed: 'List'.
- List Columns:** A table with columns W, Name, Field, and Display Name. Four rows are listed:

W	Name	Field	Display Name
>	Account Status	Account Status	Account Status
	Location	Location	Location
	Main Phone Number	Main Phone Number	Main Phone Number
	Name	Name	Name

Copyright © 2007, Oracle. All rights reserved.

11 of 24

25

Oracle Internal & Oracle Academy Use Only



Symbolic Strings

- Ensure that the same name appears exactly the same way throughout the application
- Define it once, reference it anywhere
- Provide a way to manage updates, globally
- Are used by user interface objects such as:
 - ▶ Labels
 - ▶ List columns
 - ▶ Applet titles
- Are stored in the repository containing words and phrases used in UI definitions
 - ▶ Global Dictionary contains many commonly used terms
 - ▶ Symbolic String Locale object type allows language translation

Using Symbolic Strings

- String reference:
 - ▶ Refers to symbolic string object definitions
 - ▶ Is selected from a picklist
 - ▶ Refers to the name of the symbolic string
- String override:
 - ▶ Used for exceptions when strings are infrequently displayed
 - ▶ Can be converted to symbolic strings using a Siebel-supplied string conversion utility

The screenshot shows the Siebel Object Explorer on the left and the List Column List window on the right. The Object Explorer displays a tree structure of Siebel Objects, with 'Applet' selected. The List Column List window shows a table of list columns with two rows. A yellow callout box points to the second row, which has a green background, with the text 'Reference to object in repository'. The table has columns: Name, Field, Display Name, Display Name - String Reference, and Display Name - String Override.

Name	Field	Display Name	Display Name - String Reference	Display Name - String Override
Account Status	Account Status	Account Status	SBL_ACCOUNT_STATUS-1004225945-05P	Account Status
Location	Location	Location	SBL_LOCATION-1004231258-3NO	

Copyright © 2007, Oracle. All rights reserved.

13 of 24

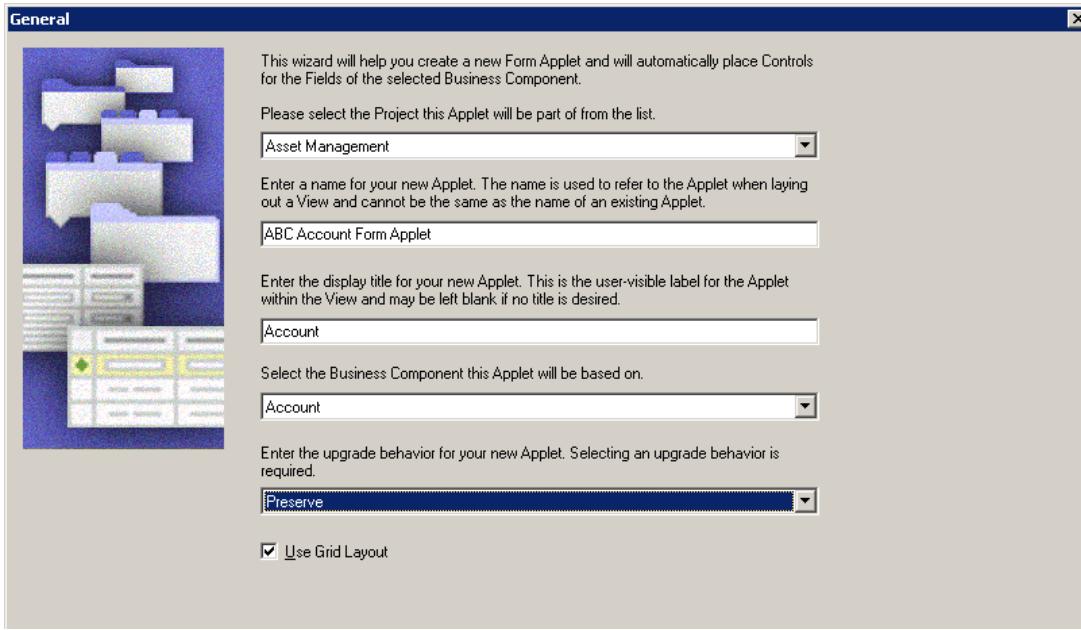
String Conversion Utility

A string conversion utility is available that exports from local tables and imports to the Symbolic String table. It allows symbolic strings to be exported to a file, translated in another application, then imported back to the Siebel application with translated values for the appropriate language. The utility also allows developers to export string overrides and re-import them as string references, and it consolidates redundant local definitions. The utility file is Strconv.bat and is located in the Tools\bin directory.

ORACLE®

Form Applet Wizard

- Use the Form Applet Wizard to create a new form applet
 - ▶ Select File > New Object, select the Applets tab, select Form Applet, and then click OK



Copyright © 2007, Oracle. All rights reserved.

14 of 24

Form Applet Wizard Inputs

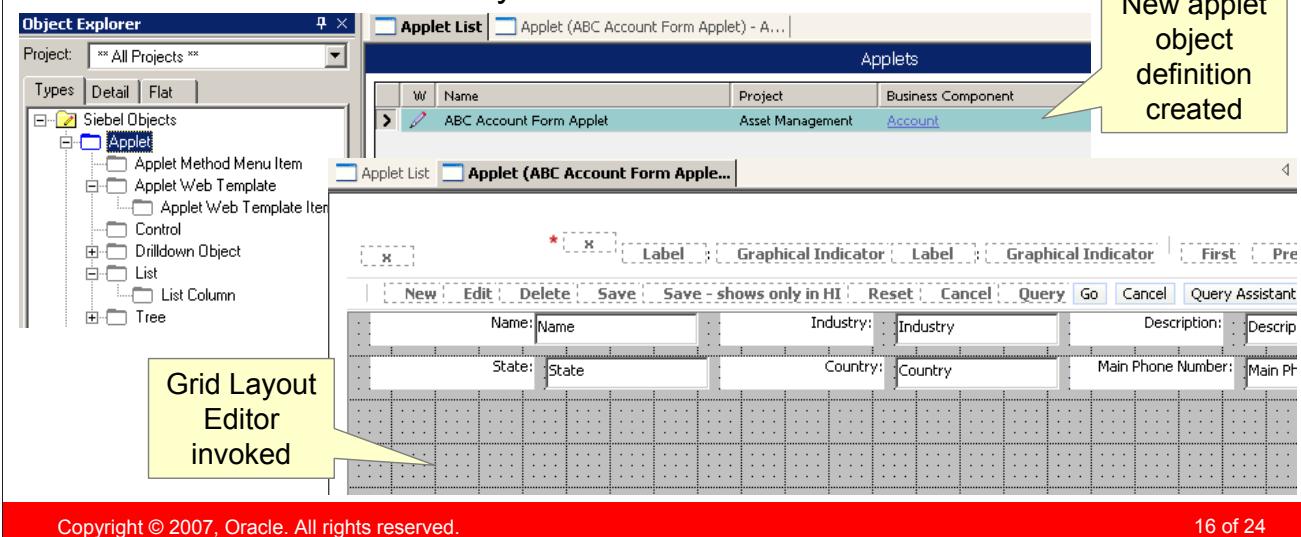
- The Form Applet Wizard requires the following inputs:
 - ▶ Project that the new applet will be part of
 - ▶ Applet name and display title
 - ▶ Business component that the applet will reference
 - ▶ Upgrade behavior
 - Admin, Non-preserved, Preserved
 - ▶ Modes to use
 - Edit mode is required for all clients
 - ▶ Uses the Applet Form Grid Layout template
 - Base mode is required for Standard Interactivity clients only
 - ▶ Business component fields to appear in Web layout
 - Creates controls and binds them to applet Web templates
 - ▶ Additional controls to be added to applet
 - By default, all standard buttons are selected

ORACLE®

Form Applet Wizard Outputs

■ The Form Applet Wizard

- ▶ Creates required object definitions for Controls
- ▶ Associates applet Web templates to the applet
- ▶ Binds columns to templates
 - Appear as applet Web template items
- ▶ Invokes the Grid Layout Editor

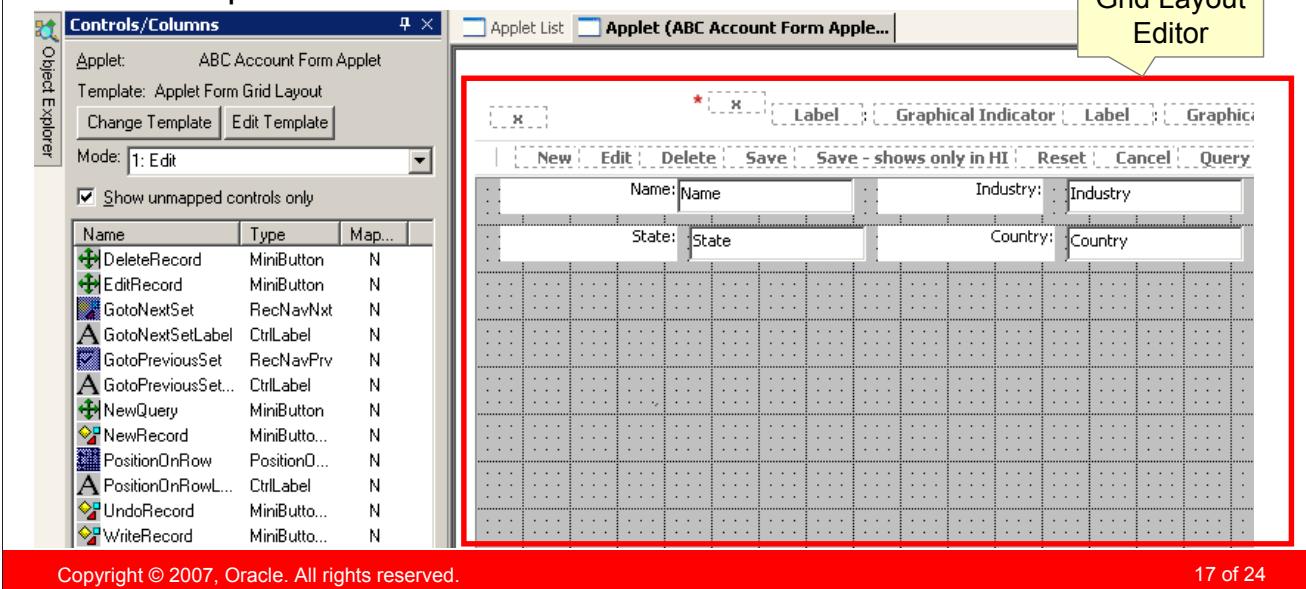


ORACLE®

Grid Layout Editor

25

- Is used to:
 - ▶ Create new control object definitions
 - ▶ Bind controls to the Web template
 - ▶ Resize controls
 - ▶ Reposition controls



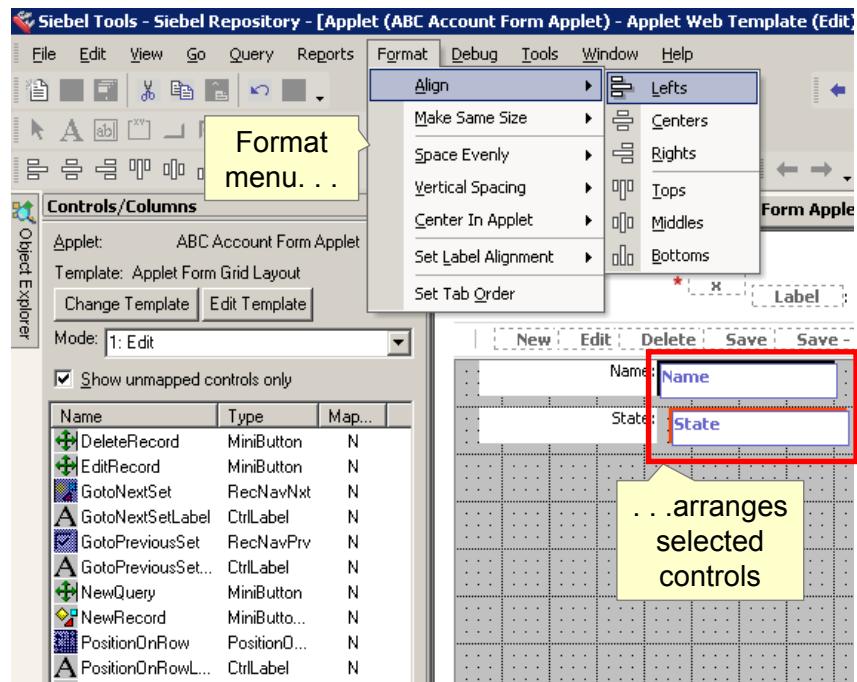
Copyright © 2007, Oracle. All rights reserved.

17 of 24

ORACLE®

Grid Layout Editor Continued

- Allows drag-and-drop layout editing
- Supports formatting options such as:
 - ▶ Align controls
 - ▶ Make same size
 - ▶ Horizontal spacing
 - ▶ Vertical spacing

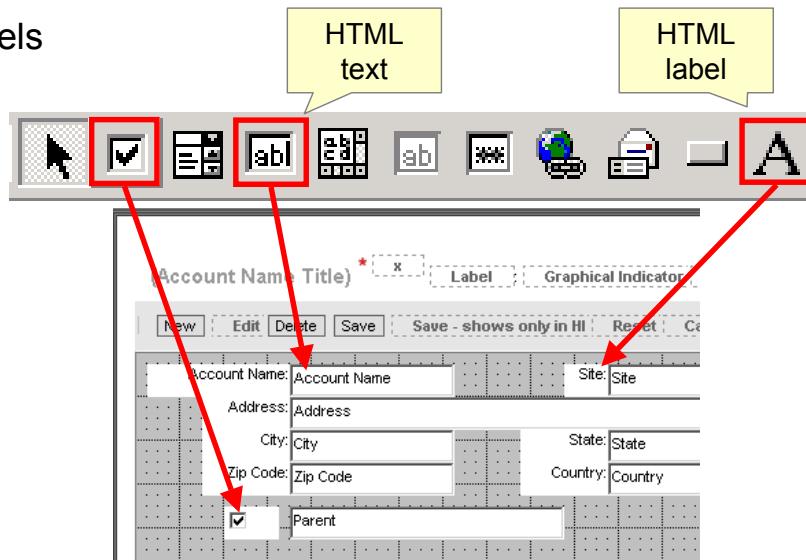


Copyright © 2007, Oracle. All rights reserved.

18 of 24

Web Controls Toolbar

- Supports drag-and-drop creation of controls
- Contains icons for:
 - ▶ Text controls
 - ▶ Check boxes
 - ▶ Labels



Copyright © 2007, Oracle. All rights reserved.

19 of 24

25

ORACLE®

Properties for Editing

- No Delete, No Insert, No Merge, No Update are set to restrict the behavior of the applet
 - ▶ Can also be set at the business component level
 - ▶ Most restrictive settings are always used

Business Component

Properties	
Business Component [Contact[Personal]]	
	Alphabetic Categorized
Name	Contact[Personal]
No Delete	FALSE
No Insert	FALSE
No Merge	FALSE
No Update	FALSE

Affects all applets referencing this BC, or...

Applet

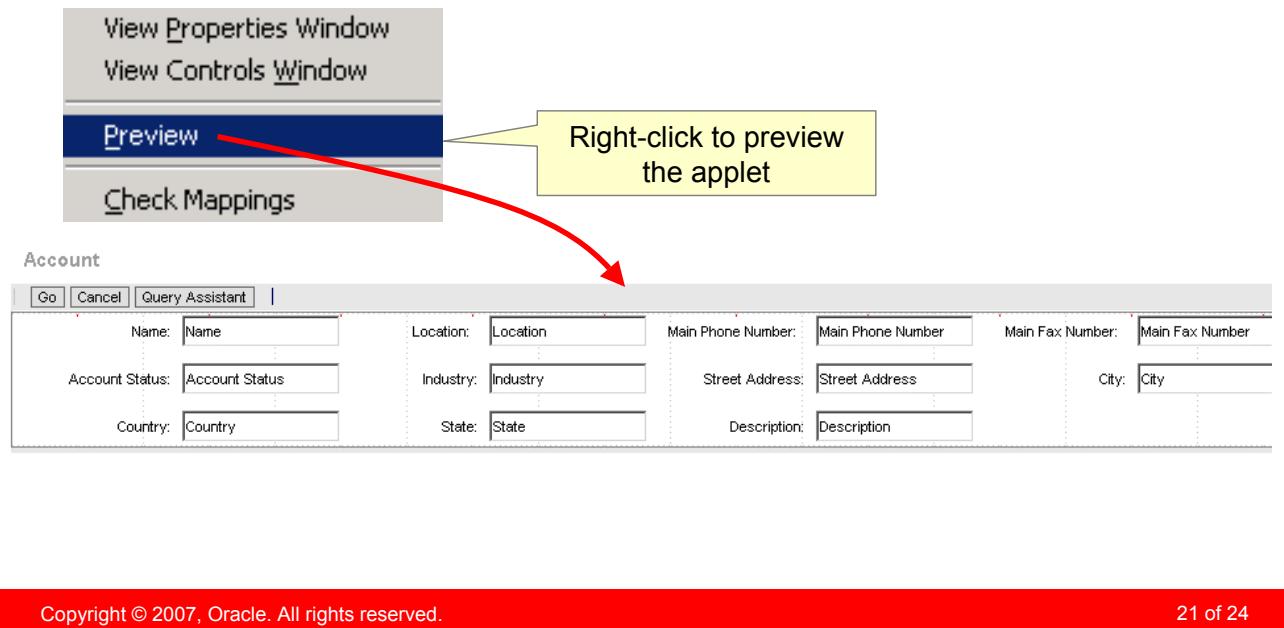
Properties	
Applet [Contact Form ReadOnly Applet]	
	Alphabetic Categorized
Name	Contact Form ReadOnly Applet
No Delete	TRUE
No Insert	TRUE
No Merge	TRUE
No Update	TRUE

...can be customized for each applet

ORACLE®

Preview the Applet

- Right-click and select Preview to see how the applet appears
 - ▶ When finished, right-click and deselect Preview to return to the Grid Layout Editor



Copyright © 2007, Oracle. All rights reserved.

21 of 24

25



Module Highlights

- List applets:
 - ▶ Display records simultaneously in a list
 - ▶ Contain fields for one record displayed in a single row
- Form applets display a single record and contain fields for the record in a grid or form
- Applet wizards ensure object definitions are created and steps are not omitted
- Web Layout Editor is used to add, remove, and reorder Web template items
- List column definitions specify field displayed and display name
- Symbolic strings ensure that the same name appears exactly the same way throughout the application



Lab

- In the lab you will:
 - ▶ Modify a form applet using the Grid Layout Editor
 - ▶ Create a new list applet using the wizard

25

ORACLE®

Siebel 8.0 Essentials

Module 26: UI Layer Configuration: Applications, Screens, and Views

26

26

Copyright © 2007, Oracle. All rights reserved.

Oracle Internal & Oracle Academy Use Only



Module Objectives

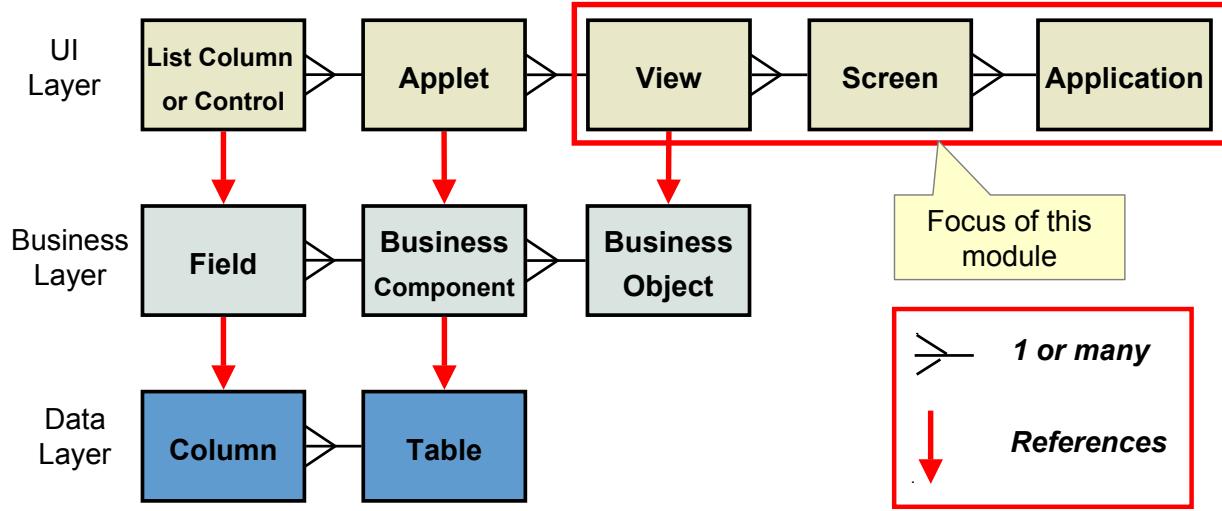
- After completing this module you should be able to:
 - ▶ Tailor applications
 - ▶ Configure how a view is accessed on a screen
 - ▶ Change the order of screen views for a screen
 - ▶ Create and administer a view
 - ▶ Associate a view with a template
- Why you need to know:
 - ▶ Enables you to modify the user interface

Logical User Interface

- Specifies the layout of the Siebel application
- Consists of:
 - ▶ Applications, screens, views, applets, controls/list columns
- This module focuses on views, screens, and applications

26

Siebel Application Architecture



Copyright © 2007, Oracle. All rights reserved.

3 of 24

Reference

[Configuring Siebel Business Applications: Configuring Screens and Views](#)
[Configuring Siebel Business Applications: Configuring Applications](#)

Applications

- Are collections of screens through which users can navigate
- Use a container page to display the Siebel Web page
- Specify the application-level menus

The screenshot shows a Microsoft Internet Explorer window displaying the Siebel Call Center interface. The title bar reads "Siebel Call Center - Microsoft Internet Explorer". The address bar shows the URL "http://localhost:8083/start.swe". The top menu bar includes File, Edit, View, Favorites, Tools, and Help. Below the menu is a toolbar with various icons. The main content area is titled "Accounts" and displays a list of accounts with columns for Account Name, Site, and Industries. An "Art.net" record is selected, showing detailed information in a modal dialog box at the bottom. The right side of the screen shows a sidebar with links like Home, Accounts, Contacts, Opportunities, Quotes, Sales Orders, and Service.

Account Name	Site	Industries
Art.net	Sterling, VA	prepackaged software
Broadband e2e	Las Angeles, CA	communications equipment
Chase Manhattan Bank	Manhattan, Ny	bank holding companies
Digital River, Inc.	San Francisco, Ca	computer related services
First Record, Inc	HQ	
Honeywell Intl (Allied Signal Aero)	Hq-Morristown, NJ	aircraft engines & engine parts
IBM Corporation	Poughkeepsie, NY	computer peripheral equipment
Imperial Tobacco	Theobald Road, London, tobacco & tobacco products	
Lexis-Nexis	Dayton, OH	information retrieval services
Nestle USA	Glendale, Ca	

Art.net

Account Name: * Art.net Site: Sterling, VA Account Team: JRUBIN Status: Active
 Address: 22110 Pacific Blvd Address Line 2:
 City: Sterling State: VA Main Phone #: (703) 205-3500 Account Type: Commercial
 Zip Code: 20166-6913 Country: USA Main Fax #: Territory:
 URL: http://www.artnet.com Industries: prepackaged softwa

Copyright © 2007, Oracle. All rights reserved.

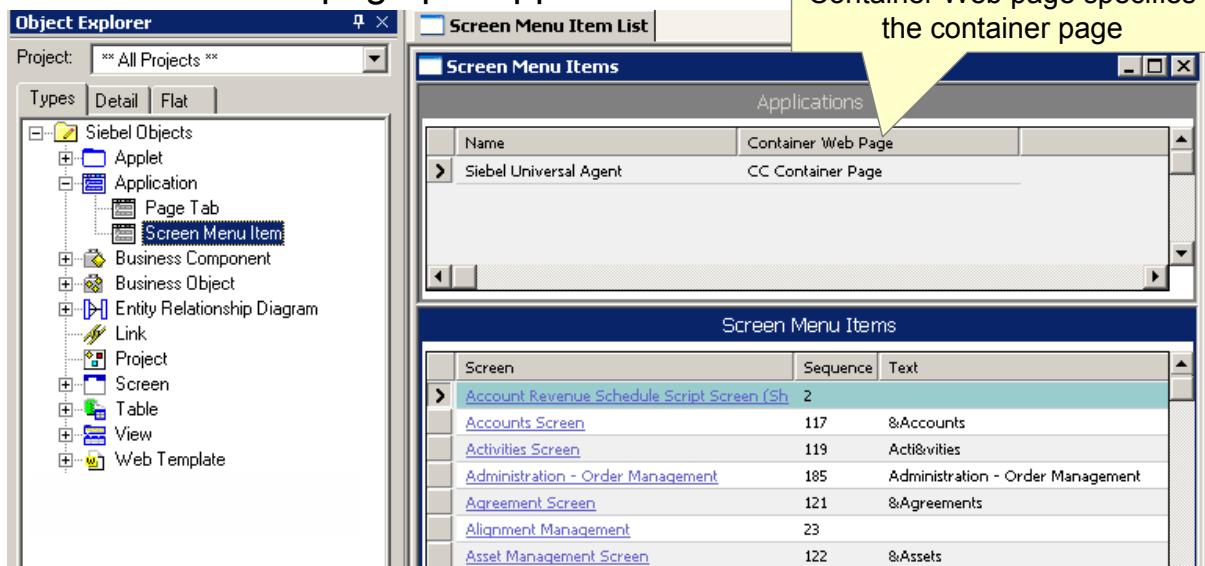
4 of 24

ORACLE®

The Container Page

- Is a Web page that displays the menus, toolbars, screen tabs, and views that appear in the application
- Is specified in the Application object definition
- One container page per application

Container Web page specifies the container page



Copyright © 2007, Oracle. All rights reserved.

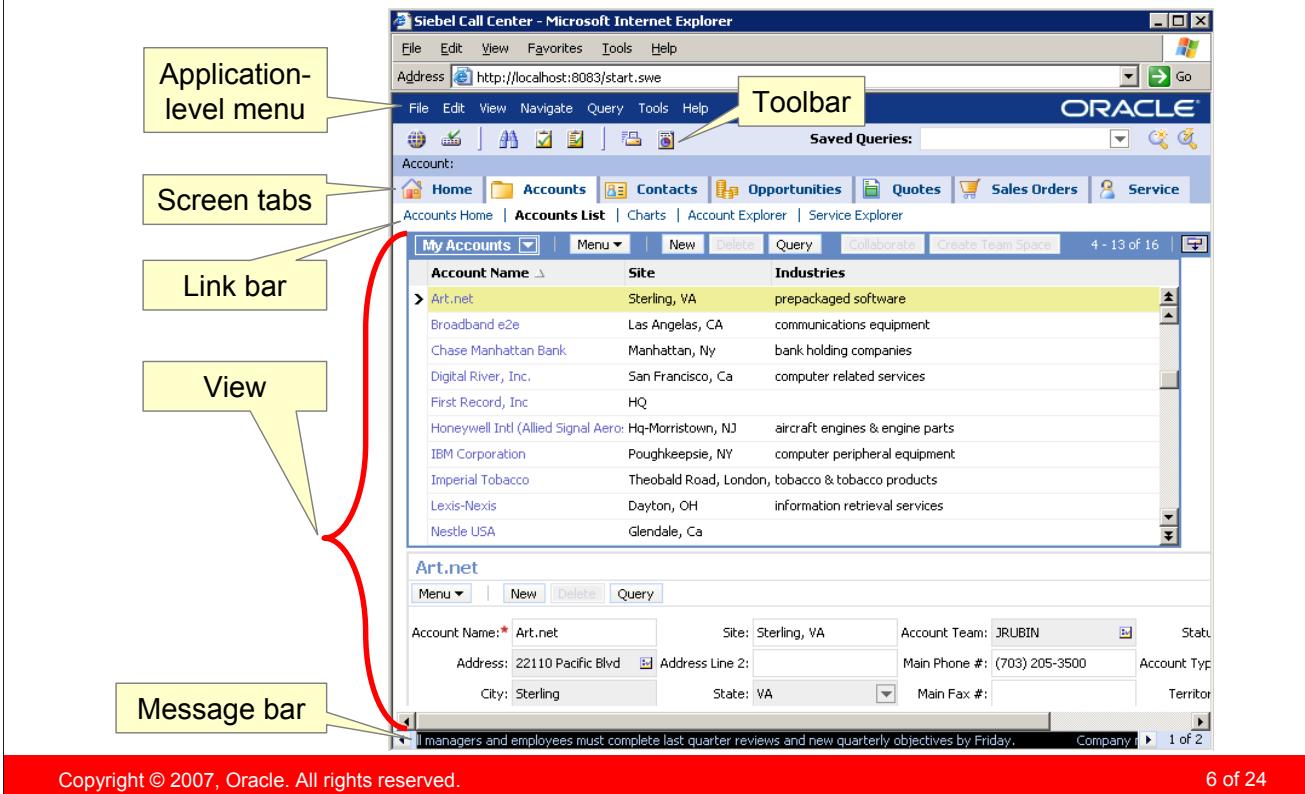
5 of 24

26

Container Page Example: Employee Application

ORACLE®

- Container page specifies application layout



Copyright © 2007, Oracle. All rights reserved.

6 of 24

Links

Links on the link bar and links used with business objects are two entirely different concepts. Links used with business objects are used to establish a relationship between parent and child business components in the business object.

Link bar links on the container page are used merely as a navigation device.

ORACLE®

Application-Level Menus

- Are specified in the Menu object type

The screenshot illustrates the Siebel Object Explorer interface, specifically focusing on the 'Menus' and 'Menu Items' sections.

Top Level: The 'File' menu bar is visible. A yellow callout box points to the 'File' menu item with the text: "Application uses Generic WEB menu...".

Object Explorer: The 'Object Explorer' window shows the project 'All Projects' and types 'Detail' and 'Flat'. It lists 'Siebel Objects' which include 'Applet', 'Application', 'Business Component', 'Business Object', 'Link', 'Menu' (selected), 'MenuItem' (selected), 'Pick List', 'Project', 'Screen', 'Symbolic String', 'Table', 'View', 'Web Page', and 'Web Template'.

Menus Section: The 'Menus' section displays a list of menus. A red box highlights the 'Generic WEB' menu item. A yellow callout box points to it with the text: "...which contains...".

Menu Items Section: The 'Menu Items' section displays a detailed list of menu items. The columns are 'Name', 'Caption', and 'Position'. The data is as follows:

Name	Caption	Position
File	&File	1
File - Connect	Connect	1.18
File - Create Bookmark	Create Bookmark...	1.2
File - Custom Print	Print...	1.11
File - Export Data	Export Data Map	1.14
File - Import Data	Import Data	1.13
File - Logout	Log Out	1.19

Yellow callouts provide additional details about the 'Caption' and 'Position' columns:

- "Caption specifies text displayed" points to the 'Caption' column.
- "Position specifies order of appearance" points to the 'Position' column.

Copyright © 2007, Oracle. All rights reserved.

7 of 24

26

ORACLE®

Screen Tabs

- Contain links to screens
- Are defined as page tabs
 - ▶ Child object of the application

The screenshot shows the Siebel Object Explorer on the left and the Page Tab List window on the right.

Object Explorer: Shows the project "All Projects" and various Siebel Objects under "Siebel Objects", including Application, Page Tab, Screen Menu Item, Business Component, Business Object, Entity Relationship Diagram, Link, Menu, Project, Screen, Table, View, and Web Template.

Page Tab List: Shows the "Page Tabs" section under "Applications". It displays a table with columns: Name, Menu, Container Web Page, and CC Container Page. A row for "Siebel Universal Agent" is selected, showing "Generic WEB" in the Menu column and "CC Container Page" in the Container Web Page column.

Page Tabs: Shows the "Page Tabs" configuration table. The table has columns: Screen, Text, and Sequence. The data is as follows:

Screen	Text	Sequence
Accounts Screen	Accounts	2
Contacts Screen	Contacts	3
Opportunities Screen	Opportunities	6
Quotes Screen	Quotes	7
Sales Order Screen	Sales Orders	9
Service Request Screen	Service	12
Web Call Center Home Screen	Home	1

Two callouts point to the "Text" and "Sequence" columns in the "Page Tabs" table:

- Specifies text that appears on the page bar
- Specifies default page tab order appearance in screenbar

Copyright © 2007, Oracle. All rights reserved.

8 of 24

ORACLE®

Screens

- Have a child object type called Screen View that specifies the views and categories that appear in the screen
 - ▶ Sequence property specifies where the view appears in the screen
 - ▶ Menu Text property defines the text that appears in Site Map
 - ▶ Type property defines the type of view within the screen
 - Four types: Aggregate category, aggregate view, detail category, and detail view

The screenshot shows the Siebel Object Explorer interface. On the left is the Object Explorer window with a tree view of Siebel Objects, including Application, Business Component, Business Object, Entity Relationship Diagram, Link, Menu, Project, Screen, and Screen View. The 'Screen View' node is selected. To the right is the 'Screen View List' window, which contains a table titled 'Screens' with one row: 'Name' (Accounts Screen). Below it is another table titled 'Screen Views' with one row: 'Name' (Account Screen Homepage View), 'View' (Account Screen Homepage View), 'Sequence' (1,001,000), 'Type' (Aggregate View), and 'Menu Text' (Accounts Home). Two callout boxes point to the 'Type' column in the 'Screen Views' table. The first box says 'Specifies where the view appears in the screen'. The second box says 'Specifies the type of view for this definition'.

Name
Accounts Screen

Name	View	Sequence	Type	Menu Text
Account Screen Homepage View	Account Screen Homepage View	1,001,000	Aggregate View	Accounts Home

Copyright © 2007, Oracle. All rights reserved.

9 of 24

26

ORACLE®

Type Property for Screen View

- An aggregate category is a container for a set of views accessible via a link in the linkbar
 - ▶ May also contain subcategories
- An aggregate view is a container for a set of views accessible via the visibility filter drop-down list, a link in the link bar, or the Home Page

The screenshot shows the Siebel application's Accounts List screen. At the top, there is a navigation bar with links for Home, Accounts, Contacts, Opportunities, Quotes, and Sales Orders. Below the navigation bar, a sub-menu for 'Accounts' is displayed, with 'Accounts List' highlighted and enclosed in a red box. A yellow callout points to this red box with the text 'Aggregate category'. In the main content area, there is a table with columns for Address Line 1, City, and State. The table has four rows of data. To the left of the table, there is a dropdown menu titled 'My Accounts' with several options: My Accounts, My Team's Accounts, All Accounts, and All Accounts Across Organizations. This dropdown is also enclosed in a red box. A yellow callout points to this red box with the text 'Aggregate views'. The data in the table is as follows:

	Address Line 1	City	State
3Com Distribution	10400 Fernwood Road	Bethesda	MD
UK	7074 N Clark St	Chicago	IL
US	1500 Northgate	London	
France	1000 6th Avenue	New York	NY
	38 quai du point du jour	Boulogne Billancourt	

ORACLE®

Type Property for Screen View continued

- A detail category is a container for a set of detail views
 - ▶ Each view is represented as a tab on the view tab bar
 - ▶ Used to provide an additional layer of navigation
- A detail view is a single view that displays data in a list or form that can be queried and edited

The screenshot shows two Siebel screens. The top screen is titled 'Accounts List' under 'Cambrian Ventures'. It displays account details for 'Cambrian Ventures' with fields for Account Name, Address, City, Zip Code, Site, and Address Line 2. The bottom screen is titled 'Enterprise Selling Process' under 'BU/SU Overview'. It displays a table with columns for Business / Service Unit, Culture, and Strategy Type, showing data for 'Cambrian West'.

Annotations explain the components:

- A yellow box labeled 'Aggregate category' points to the 'Accounts List' tab.
- A yellow box labeled 'Detail category...' points to the table on the 'Enterprise Selling Process' screen.
- A yellow box labeled '...displays detail view tabs' points to the tab bar on the 'Enterprise Selling Process' screen.
- A yellow box labeled 'Detail view' points to the table on the 'Enterprise Selling Process' screen.
- A red box highlights the 'Enterprise Selling Process' tab on the 'BU/SU Overview' screen.

Copyright © 2007, Oracle. All rights reserved.

11 of 24

26



Sequence Editor

- Assists in defining where view appears in the application
 - ▶ Updates the sequence property of the screen view object definition
 - ▶ Lists four screen view types in hierarchy

The screenshot shows the Siebel Sequence Editor interface. On the left is a tree view of screen categories like 'Screen - Accounts Screen' and 'Detail Category - ESP'. To the right is a table titled 'Screen Views' with columns for Name, View, Sequence, and Type. A specific row for 'Account Screen Homepage View' is highlighted with a red box around its 'Sequence' value, which is '1,001,000,000'. An 'Aggregate View' type is also indicated. A context menu is open at the top right of the table, listing options like 'Edit Screen View Sequence', 'Compare Objects', etc. A yellow callout points to the 'Edit Screen View Sequence' option with the text 'Right-click to invoke editor'. Another yellow callout points to the 'Sequence' column header with the text 'Right-click to edit sequence'. A separate 'View Properties Window' is shown on the right with keyboard shortcuts for moving views up and down.

Name	View	Sequence	Type
Account Screen Homepage View	Account Screen Homepage View	1,001,000,000	Aggregate View
Aggregate Category - Account List			
Aggregate View - Account List View			
Aggregate View - Complex Product Runtime Instance View (JS) - Account			
Aggregate View - Account List View			
Aggregate View - Manager's Account List View			
Aggregate View - All Account List View			
Aggregate View - All Accounts across Organizations			
Detail View - Account Detail View			
Detail View - Account Detail - Activities View			
Detail View - Account Asset Mgmt - Asset View			
Detail View - Account Attachment View			
Detail View - Account Detail - Contacts View			
Detail Category - ESP			
Detail View - ESP Business Service Unit Overview View			
Detail View - ESP Business Service Unit Offerings View			
Detail View - ESP Business Service Unit Contacts View			
Detail View - ESP Business Service Unit Partners View			
Detail View - ESP Sub Account Offerings View			

Copyright © 2007, Oracle. All rights reserved.

12 of 24



View

- Consists of one or more applets
- References a business object that defines the relationships between data in the view
- Is associated with a Web template that defines its layout

26

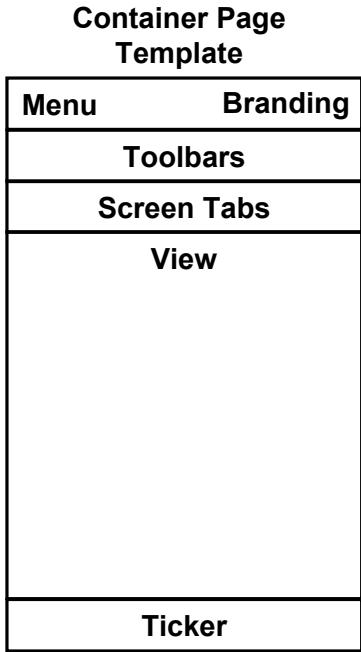
Copyright © 2007, Oracle. All rights reserved.

13 of 24

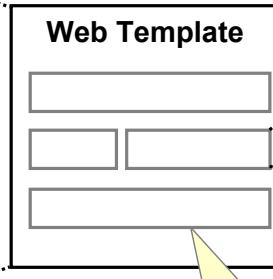
View and Templates

ORACLE®

- Views use templates to organize data layout

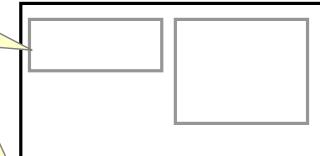


View Web Template

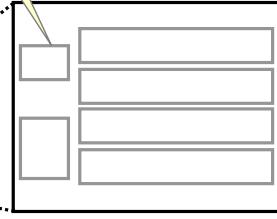


Control placeholders

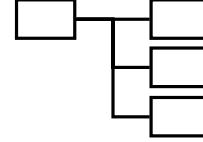
Form Applet Web Template



List Applet Web Template



Tree Applet Web Template



Applet placeholder

Copyright © 2007, Oracle. All rights reserved.

14 of 24



Selecting a View Template

- Find an existing view that has desired layout
- Examine the View Web Template object definition to determine the view template to use

The Account list view...

The screenshot illustrates the process of selecting a view template. On the left, the 'My Accounts' screen shows a list of accounts with columns for 'Account Name' and 'Site'. One account, 'Art.net', is highlighted. A yellow callout points from this account to the 'Views' window on the right. The 'Views' window lists 'Account List View' as the current view, which is defined by the 'View Detail (Parent with Pointer)' template. Below it, the 'View Web Templates' window shows a list of templates, with 'Base' selected and its 'Web Template' type set to 'View Detail (Parent with Pointer)'. Both the 'Views' and 'View Web Templates' windows have red boxes highlighting the 'Web Template' column.

Copyright © 2007, Oracle. All rights reserved.

15 of 24

26

Best Practice

Siebel best practice is to modify an existing template.

Steps to create and register a view template:

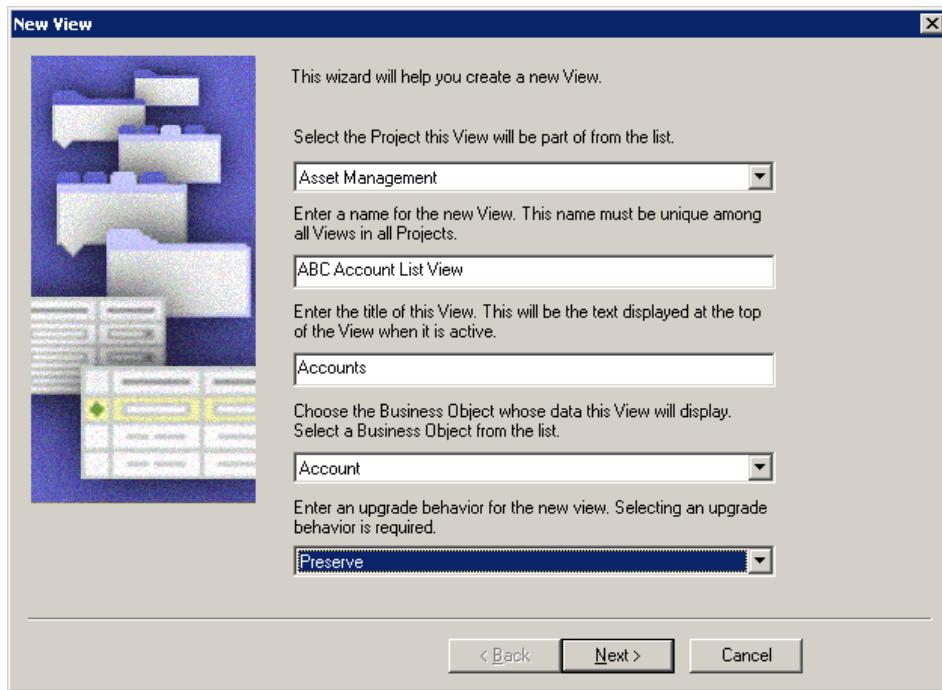
- Create a Web Template object.
- Create a Web Template file.
- Point to the view template file in the webtempl directory.
- Give a logical name to the template file so that it can be referenced by other objects.

Registering the template creates a logical reference in the repository. It tells the repository that the template file exists.

ORACLE®

View Wizard

- Use the View Wizard to create a new view
 - ▶ Select File > New Object, select View, and then click OK



Copyright © 2007, Oracle. All rights reserved.

16 of 24



View Wizard Inputs

- The View Wizard requires the following inputs:
 - ▶ Project that the new view will be part of
 - ▶ View name and display title
 - ▶ Business object that the view will reference
 - ▶ The upgrade behavior
 - Admin, Non-preserved, Preserved
 - ▶ The Web templates that will be used
 - ▶ The applets that make up the view
 - These applets will appear in the Web layout

26

ORACLE®

View Wizard Outputs

- The View Wizard:
 - ▶ Creates a view object
 - ▶ References view to a business object
 - ▶ Associates Web templates to the view
 - ▶ Invokes the Web Layout Editor

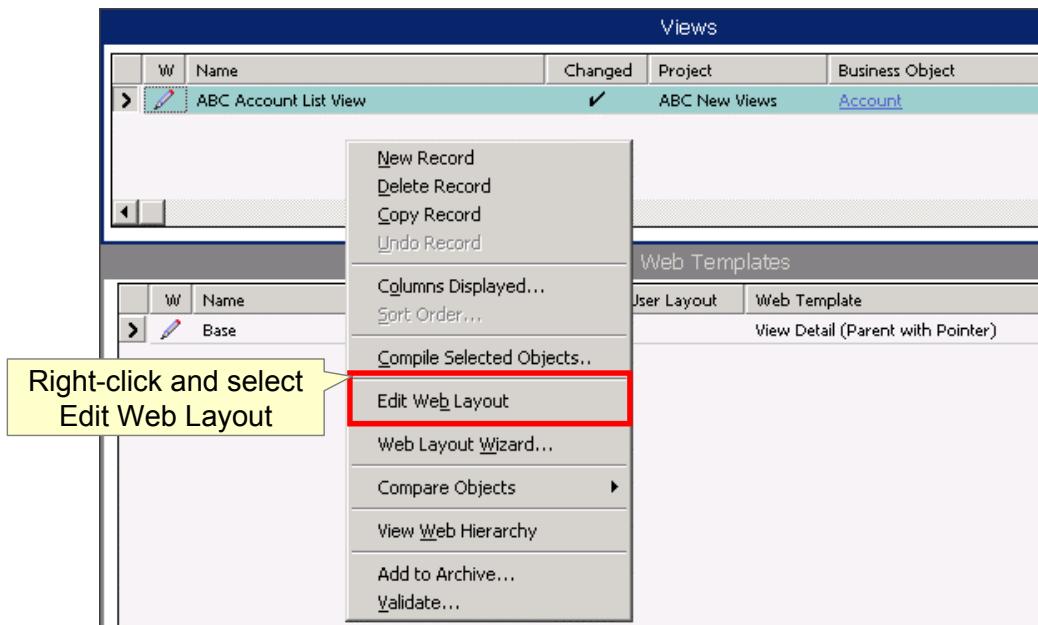
The screenshot shows the Siebel Object Explorer and the View List window. In the Object Explorer, under Siebel Objects > View, a new view named 'ABC Account List View' is being created. The View List window shows this view has been added to the 'Views' list under the 'Asset Management' project and is associated with the 'Account' business object. Annotations point to these areas with yellow callouts: 'Creates view' points to the new view entry in the View List, 'References business object' points to the 'Business Object' column, and 'Invokes Web Layout Editor' points to the main view area.

Copyright © 2007, Oracle. All rights reserved. 18 of 24

ORACLE®

Web Layout Editor

- Is launched by the View Wizard
- Can be launched by right-clicking a view definition in the OBLE



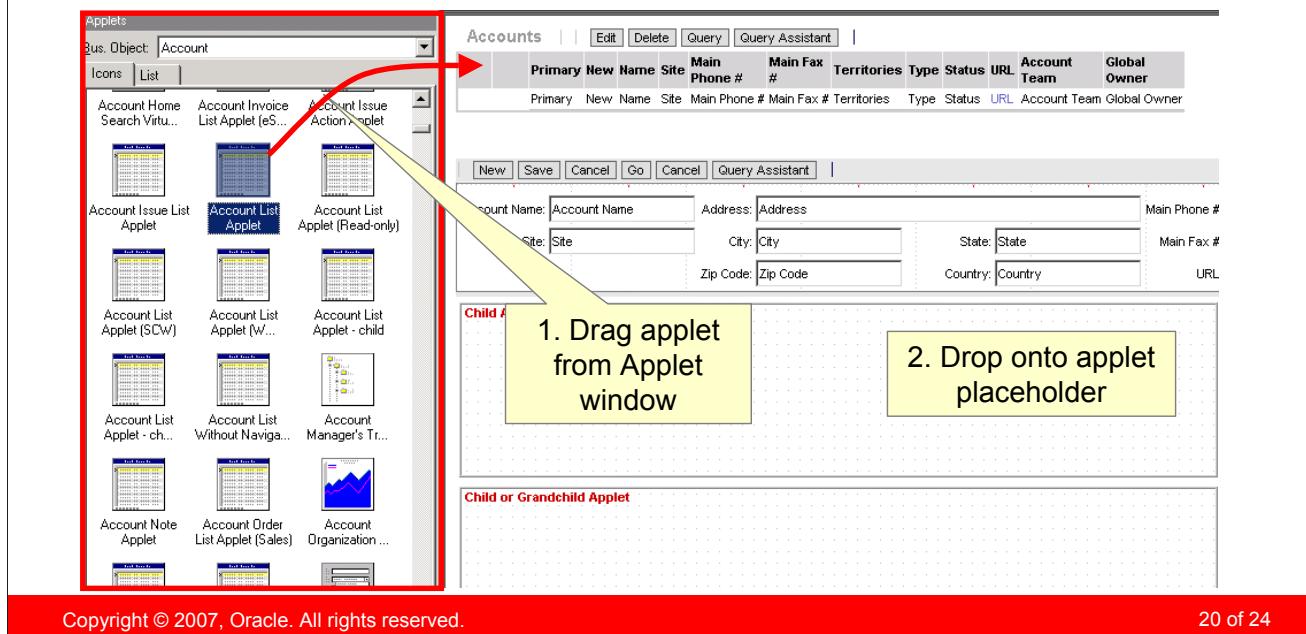
Copyright © 2007, Oracle. All rights reserved.

19 of 24

ORACLE®

Web Layout Editor Continued

- Displays the current layout of the view
- Applet window shows the applets available for display within the view
 - ▶ Constrained by the view's business object property



Copyright © 2007, Oracle. All rights reserved.

20 of 24

ORACLE®

View Web Template Items

- Are applets that have been bound to a view
 - ▶ Item identifier specifies where they are to appear
 - Used in the markup language tag that specifies corresponding control in a template

The screenshot shows the Siebel Object Explorer on the left and two windows titled 'View Web Template Item List' on the right. The Object Explorer lists various Siebel Objects like Applet, Application, Business Component, etc. The top 'View Web Template Item List' window shows a table with one row for 'Base'. The bottom window shows a table with two rows: 'ABC Asset Supply Plan List Applet' (Item Identifier 2) and 'Asset Mgmt - Asset Detail Applet' (Item Identifier 1). A yellow callout box points to the 'ABC Asset Supply Plan List Applet' row with the text: 'Binds a control to specific position on Web page'.

Name	Web Template
Base	View Detail

Name	Item Identifier	Applet Mode
ABC Asset Supply Plan List Applet	2	Edit List
Asset Mgmt - Asset Detail Applet	1	Edit

Copyright © 2007, Oracle. All rights reserved.

21 of 24

26

ORACLE®

Register and Assign View

- Views must be administered in the client to make them available to users
 - ▶ Navigate to Administration - Application > Views in the client application to register the view
 - Copy and paste the view name from the object definition in Tools to the view record
 - ▶ Navigate to Administration - Application > Responsibilities to assign the view to one or more responsibilities
 - Assign the view to developers for unit testing and to users for system testing and production

The screenshot shows the Siebel user interface with the following components:

- Top Navigation Bar:** Home, Accounts, Contacts, Opportunities, Quotes, Sales Orders, Service, Administ.
- Sub-navigation:** Branch Locator, Contact Us, Alerts Online, License Keys, Predefined Queries, Reports Server Administrator Profile, Responsibilities.
- Views Screen:** A table with columns View Name, Description, and Default Local Access. It shows a row for "ABC Account List View" with "My New View" as the description. A yellow callout box labeled "1. Register the view" points to this row.
- Responsibilities Screen:** A table with columns Responsibility, Description, Organization, and Web Access. It shows a row for "ABC Developer" with "Default Organization Yes" as the organization. A yellow callout box labeled "2. Assign the view" points to this row.
- Action Buttons:** New, Delete, Query, Menu, Clear Cache (highlighted with a red box).
- Help Text:** "3. Clear the cache" (highlighted with a yellow box) is shown above the "Clear Cache" button.
- Page Footer:** Copyright © 2007, Oracle. All rights reserved. 22 of 24.



Module Highlights

- Applications are groups of screens through which users navigate
- Screens are containers for views
 - ▶ Screen views come in four types: aggregate view, aggregate category, detail category, and detail view
- Views:
 - ▶ Consist of one or more applets
 - ▶ Reference a BO that defines the relationships between data in the view
 - ▶ Are associated with a Web template that defines their layout



Lab

- In the lab you will:
 - ▶ Create a view
 - ▶ Add the view to a screen using a specified view navigation mechanism
 - ▶ Administer the new view

ORACLE®

Siebel 8.0 Essentials

Module 27: UI Layer Configuration: Drilldowns

27

Copyright © 2007, Oracle. All rights reserved.



27

Oracle Internal & Oracle Academy Use One



Module Objectives

- After completing this module you should be able to:
 - ▶ Configure drilldown to a related view
 - ▶ Enable the thread bar
- Why you need to know:
 - ▶ Enables you to add drilldowns to assist users with navigation
 - ▶ Enables you to activate the thread bar to assist users



Navigation Using Drilldown

- You can configure list columns or controls for drilldown
 - ▶ When the user clicks the hyperlinked value, the application navigates to another view
- You can configure drilldown as static or dynamic
 - ▶ Static: Clicking the hyperlink always navigates to the same target view
 - ▶ Dynamic: Clicking the hyperlink navigates to a target view determined by values shown in the current view

Copyright © 2007, Oracle. All rights reserved.

3 of 16



27

Oracle Internal & Oracle Academy Use Only

Reference

Configuring Siebel Business Applications: Configuring Screens and Views
Configuring Siebel Business Applications: Configuring Applications

ORACLE®

Static Drilldown: Same Business Component

- Drill down to another view, keeping the business component (BC) context

The screenshot illustrates a static drilldown within the same business component (BC) context. It shows two views for a contact record named Dorothy Bates.

List View (Left):

- Header: My Contacts ▾ | Menu ▾ | New | Delete | Query
- Columns: Last Name, First Name, Mr/Ms, Work Phone #
- Data row: Bates, Dorothy, Mrs., Dorothy Bates
- A yellow callout box points to the "Dorothy Bates" entry with the text: "Contact record in Visible Contacts list view".

Detail View (Right):

- Header: Menu ▾ | New | Delete | Query
- Form fields:
 - Last Name: * Bates
 - First Name: * Dorothy
 - Middle Initial:
 - Mr/Ms: Mrs. (dropdown menu)
 - Job Title:
 - Work Phone #:
 - Work Fax #:
 - Mobile Phone #:
 - Home Phone #:
 - Email:
- Buttons: More Info | Activities | Notes | Opportunities | Service Requests | Agreements | Attachments
- Header: New Description Type Start ▾ End Due
- Information panel (bottom):
 - Screen: Contacts Screen
 - View: Contact Detail View
 - Business Object: Contact
 - Applets: Applet[0]: Contact Form Applet; Applet[1]: Contact Activity List Applet;
 - Business Components: BusComp[0]: Contact; BusComp[1]: Action;

A red arrow points from the "Dorothy Bates" entry in the list view to the same entry in the detail view. A yellow callout box points to the "Dorothy Bates" entry in the detail view with the text: "Same contact record in Contact Detail view".

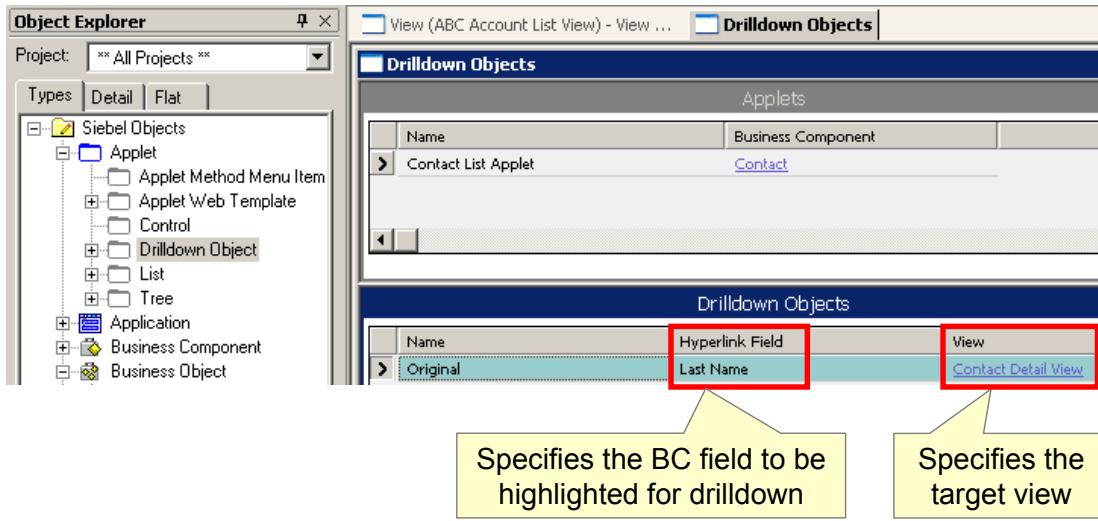
Copyright © 2007, Oracle. All rights reserved.

4 of 16

ORACLE®

Configuring Drilldown Within Same Business Component

- Create a new drilldown object
 - ▶ Set the Hyperlink Field and View properties to enable static drilldown
 - ▶ Drilldown object is child of an applet object



Copyright © 2007, Oracle. All rights reserved.

5 of 16

27

Oracle Internal & Oracle Academy Use Only

ORACLE®

Static Drilldown: Different Business Component

- Drill down to another view, changing the business component context

The screenshot illustrates a static drilldown between two Siebel screens: 'All Contacts Across Organizations' and 'AEP Communications'.

Contact Screen: Shows a list of contacts. A contact named 'Shashi' is selected, and the account associated with her, 'AEP Communications', is highlighted with a red box. A callout bubble points to this account field with the text: "The account field of the contact record in the Contacts list view".

Account Screen: A modal window titled 'AEP Communications' displays account details. A callout bubble points to this screen with the text: "Related account record in the Account Detail - Contacts view".

Details: The 'AEP Communications' account screen shows the following fields:

Account Name: * AEP Communications	Site: Columbus, OH	Account Team: MMAY
Address: 1 Riverside Plaza	Address Line 2:	Main Phone #: (800) 477-5
City: Columbus	State: OH	Main Fax #:
Zip Code: 43215-2373	Country: USA	URL: www.aep.c

Bottom Status Bar:

Copyright © 2007, Oracle. All rights reserved. 6 of 16

Configuring Drilldown to Different Business Component

- Create a new drilldown object for the applet
 - ▶ Set Hyperlink field and View properties as before
 - ▶ Set Business Component, Source Field, and Destination Field properties to enable drilldown to a view based on a different parent record

The screenshot shows the Siebel Object Explorer on the left and the Drilldown Objects screen on the right.

Object Explorer:

- Project: ** All Projects **
- Types: Siebel Objects, Applet, Application, Business Component, Business Object
- Applet node expanded, showing sub-items: Applet Method Menu Item, Applet Web Template, Control, Drilldown Object, List, Tree.

Drilldown Objects:

Applets:

Name	Business Component
Contact List Applet	Contact

Drilldown Objects:

Name	Hyperlink Field	View	Source Field	Business Component	Destination Field
Account	Account	Account Detail - Contacts View	Account Id	Account	

Annotations on the Drilldown Objects table:

- A callout points to the "Business Component" column with the text "BC in target view".
- A callout points to the "Source Field" column with the text "Foreign Key field in record pointing to target BC".
- A callout points to the "Destination Field" column with the text "Primary Key field in target BC (default is ROW_ID)".

Copyright © 2007, Oracle. All rights reserved. 7 of 16

Reference

Configuring Siebel Business Applications: Configuring Screens and Views:
About Drilldowns
Siebel Security Guide: Configuring Drilldown Visibility

ORACLE®

Dynamic Drilldown

- Enables drilldown to multiple views from the same hyperlink field, depending on value of a field in the active record of the list

For business contacts, go to the Contact Detail view

For personal contacts, go to the Personal Contact List view

Contact Detail View (Glen Abboline):

Last Name	First Name	Mr/Ms
Abboline	Glen	Mr.

Personal Contact List View (Max Adams):

Last Name	First Name	Mr/Ms	Work Phone #
Adams	Max	Mr.	(650) 295-5000

Copyright © 2007, Oracle. All rights reserved.

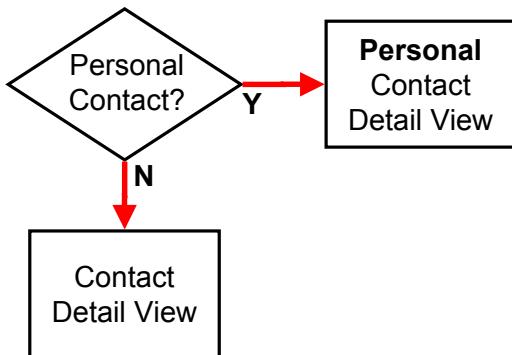
8 of 16

Note

The single value field that controls which view the dynamic drilldown navigates the user to is configured when the business component is created. In the example here, the business component is "Contact" and the single valued field is "Personal Contact." For Adams, "Personal Contact" = Y (yes), while for Abboline "Personal Contact" = N (no).

Matching Conditions

- Identify one or more matching conditions to trigger drilling down to a specific target view
 - ▶ For example:



- Determine the order to check the matching conditions
 - ▶ Conditions can involve different fields
 - ▶ Multiple matches could occur for a given record

ORACLE®

Configuring Dynamic Drilldown

- Create the drilldown objects for each view
 - ▶ Multiple drilldown objects with same hyperlink but different views
- Set sequence number to determine where to look for dynamic drilldown destinations
 - ▶ The drilldown object with the lowest sequence number is checked for dynamic drilldown destination child object definitions

The Original and Personal drilldown objects have the same Hyperlink field but different views

The Original drilldown object has the lower sequence number, so it is checked for dynamic drilldown destinations

W	Name	Hyperlink Field	View	Sequence
	Account	Account	Account Detail - Contacts View	2
	Original	Last Name	Contact Detail View	1
	Personal	Last Name	Personal Contact List View	3

W	Name	Field	Value	Destination Drilldown Object	Sequence
	Contact	Personal Contact	N	Original	2
	Personal	Personal Contact	Y	Personal	1

Copyright © 2007, Oracle. All rights reserved.

10 of 16

ORACLE®

Configuring Dynamic Drilldown Continued

- Create a dynamic drilldown destination definition for each condition
 - ▶ Each condition specifies a destination drilldown object to use
 - ▶ Recommended practice: Create a default dynamic drilldown destination to specify a desired view if there are no matches
 - If no default is configured, navigates to the view specified by the parent drilldown object

Dynamic Drilldown Destinations

W	Name	Hyperlink Field	View
>	Account	Account	Account Detail - Contacts View
	Original	Last Name	Contact Detail View
	Personal	Last Name	Personal Contact List View

Dynamic Drilldown Destinations

W	Name	Field	Value	Destination Drilldown Object	Sequence
>	Contact	Personal Contact	N	Original	2
	Personal	Personal Contact	Y	Personal	1

Copyright © 2007, Oracle. All rights reserved.

11 of 16

ORACLE®

Thread Bar

- Tracks previous business object (view) and active record
- Updates whenever the user navigates to a different business object
- Provides hyperlinks to previous views
 - ▶ Allows user to backtrack easily

Copyright © 2007, Oracle. All rights reserved.

12 of 16

Note

The thread bar allows the user to navigate to a specific view or record previously visited, by selecting the item of interest in the thread bar. Drilldowns implement the user's business logic; they are always parts of business processes. When a part of a business process is completed, the user will usually return to an earlier view or record, and the thread bar facilitates this navigation.

ORACLE®

Thread Properties

- Specifies the text that appears in the thread bar

The screenshot shows the Siebel interface with the following elements:

- Toolbar:** Shows icons for globe, keyboard, search, etc., followed by "Contact:Aamot > Account:".
- Thread title:** Labeled in a yellow box below the toolbar.
- Thread field:** Labeled in a yellow box next to the Thread title.
- Properties Dialog:** Titled "Properties" for "View [Contact List View]". It has tabs for "Alphabetic" and "Categorized".
- Basic Tab Data:**

Add To History	TRUE
Admin Mode Flag	FALSE
Business Object	Contact
Comments	
Inactive	FALSE
Module	
Name	Contact List View
Object Language Locked	
Object Locked	FALSE
Object Locked By Name	
Object Locked Date	
Thread Applet	Contact List Applet
Thread Field	Last Name
Thread Title	Contact
- Annotations:**
 - A callout points from the "Thread Title" property in the dialog to the "Thread Title" label in the toolbar, with the text: "Specifies the record to ‘remember’ in the hyperlink".
 - A callout points from the "Thread Title" label in the toolbar to the "Thread Title" property in the dialog, with the text: "Thread Title property specifies prior business object in hyperlink; if not entered, application uses Title property".

Copyright © 2007, Oracle. All rights reserved.

13 of 16

27



Module Highlights

- List columns or controls can be configured for drilldown
- Drilldowns can be static or dynamic
 - ▶ Static drilldowns always navigate the user to the same target view
 - ▶ Dynamic drilldowns navigate the user to a target view determined by the values shown in the current view
 - ▶ Static drilldowns can be configured to drill down to the same business component or to a different business component
- Thread bar allows user to easily backtrack
 - ▶ Configured in Tools



Lab

- In the lab you will:
 - ▶ Create and configure dynamic drilldowns
 - ▶ Enable and configure thread support

Copyright © 2007, Oracle. All rights reserved.

15 of 16

27

Oracle Internal & Oracle Academy Use Only

ORACLE®

Siebel 8.0 Essentials

Module 28: Business Layer Configuration: Joins

28

Copyright © 2007, Oracle. All rights reserved.

Oracle Internal & Oracle Academy
28 Use Only



Module Objectives

- After completing this module you should be able to:
 - ▶ Create a join that brings data from a standard table into a standard business component
 - ▶ Create a join that brings data from a party table into a standard business component
 - ▶ Create a join that brings data from a party table into another party business component
- Why you need to know:
 - ▶ Enables you to configure your company's business logic

Copyright © 2007, Oracle. All rights reserved.

2 of 24

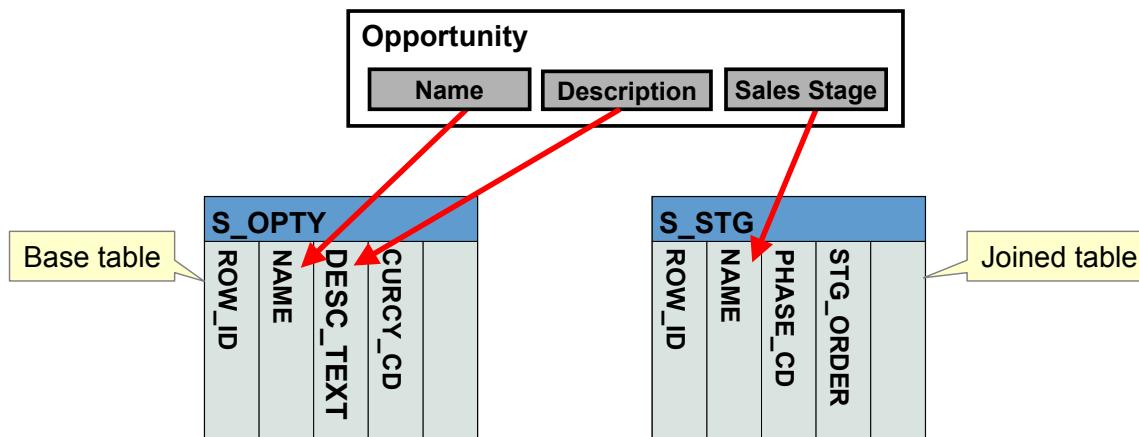
Reference

Configuring Siebel Business Applications: Configuring Joins

ORACLE®

Joined Tables Review

- Business components can include data from related joined tables
 - ▶ Brings in data as needed to meet the business component's data requirements
 - For display in applets
 - For use in processing by the business component



Copyright © 2007, Oracle. All rights reserved.

3 of 24

ORACLE®

Three Definitions Are Used to Implement a Join Review

- Join definition specifies the joined table
- Join specification specifies the FK and PK used to relate the base and joined tables
- Single-value field (SVF) references the join object definition

The screenshot shows the Siebel Business Components interface with three main windows:

- Join Specifications (Top Window):** Shows a join named "S_STG" with an alias "Sales Stage". A yellow callout labeled "1. Join definition" points to this row.
- Join Specifications (Middle Window):** Shows a join specification for "Sales Stage Id" mapping to "ROW_ID" and "Sales Stage Id". A yellow callout labeled "2. Join specification" points to the "Name" column of the first row.
- Single Value Fields (Bottom Window):** Shows a single-value field named "Sales Stage" with a join to "Sales Stage" and a column "NAME". A yellow callout labeled "3. Single-value field" points to the "Column" column.

A red arrow points from the "1. Join definition" callout to the "Sales Stage" alias in the top window. Another red arrow points from the "2. Join specification" callout to the "Name" column in the middle window. A third red arrow points from the "3. Single-value field" callout to the "Column" column in the bottom window.

Copyright © 2007, Oracle. All rights reserved.

4 of 24

ORACLE®

Join Example

- Will bring Sales Stage data from S_STG into the Opportunity BC
 - ▶ S_OPTY is the base table, S_STG is the joined table

The screenshot shows the Siebel Opportunities List screen. At the top, there's a navigation bar with links for Home, Accounts, Contacts, Opportunities, and Quotas. Below that is a toolbar with buttons for My Opportunities, Menu, New, Delete, and Query. The main area is a grid titled 'Opportunities List' with columns: Opportunity Name, Account, and Sales Stage. Three rows of data are shown:

Opportunity Name	Account	Sales Stage
200 PCS CS Laptop units	AT&T	J1 - Prospecting
200 PCS Puma Laptop EB units	Imperial Tobacco	J2 - Qualification
200 PCS Puma Laptop EB units	Broadband e2e	

Annotations with arrows point from the 'Opportunity Name' column to the text 'From S_OPTY' and from the 'Sales Stage' column to the text 'From S_STG'. The first two rows have their entire row boxes highlighted with red borders, while the third row does not.

Copyright © 2007, Oracle. All rights reserved.

5 of 24

0/6



Building a Join

1. Locate the Data Source
2. Examine Existing Joins
3. Diagram the Join
4. Create the Join Definition
5. Create the Join Specification
6. Map the Single Value Field(s)

1. Locate the Data Source

- Identify potential data sources
 - ▶ Peruse application for data that you seek
 - The required data could exist in another business component
- Once you find the data in the UI, identify the specific table and column where the data resides
 - ▶ Trace down from the UI element to the table column
 - For assistance, use the job aid from the Business Component lab

Identify table and column that contains data you seek

Layer	Step	Object You're looking For	Source Where Information About The Object Can Be Found
User Interface Objects 	1	Display Name	Look for the text label: • In a list applet, the column heading above the field in question • In a form applet, the control label next to the field
	2	Applet Name	In UI: Help > About View
Business Objects 	3	Business Component	In UI: Help > About View
	4	Business Component Field	a. In Tools: <applet name> List :: List List Column. b. Query Display Name for <display name> (from step 1). c. Value you need is the Field property.
Data Objects 	5	Table Name Join Table? (if yes: <input checked="" type="checkbox"/>) 	In Tools: Business Component :: <business component name> Field :: <field name> IF value under Join = empty Enter base table name ELSE Enter value under Join Note: if query fails, value in "Field" is calculated. Try this: query Calculated Value for "<field name>"
	6	Table Column Name	In Tools: Business Component :: <business component name> Field :: <field name>. Value under Column

2. Examine Existing Joins

- Determine if a join already exists
 - ▶ Peruse existing joins for joins that might be candidates
 - ▶ Oracle has already provided many commonly-used joins
- If a join exists that meets data display requirements, use it
- Otherwise, create a new join

The screenshot shows the Siebel Object Explorer and the Join List view side-by-side.

Object Explorer:

- Project: ** All Projects **
- Types: Siebel Objects, Applet, Application, Business Component, Business Object, Entity Relationship Diagram, Link, Menu, Project, Screen.
- Selected: Business Component > Join

Join List:

- View: ABC Account List View - View ...
- Tab: Join List
- Section: Joins
- Table: Joins
- Columns: Table, Alias, Outer Join Flag
- Data:

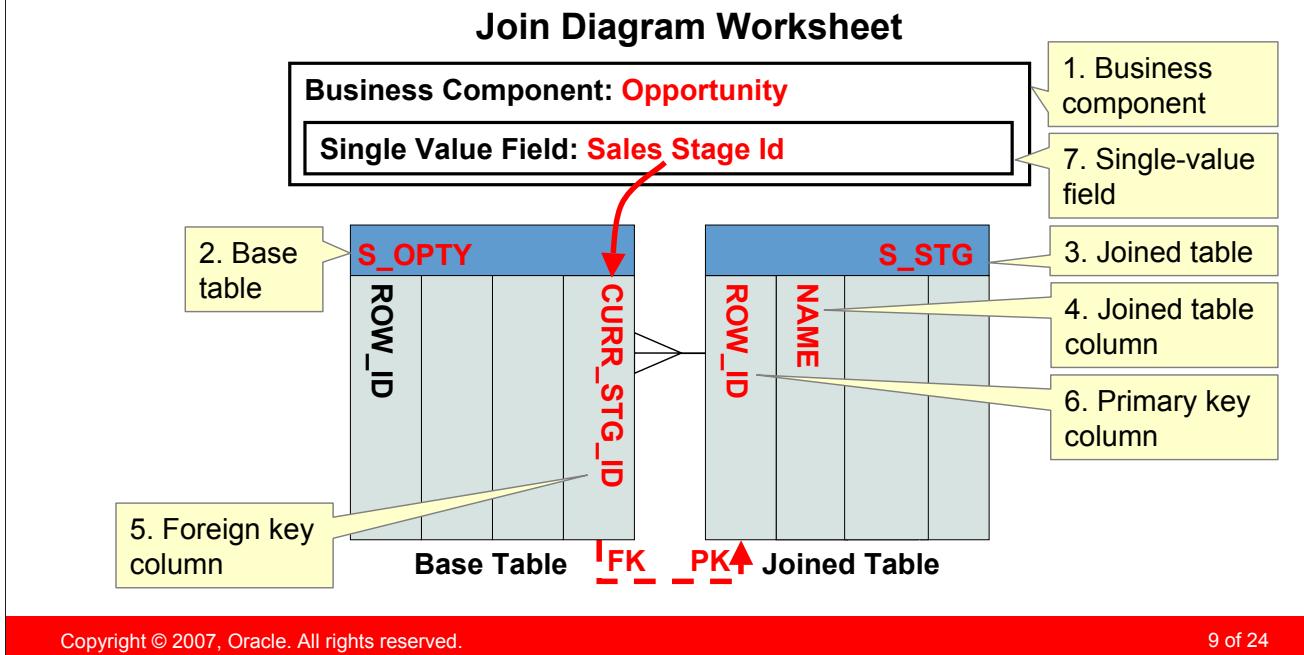
Table	Alias	Outer Join Flag
S_USER	CREATED_BY	✓
S_SRC	Campaign Name	✓
S_SYS_KEYMAP	Inbound Opportunity	✓
S_DMND_CRTN_PRG	Offer Name	✓
S_SYS_KEYMAP	Outbound Opportunity	✓
S_OPTY	Parent Opportunity	✓
S_BLU	Partner	✓

Copyright © 2007, Oracle. All rights reserved.

8 of 24

3. Diagram the Join

- The Join Diagram template simplifies join creation
 - ▶ Gather information first, then create the join
- Indicate each of the numbered items



Academy Only

28

Procedure Quick Start

1. Enter business component name.
2. Enter base table name: value of Table property for BC.
3. Enter joined table name.
4. Enter joined table column name that contains data to be brought into this BC.
5. Enter foreign key column name. Select Table :: [base table] | Column. Query Foreign Key Table property for [joined table name]. For example, Table :: S_OPTY | Column. Foreign Key Table = S_STG.
6. Enter primary key column name. Select Table :: [joined table] | Column. Query Primary Key property for TRUE. For example, Table :: S_STG | Column. Primary Key = TRUE.
7. Enter SVF name. Select [business component] :: Single Value Field. Query Column property for [FK column]. For example, Opportunity :: Single Value Field. Column = CURR_STG_ID.

4. Create the Join Definition

- Table property specifies joined table
- Alias property is the name of join definition
 - ▶ Alias always defaults to the joined table's name
 - Displayed in ALL CAPS
 - ▶ Additional aliases for self-joins or multiple joins to the same table are specified by the developer
 - Recommended practice is to use Initial Caps

Object Explorer

Project: ** All Projects **

Types Detail Flat

- Siebel Objects
- Applet
- Application
- Business Component
 - BusComp View Mode
 - Business Component User Prop
 - Field
 - Join**
 - Multi Value Field
 - Multi Value Link
 - Single Value Field
- Business Object
- Entity Relationship Diagram

View (ABC Account List View) - View ... Join List

Joins

Name	Table	Alias	Outer Join Flag
Opportunity	S_USER	CREATED_BY	✓
	S_USER	S_USER	✓
	S_USER	Updated By Login	✓
	S_TEAMSPACESEARCH	S_TEAMSPACESEARCH	✓
	S_SYS_KEYMAP	Outbound Opportunity	✓
	S_STG	Inbound Opportunity	✓
	S_STG	Sales Stage	✓

Copyright © 2007, Oracle. All rights reserved.

10 of 24

5. Create the Join Specification

- Specify foreign key field of the base business component in the Source Field property
- Specify primary key column of the joined table in the Destination Column property

The screenshot shows the Siebel Project Explorer interface with the 'Join' tab selected. In the center, the 'Join Specifications' window displays a 'Joins' section for 'Sales Stage' with alias 'S_STG'. Below it is a 'Join Specifications' table:

Name	Destination Column	Source Field
Sales Stage Id	ROW_ID	Sales Stage Id

Red boxes highlight the 'ROW_ID' columns in both the destination and source fields. To the right, a conceptual diagram illustrates the join between two tables:

```

graph LR
    subgraph Base_Table [Base Table]
        direction LR
        S_OPTY[ROW_ID] --- CURR_STG_ID[CURR_STG_ID]
    end
    subgraph Joined_Table [Joined Table]
        direction LR
        S_STG[ROW_ID] --- NAME[NAME]
    end
    CURR_STG_ID -- FK --> S_STG[ROW_ID]
    CURR_STG_ID -- PK --> S_OPTY[ROW_ID]

```

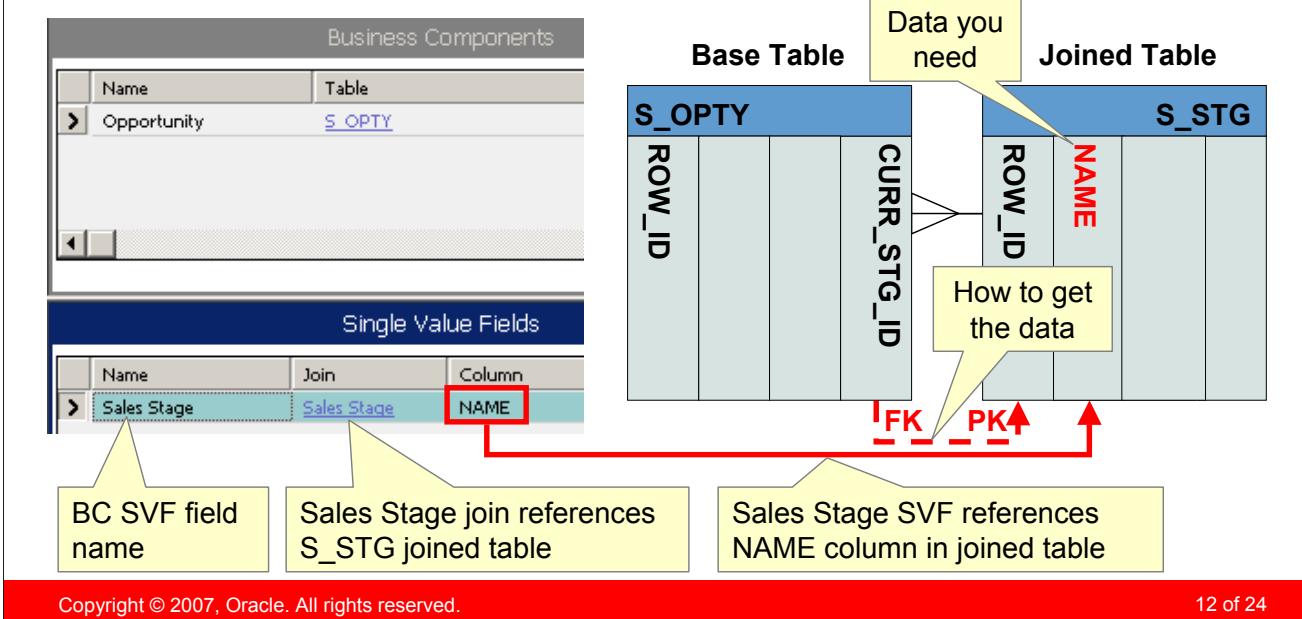
The diagram shows a one-to-many relationship where the primary key 'ROW_ID' from the 'S_OPTY' table (labeled 'PK') joins the foreign key 'CURR_STG_ID' in the 'S_STG' table (labeled 'FK').

Copyright © 2007, Oracle. All rights reserved.

11 of 24

6. Map the Single Value Field(s)

- Specify business component field name, the joined table, and column name of the joined table
 - ▶ SVF specifies the data you seek
 - ▶ The join determines how you access that data



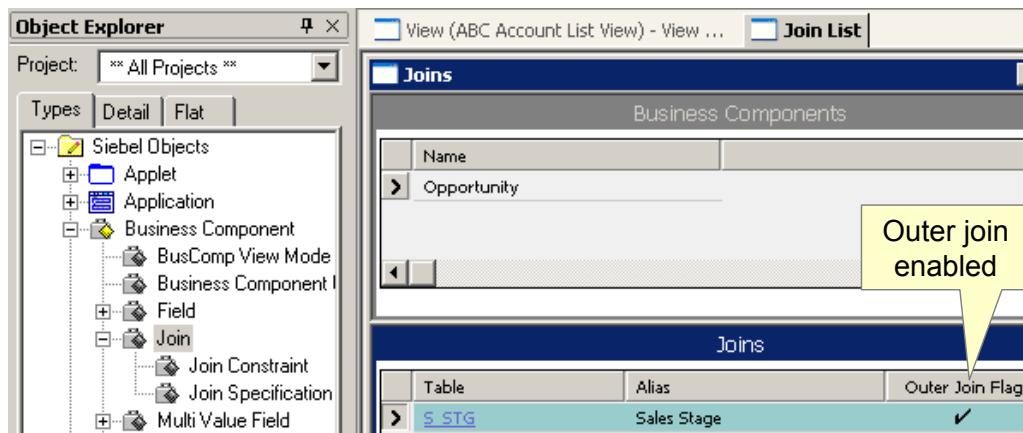
Copyright © 2007, Oracle. All rights reserved.

12 of 24

ORACLE®

Outer Join Flag

- An option in the join definition
- Set Outer Join Flag to TRUE to return all records from the base table, even when there is no related row in the joined table
- Outer joins will affect performance
 - ▶ Do not enable outer join if there is always a related row in the joined table



Copyright © 2007, Oracle. All rights reserved.

13 of 24

Only

28

Oracle Internal & Oracle Academy Only

ORACLE®

Outer Join Flag Continued

- When Outer Join Flag is TRUE (checked), a *left outer join* is used
 - ▶ All records in source table are displayed
 - Even if there is no related record in the target table
 - ▶ Records with no related record in target table display target values as null

S_OPTY			
ROW_ID	NAME	DESC_TEXT	CURR_STG_ID
001	200 PCS CS Laptop units	AT&T	105
002	200 PCS Puma Laptop EB	Imperial Tobacco	106
003	200 PCS Puma Laptop EB	Broadband e2e	NULL

All records displayed from source

FK!

S_STG		
ROW_ID	NAME	PHASE_CD
101	03 - Closing	9/7/2001
102	05 - Won	9/18/2001
103	07 - Selected	2/20/2004
104	06 - Short List	9/18/2001
105	01 - Prospecting	9/18/2001
106	02 - Qualification	9/7/2001

No corresponding record for Broadband e2e

APK

0/3

ORACLE®

Joins and Party Business Data

Joins and Party Business Components

Bringing Party Data into a Standard Business Component

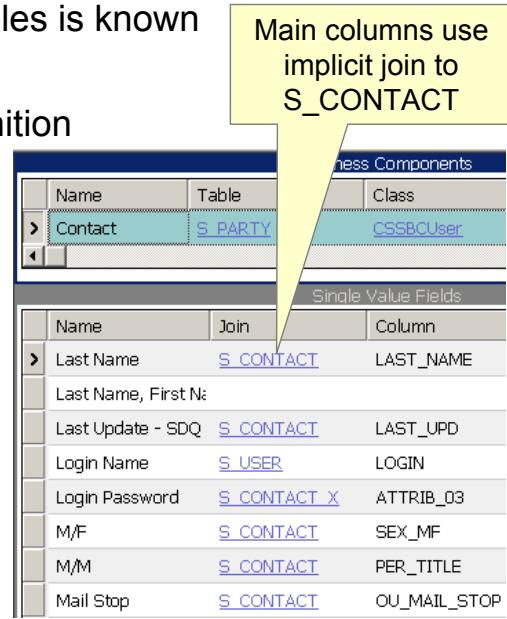
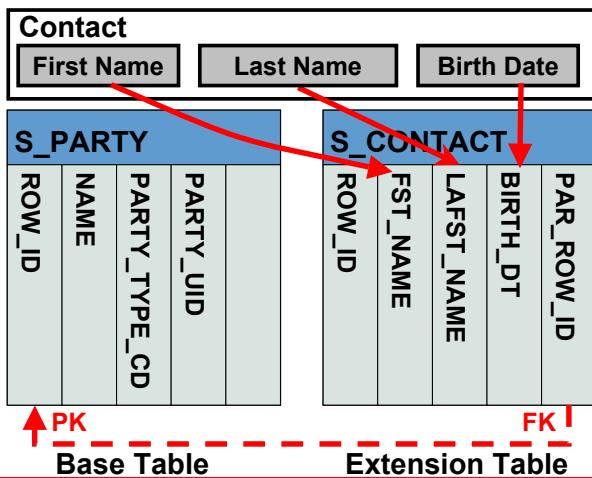
Bringing Party Data into a Party Business Component

Copyright © 2007, Oracle. All rights reserved.

15 of 24

Joins and Party Business Components Review

- Party Business Components have S_PARTY as the base table, but store main data in S_PARTY extension tables
- An implicit join is used to reference fields from extension tables
 - ▶ Used when relationship between tables is known
 - ▶ Appears in Join property on SFV
 - ▶ Does not appear as Join object definition



Copyright © 2007, Oracle. All rights reserved.

16 of 24

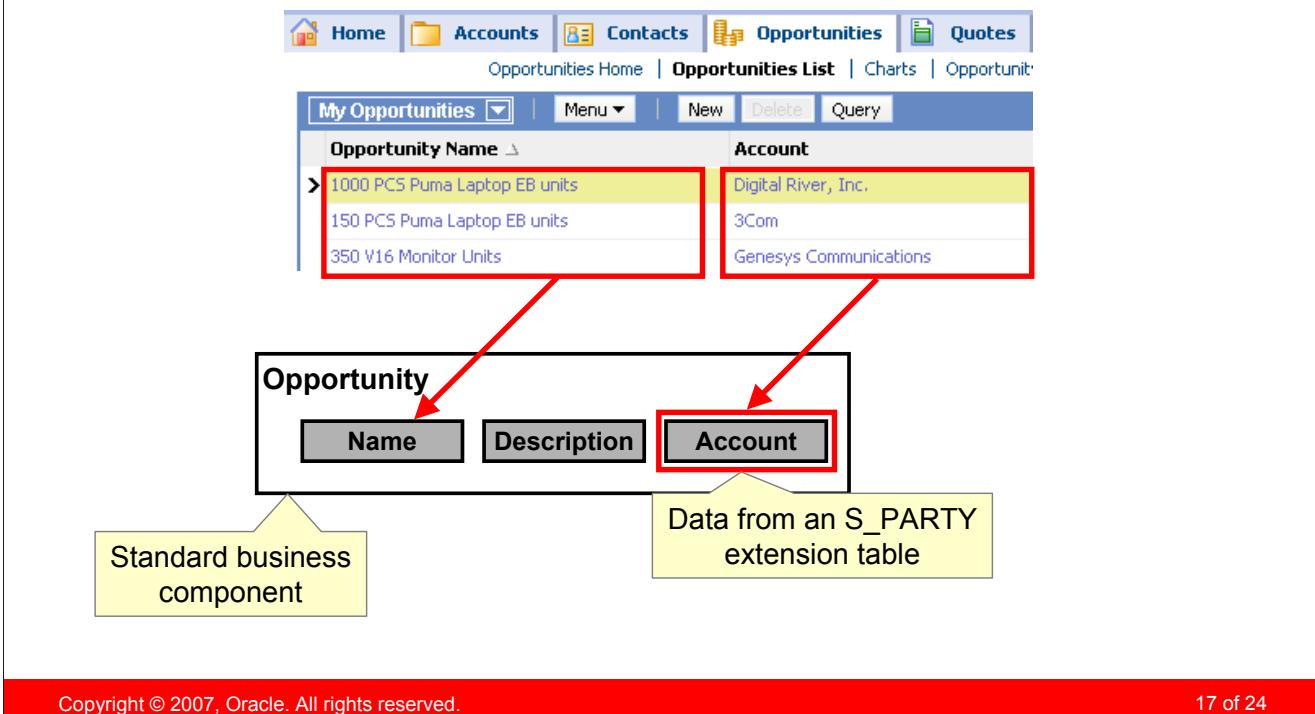
S_PARTY Extension The following tables are some of the extension tables for S_PARTY:

Tables

- S_BU
- S_CONTACT
- S_ORG_EXT
- S_POSTN
- S_USER

Bringing Party Data into a Standard Business Component

- Example: Bringing account data into the Opportunity business component for display in an Opportunity applet



Copyright © 2007, Oracle. All rights reserved.

17 of 24

Three Definitions Bring Party Data into a Standard BC

- The Join definition specifies which S_PARTY extension table to use
- The Join specification specifies to use PAR_ROW_ID
- The SVF is the desired column from the joined extension table

The screenshot shows the Siebel Object Explorer interface with three highlighted components:

- 1. Join definition:** A red box highlights the "Join Specifications" table in the "Joins" section, specifically the row where the Alias is "S_ORG_EXT" and the Table is "S_ORG_EXT".
- 2. Join specification:** A yellow box highlights the "Join Specifications" table in the "Join Specifications" section, specifically the row where the Name is "Account Id", the Destination Column is "PAR_ROW_ID", and the Source Field is "Account Id".
- 3. Single-value field:** A yellow box highlights the "Single Value Fields" table in the "Single Value Fields" section, specifically the row where the Name is "Opportunity" and the Table is "S_OPTY".

Copyright © 2007, Oracle. All rights reserved.

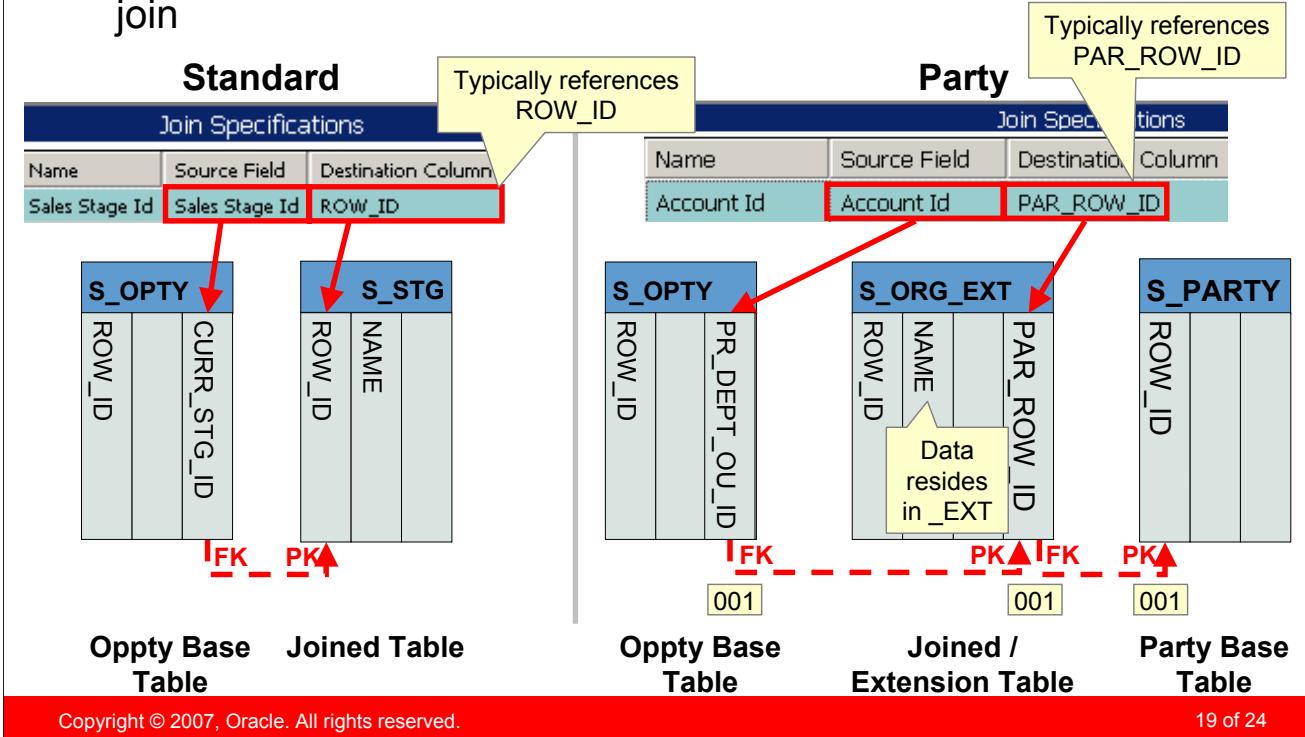
18 of 24

PAR_ROW_ID

PAR_ROW_ID is referenced in the join specification and identifies the primary key of the related account record.

Join Specification with Party: Similar to Standard But

- Destination references PAR_ROW_ID, not ROW_ID
- Keys reference base to extension to join, rather than base to join

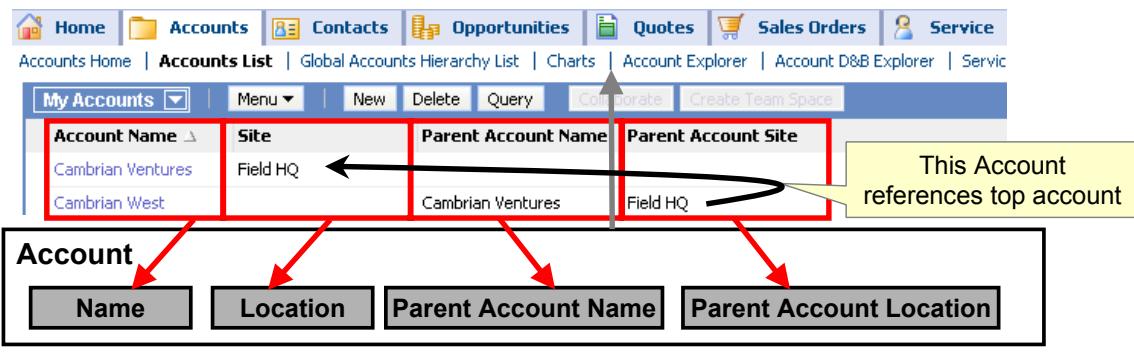


PAR_ROW_ID

PAR_ROW_ID is referenced in the join specification and identifies the primary key of the related account record.

Bringing Party Data into a Party Business Component

- Example: Bringing parent account data into the Account BC for display in an Account applet
 - ▶ A join in a BC that refers back to data in that same BC



Join Definition, Bringing Party Data into a Party BC

- Must use an explicit join to an extension table based on the appropriate foreign keys
 - ▶ The alias must be different from those used by implicit joins in this BC
- Must not use an existing implicit join to that extension table

Object Explorer

Project: ** All Projects **

Types Detail Flat

Siebel Objects

- Applet
- Application
- Business Component**
 - BusComp View Mode
 - Business Component I
 - Field
 - Join**
 - Join Constraint
 - Join Specification
 - Multi Value Field
 - Multi Value Link
 - Single Value Field
- Business Object
- Entity Relationship Diagram

View (ABC Account List View) - View ... Join List

Joins

Business Components

Name
Account

Since an implicit join to S_ORG_EXT already exists...

...an explicit join must be used, and it must have an alias

Alias	Table	Outer Join Flag
Parent Account	S_ORG_EXT	✓
Primary Account Address	S_ADDR_ORG	✓
Primary Account Position	S_ACCT_POSTN	✓
Primary Bill To Address	S_ADDR_ORG	✓
Primary Bill To Person	S_CONTACT	✓

Copyright © 2007, Oracle. All rights reserved.

21 of 24

Implicit Joins

Remember that there is an implicit join available for each extension table for a base table. S_PARTY has many extension tables, including S_ORG_EXT, S_CONTACT, S_POSTN, and S_USER. These implicit joins are used in other party business components to map their main data.

Map Field to Column in Party Table

- Select the appropriate explicit join
 - ▶ Do not use an implicit join to the table
- Select the desired column in the joined table
- Set the appropriate type (DTYPE_TEXT, etc.)

The screenshot shows the Siebel Object Explorer interface. On the left, the Object Explorer tree is visible under the 'Siebel Objects' category, with 'Single Value Field' selected. In the center, the 'Single Value Field List' window is open, showing a table named 'S_PARTY'. Below it, the 'Single Value Fields' section lists a field named 'A New Joined Field'. A red arrow points from the 'Join' button in this section to the 'Contact - S_ORG_EXT' entry in the 'Join' list, which is highlighted with a yellow callout box containing the text 'Select the explicit join instead of S_ORG_EXT'.

Name	Table
Account	S_PARTY

Name	Join	Column
A New Joined Field	Contact	S_ORG_EXT

Copyright © 2007, Oracle. All rights reserved. 22 of 24



Module Highlights

- BCs can include data from related joined tables
- Join definitions and join specifications define how to access joined table data
- Single-value fields reference the join definition and define the field to access in the joined table
- To create a join:
 - ▶ Locate the data source, examine existing joins, diagram the join, create the join definition, create the join specification, and define the SVF
- Joins that involve party data are similar to standard data; however, when bringing party data into:
 - A non-party BC, create a join specification based on PAR_ROW_ID
 - A party BC, use the appropriate explicit join



Lab

- In the lab you will:
 - ▶ Examine references when a join is not present as well as when a join is present
 - ▶ Create a join that brings in data from joined table and display it in the UI

ORACLE®

Siebel 8.0 Essentials

Module 29: Business Layer Configuration: Existing Business Components and Fields

29

Copyright © 2007, Oracle. All rights reserved.



Module Objectives

- After completing this module you should be able to:
 - ▶ Edit business component properties to capture business logic
 - ▶ Describe business component view modes
 - ▶ Edit field properties to capture business logic
 - ▶ Specify business component and field user properties
- Why you need to know:
 - ▶ Editing the properties of business components and fields helps you to more accurately and fully capture your business logic

ORACLE®

Business Component Properties

- Capture business logic
- Include:
 - ▶ Properties for editing
 - No Delete
 - No Insert
 - No Merge
 - No Update
 - ▶ Owner Delete
 - ▶ Search Specification
 - ▶ Sort Specification

Properties	
Business Component [Contact(Personal)]	
Name	Contact(Personal)
No Delete	FALSE
No Insert	FALSE
No Merge	FALSE
No Update	FALSE
Owner Delete	TRUE
Placeholder	
Popup Visibility Auto All	
Popup Visibility Type	
PreFetch Size	
Recipient Id Field	Id
Reverse Fill Threshold	
Scripted	FALSE
Search Specification	[Personal Contact] = 'Y'
Sort Specification	Last Name, First Name
Status Field	Row Status

Copyright © 2007, Oracle. All rights reserved.

3 of 30

Reference

Configuring Siebel Business Applications: Configuring Business Components

0/4

ORACLE®

Business Component Properties

Properties for Editing

Owner Delete Property

Search Specification Property

Sort Specification Property

Copyright © 2007, Oracle. All rights reserved.

4 of 30

Properties for Editing

- Set editing properties at the business component level to prevent deleting, inserting, merging, and updating records in all applets
- Example: To maintain record continuity, users cannot delete or change price lists once they have been created

The screenshot shows the Siebel Properties window for a 'Business Component [Price List]'. A red box highlights the 'No Delete' row in the table below. The main pane displays a list of price lists with columns: Name, Currency, Effective From, and Effective To.

Name	Currency	Effective From	Effective To
Americas Price List	USD	1/1/1998 11:59:59 AM	12/30/2008 4:00:00 PM
Interactive Selling for Horiz...	USD	1/15/2001 7:59:59 AM	12/30/2008 4:00:00 PM
Master Price List	USD	11/5/1999 11:59:59 PM	12/30/2008 4:00:00 PM
PCS	USD	8/6/2003 5:00:00 PM	

Copyright © 2007, Oracle. All rights reserved.

5 of 30

No Delete	If TRUE, once a record is created and committed, users cannot delete it.
No Insert	If TRUE, users cannot add new records.
No Merge	If TRUE, users cannot merge records.
No Update	If TRUE, users cannot change existing records.

Properties for Editing Continued

- These properties also appear on applets
 - ▶ Most restrictive settings are always used
 - ▶ Set these properties to FALSE at the business component level so you can make exceptions at the applet level

Business Component

Properties	
Business Component [Contact[Personal]]	
	Alphabetic Categorized
Name	Contact[Personal]
No Delete	FALSE
No Insert	FALSE
No Merge	FALSE
No Update	FALSE

Affects all applets referencing
this BC, or . . .

Applet

Properties	
Applet [Contact Form ReadOnly Applet]	
	Alphabetic Categorized
Name	Contact Form ReadOnly Applet
No Delete	TRUE
No Insert	TRUE
No Merge	TRUE
No Update	TRUE

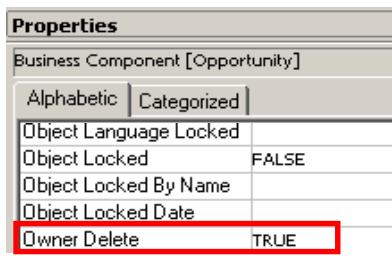
. . . can be customized for
each applet

2/4

ORACLE®

Owner Delete Property

- Set the Owner Delete property to TRUE to enable only the owner of a record to delete it
 - ▶ The primary on a team-based business component is the owner of the record



The screenshot shows the Siebel Properties window for a Business Component [Opportunity]. The 'Properties' tab is selected. In the list view, the 'Owner Delete' field is highlighted with a red border and contains the value 'TRUE'. Other fields shown include 'Object Language Locked' (Alphabetic), 'Object Locked' (FALSE), 'Object Locked By Name', and 'Object Locked Date'.

JRUBIN is primary owner; only JRUBIN can delete

Opportunity Name	Primary	Revenue	Sales Stage
Laptops For Kaboom	JRUBIN	\$25,000.00	

Copyright © 2007, Oracle. All rights reserved.

7 of 30

Search Specification Property

- Specifies records to be retrieved by the business component
 - ▶ Typically used when there are multiple business components based on the same table

The screenshot shows the Siebel Object Explorer on the left and the Business Component List on the right.

Object Explorer: Shows a tree structure under Siebel Objects > Business Component. Nodes include BusComp View Mode, Business Component!, Field, Join, Multi Value Field, Multi Value Link, Single Value Field, and Business Object.

Business Component List: Shows a table titled "Business Components". A single row is selected for "Contact(All)". The "Search Specification" column contains the expression: "[Personal Contact] = 'N' or ([Personal Contact] = 'Y' and [Owned By Id] = LoginId())". Two callout boxes point to this expression: one labeled "BC field" pointing to the "[Personal Contact]" part, and another labeled "Conditional statement" pointing to the "or" operator.

Name	Search Specification
Contact(All)	[Personal Contact] = 'N' or ([Personal Contact] = 'Y' and [Owned By Id] = LoginId())

Search Specification Expression

- Consists of field names, constants, functions, logical operators, and comparison operators
- Example: For a given user, Contact(All) retrieves:
 - ▶ All contacts where the Personal Contact flag is “N”
 - ▶ All personal contacts belonging to the user

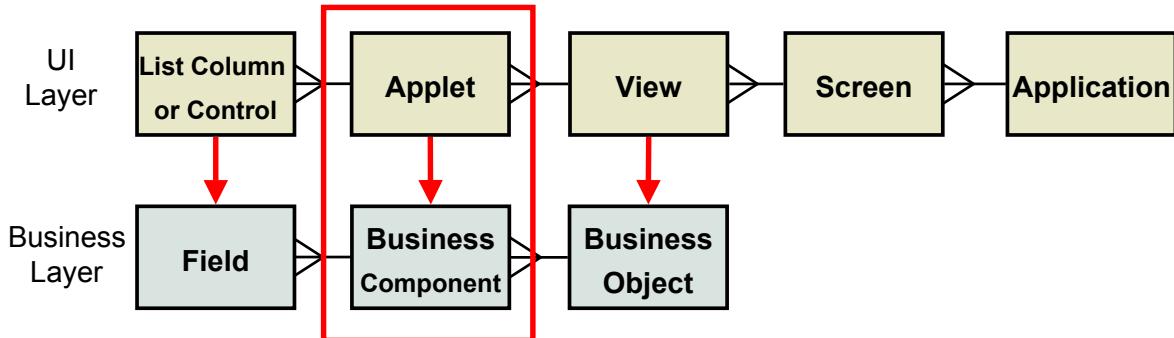
The screenshot shows the Siebel Object Explorer interface with the Business Component List selected. In the center, the search specification for Contact(All) is displayed as: `[Personal Contact] = 'N' or ([Personal Contact] = 'Y' and [Owned By Id] = LoginId())`. A callout box highlights the entire expression. Annotations explain the components: '...must match exactly' points to the field name `[Personal Contact]`; 'Logical operator' points to the `=` and `or` operators; 'Function' points to the `LoginId()` function.

Copyright © 2007, Oracle. All rights reserved.

9 of 30

Search Specification Considerations

- Appear on business component properties and applet properties
 - ▶ Search specifications are part of the WHERE clause for all applets that display this business component
 - Predefined queries may also contribute to the WHERE clause
 - ▶ Search specifications on the applet are combined with search specifications on the business component with an AND in the resulting SQL statement
- Avoid mutually exclusive search specifications
 - ▶ Avoid competing search specifications that preclude record display



Copyright © 2007, Oracle. All rights reserved.

10 of 30

Reference

Object Types Reference

4/4

ORACLE®

Sort Specification Property

- Determines the sort order of the retrieved records
 - ▶ Use (DESC) or (DESCENDING) to sort that field in reverse order
 - ▶ Cannot be set at the applet level

The screenshot shows the Siebel Contacts List applet interface. A yellow callout box points to the contact list with the text: "To sort by last name, then first name...". Another yellow callout box points to the "Sort Specification" tab in the Business Components window with the text: "...configure a sort specification". A red box highlights the "Sort Specification" tab. A third yellow callout box points to the "Name" column in the "Sort Specification" table with the text: "Must match exactly". A red box highlights the "Name" column header.

Name	Sort Specification
Contact>All)	Last Name, First Name

The screenshot shows the Siebel Indexes configuration window. A yellow callout box points to the "Index Columns" table with the text: "Ensure that an index exists to support the sort specification". A red box highlights the "Index Columns" table.

Column Name	Sequence	Sort Order
FST_NAME	3	Asc
LAST_NAME	2	Asc
OWNER_PER_ID	1	Asc
PRIV_FLG	4	Asc

Copyright © 2007, Oracle. All rights reserved.

11 of 30

ORACLE®

BusComp View Mode

- Specifies that the BC is subject to Access Control
 - ▶ Owner Type property identifies the Access Control mechanism that limits records a user can access
- BCs may have multiple access mechanisms
 - ▶ Allows records to be associated with different owner types
 - ▶ Supports multiple view modes

Name	Owner Type	Visibility Field	Visibility MvField
Organization	Organization	Organization	
Sales Rep	Position		Sales Rep

Copyright © 2007, Oracle. All rights reserved.

12 of 30

Reference

[Object Types Reference](#)

[Security Guide for eBusiness Applications](#)

ORACLE®

BusComp View Mode, Example

- The Opportunity business component has a position-based BusComp View mode
 - ▶ Invoked when user navigates to the My Opportunities or My Team's Opportunities view
- Visibility MVField property is set to Sales Rep, so application reads list of positions from Sales Rep MVF to determine record visibility
 - ▶ The My Opportunities view shows records with the user's current position listed in the Sales Rep MVF
 - ▶ The My Team's Opportunities view shows records where user's direct or indirect reports are primaries on the Sales Rep MVF



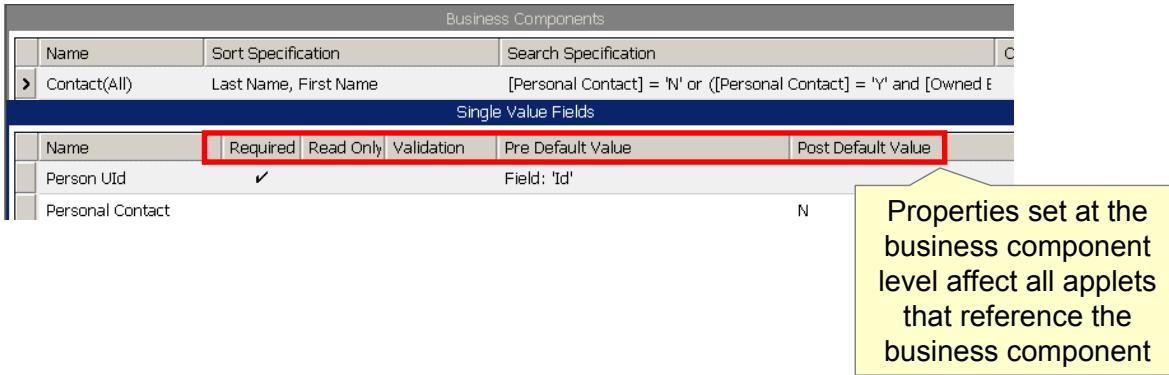
Copyright © 2007, Oracle. All rights reserved.

13 of 30

ORACLE®

Field Properties

- Customize fields by editing property values
- Setting field properties at the business component level sets them across all applets that reference the business component
 - ▶ Each applet references one, and only one, business component



Copyright © 2007, Oracle. All rights reserved.

14 of 30

Reference

Object Types References

0/4

ORACLE®

Business Component Field Properties

Required and Read Only

Validation

Default Values

Calculated Fields

Copyright © 2007, Oracle. All rights reserved.

15 of 30

Required and Read Only Properties

- Setting Required to TRUE prevents user from leaving field blank
- Setting Read Only to TRUE prevents user from editing the value

The screenshot illustrates the configuration of business components and the resulting user interface behavior.

Business Components Configuration: On the left, the 'Business Components' screen shows a 'Single Value Fields' section. A row for 'Last Name' has a red border around its input field, and a tooltip '1. Contacts require last names' points to it. Another row for 'Name' has a red border around its input field, and a tooltip 'Required' points to it.

User Interface: On the right, a 'Contact' creation dialog is shown. It includes fields for 'Last Name*', 'First Name*', 'Middle Initial', 'Mr/Ms', and 'Job Title'. The 'Last Name*' field is highlighted with a red border, and a tooltip '2. Required field marked with asterisk' points to it. A Siebel error dialog box titled 'Siebel' displays the message: "'Last Name' is a required field. Please enter a value for the field. (SBL-DAT-00498)'. A tooltip '3. Attempting to save without last name causes error message' points to this dialog.

Copyright © 2007, Oracle. All rights reserved.

16 of 30

Note

To display custom error messages, you must use Data Validation Manager.

Validation Property

- Checks the rule when a record is saved to ensure that the field data is valid
- Message is customizable
- Can refer only to business component fields in the same record

Business Components

Name	Project	Table
Contact	Contact	S_PARTY

Single Value Fields

Name	Validation	Validation Message	Validation Message - String Ref	Validation Message - String Override	Message Display Mode
Birth Date	<code><=Today()</code>	Enter today's date or a date prior to today		Enter today's date or a date prior to	User Msg with Error Code

Birth date must be prior to the current date

Text displayed when validation fails

Specify existing string in SRF...

...or enter a custom message

Option to include system error message

Copyright © 2007, Oracle. All rights reserved.

17 of 30

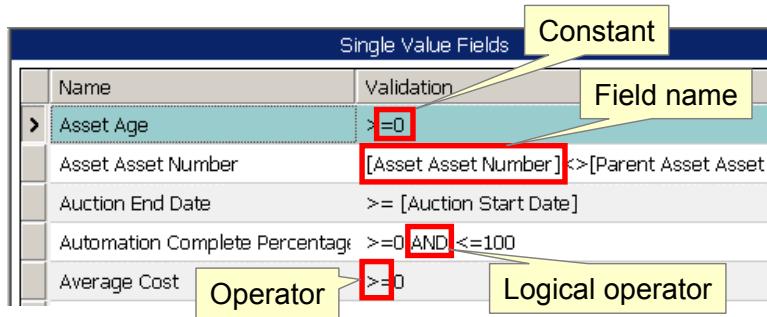
Reference

[Object Types Reference](#)

[Siebel Developer's Reference](#)

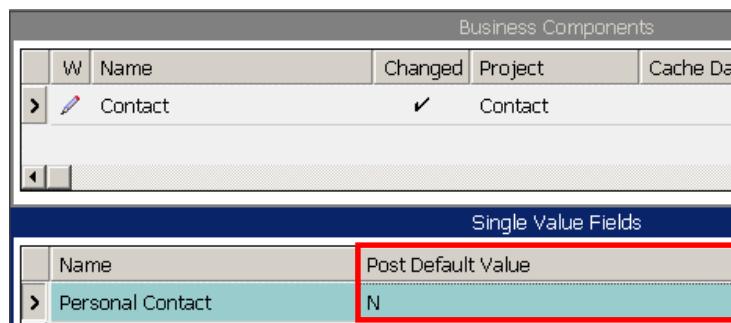
Validation Property continued

- Rules are expressed as a combination of logical operators, constants, field names, and predefined functions



Default Values for Fields

- Pre Default Value
 - ▶ Automatically assigns a value to a field for a new record
- Post Default Value
 - ▶ Assigns a value to a field, if not entered by the user, before the record is inserted into the database
- Example: If the user does not designate a contact as personal, the system assumes it is not personal, and sets it to N



Calculated Fields

- Derive their value from values in other fields in the same record of the business component in which the calculated field resides
- Cannot be stored in the database; therefore, there is no associated column
- To enable, set Calculated property to TRUE (checked)

The screenshot shows the Siebel Business Components interface. At the top, it says "Business Components" and "Contact". Below that, there's a toolbar with icons for New, Edit, Name, Changed, Project, Cache Data, and Class. The "Class" dropdown is set to "CSSBCU". A yellow callout box points to the "Column" column in the table below, stating: "Column property is empty because calculated fields are not stored in the database".

Name	Calculated	Calculated Value	Column
Last Name	<input checked="" type="checkbox"/>	IIF (Language () = "JPN" OR Language = "DEU", [Last Name] + ", " + [First Name], [Last Name])	LAST_NAME
Last Name, First Name	<input checked="" type="checkbox"/>	[Last Name] + ", " + [First Name]	
Last Name, First Name ENU	<input checked="" type="checkbox"/>	[Last Name] + [First Name]	
Last Name, First Name JPN	<input checked="" type="checkbox"/>	[Last Name] + [First Name]	

Copyright © 2007, Oracle. All rights reserved.

20 of 30

Restrictions on Calculated Fields

- Calculated fields are read-only
- System does not validate values of calculated fields
- Sorting on calculated fields is not supported
- Querying on calculated fields is supported
 - ▶ Performance depends upon whether functions in the query expression can be incorporated into the SQL statement

Calculated Value Property

- Is an expression built from:
 - ▶ Field names in the same business component
 - ▶ Field names from the parent business component
 - Current BC must have the child in a detail view
 - ▶ Standard functions
 - ▶ String, numeric, and logical operators

Single Value Fields				
Name	Calculate	Calculated Value	Comments	
Name and Location	✓	[Name] + ": " + [Location]		
Row Status Asterisk	✓	IIF ([Row Status] = "Y", "***", "")		
Timestamp	✓	Timestamp ()		
Today	✓	Today ()		

A yellow callout box labeled "System functions" points to the "Timestamp ()" and "Today ()" entries in the table.

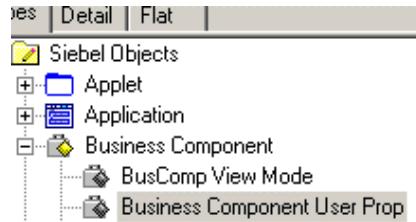
Reference

[Object Types Reference](#)
[Siebel Developer's Reference](#)

ORACLE®

User Properties

- Are object definitions added as children to an applet, business component, control, field, or list column
- Used to configure specialized behavior beyond what is configured in the parent object definition's properties
- Each user property contains its own conditional logic
 - ▶ Can implement specific, custom IF/THEN logic
 - ▶ Many use their own unique syntax
- Can be used as a declarative alternative to scripting
- A wide variety of user properties exist, such as:
 - ▶ BC Read Only Field, Field Read Only Field, and On Field Update Set



Copyright © 2007, Oracle. All rights reserved.

23 of 30

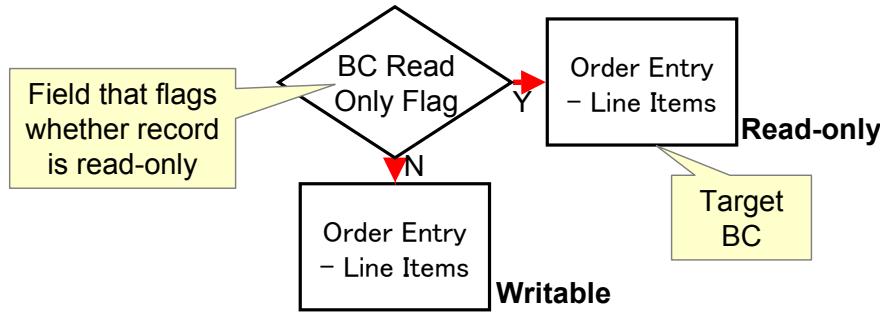
User Properties

There are a large number of user properties, each with its own specialized behavior. For additional information, refer to Siebel Developer's Reference.

ORACLE®

BC User Property Scenario

- Business challenge
 - ▶ Based on values the end user enters at run time, dynamically disallow updates to record
- Business solution
 - ▶ Use the BC Read Only Field user property
 - Allows you to specify a field on the business component that determines whether individual records are read-only
 - ▶ Example: Shipped orders cannot accept changes to order entry line item records in an existing order
 - Order has shipped, and it is too late to add more line items to the order



Copyright © 2007, Oracle. All rights reserved.

24 of 30

ORACLE®

User Property, BC Read Only Field Example

- Create a calculated field on the Order Entry – Line Items business component whose value is either Y or N
 - ▶ BC Read Only Field user property requires an input value of Y or N, so create a calculated field whose values are limited to those two values
- For this example, create a calculated field named BC Read Only Flag with the expression: IF([Status] = "Shipped", "Y", "N")
 - ▶ This sets the value of the calculated field to Y if the status is shipped, N otherwise

Single Value Fields		
Business Components		
Name	Table	
Order Entry - Line Items	S_ORDER_ITEM	
Single Value Fields		
Name	Calculated	Calculated Value
BC Read Only Flag	✓	IF([Status] = "Shipped", "Y", "N")

BC Read Only Flag SVF has a value of Y if the status of the item is shipped, N otherwise

Copyright © 2007, Oracle. All rights reserved.

25 of 30

ORACLE®

User Property, BC Read Only Field Example Continued

- Add the BC Read Only Field user property
 - ▶ Set the Value property to BC Read Only Flag
 - Value is the name of the calculated field on the BC
 - ▶ When the value of this field is “Y”, the current record is read-only

Create User Property which references the calculated field

The screenshot shows the Siebel Business Component User Properties interface. A yellow callout box points to the top right of the window, containing the text "Create User Property which references the calculated field". Below this, the main window displays the "Single Value Fields" section for the "Order Entry - Line Items" business component. A red arrow points from the "Name" column of the table to the "BC Read Only Flag" entry. The "Value" column for this entry contains the formula "IF([Status] = "Shipped", "Y", "N")". The "Comments" column notes "Make BC read-only if order already shipped".

Name	Table	Business Components
Order Entry - Line Items	S_ORDER_ITEM	Business Components
Business Component User Properties		
Single Value Fields		
Name	Value	Comments
BC Read Only Flag	BC Read Only Flag	Make BC read-only if order already shipped
BC Read Only Flag	✓	IF([Status] = "Shipped", "Y", "N")

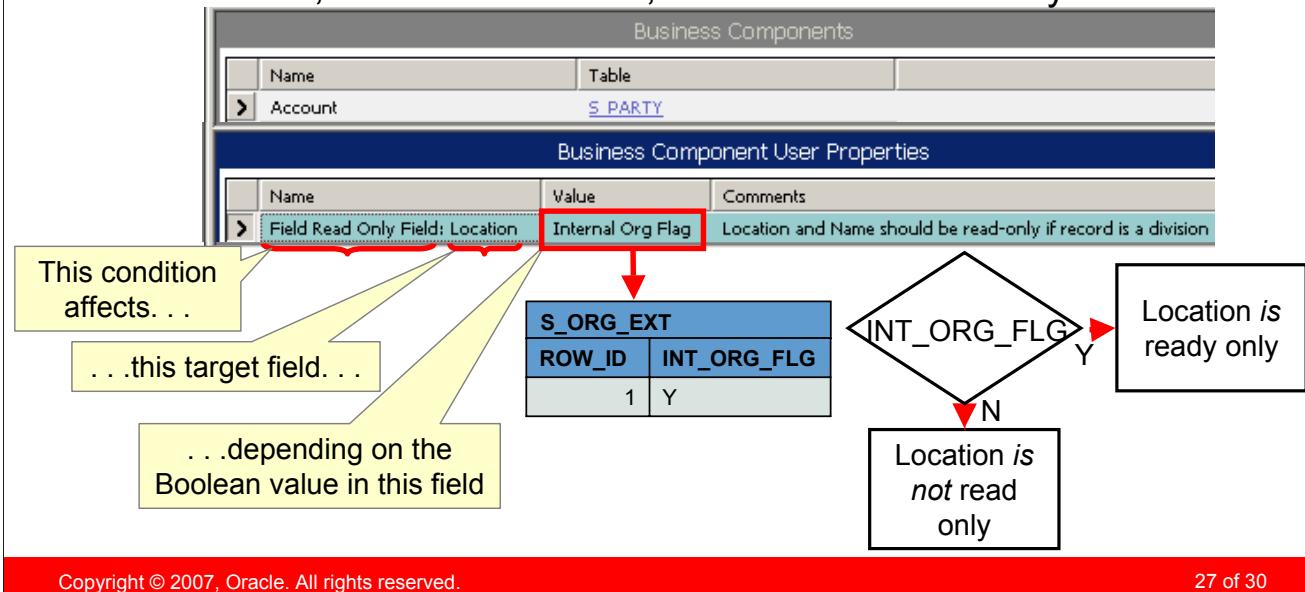
Copyright © 2007, Oracle. All rights reserved.

26 of 30

ORACLE®

User Property, Field Read Only Field

- Sets a specific field in a business component to be read-only
 - ▶ Field Read Only Field: Location sets Location field as the target of a conditional statement
 - ▶ Value is the name of a field that contains a Boolean value
- At run-time, if condition is met, field is set to read only



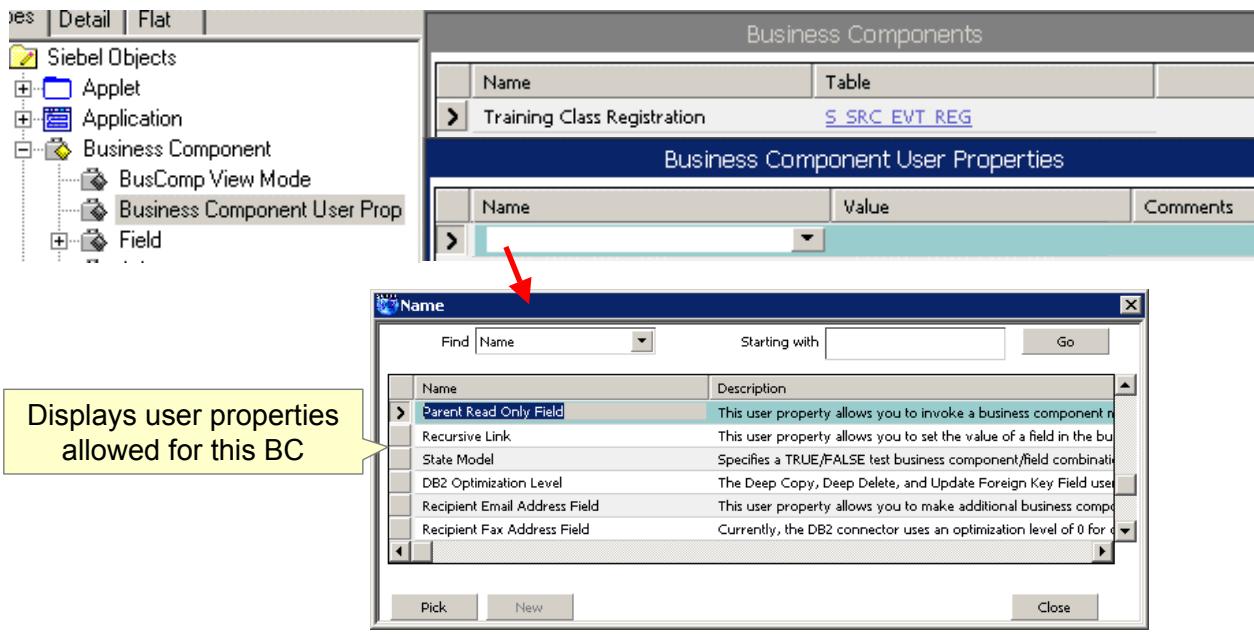
Copyright © 2007, Oracle. All rights reserved.

27 of 30

ORACLE®

Adding a User Property

- Is done via a picklist in the Name property in the Business Component User Properties window
 - ▶ Only those user properties allowed for parent BC are displayed



Copyright © 2007, Oracle. All rights reserved.

28 of 30



Module Highlights

- BC properties are set to control behavior:
 - ▶ Editing properties, Owner Delete, Search Specification, and Sort Specification
- BC field properties are set to control behavior:
 - ▶ Required, Read Only, Validation, Pre and Post Default, and Force Case
- BusComp View Mode specifies BC is subject to access control
- Calculated fields derive their value from other fields in the same record of the BC in which the calculated field resides
- User Properties have their own specialized logic that can be added to a BC
 - ▶ Declarative alternative to scripting



Lab

- In the lab you will:
 - ▶ Incorporate business logic into the application by configuring properties of business components and fields

Copyright © 2007, Oracle. All rights reserved.

30 of 30

ORACLE®

Siebel 8.0 Essentials

Module 30: Business Layer Configuration: New Business Components and Fields

30

Copyright © 2007, Oracle. All rights reserved.

30

Oracle Internal & Oracle Academy Use Only



Module Objectives

- After completing this module you should be able to:
 - ▶ Create a new business component (BC)
 - ▶ Add a business component to a business object
- Why you need to know:
 - ▶ Enables you to incorporate additional business entities that do not correspond to business components in a standard Siebel application
 - ▶ Enables you to tailor your business logic



Business Challenge

- Siebel-provided business components capture most commonly-used business entities, but they do not cover every possibility
- Example: Sales organizations might record personal data about contacts, such as:
 - ▶ What colleges the contact attended
 - Name of college, years attended, major field of study, sports played, honors received, and so on
 - ▶ The contact's favorite restaurants
 - Name and location, price range, type of cuisine, and so on
- To capture this kind of information may require:
 - ▶ Multiple fields to capture the details
 - ▶ 1:M or M:M relationship to the parent

Copyright © 2007, Oracle. All rights reserved.

3 of 22

Reference

Configuring Siebel Business Applications: Configuring Business Components

30

Oracle Internal & Oracle Academy Use Only



Business Solution: Create New Business Components

- To capture this sort of information, create new business components
 - ▶ For example, create college and restaurant child BCs of the Contact parent BC
- Used to add entities specific to your organization that are not in the Siebel repository
- You can base these business components on tables supplied by Oracle
 - ▶ Standard
 - ▶ Extension
 - ▶ 1:M
 - ▶ M:M

ORACLE®

Scenario: 1:M Relationship With Existing Table

- Example: Create Colleges Attended and Favorite Restaurants business components for sales organization
- Colleges Attended and Favorite Restaurants have a 1:M relationship with Contact
 - ▶ Requires two business components, Colleges Attended and Favorite Restaurants
 - One for Contact : Colleges Attended
 - One for Contact : Favorite Restaurants
- When creating a BC with a 1:M relationship with the parent business component, consider using a 1:M extension table



ORACLE®

Review: Standard 1:M Extension Tables

- Predefined in the repository for many business components
- Have name of parent table appended with _XM

The screenshot shows the Siebel Business Layer Configuration interface. In the top navigation bar, there are tabs for 'Tables', 'Extend', 'Apply', and 'Activate'. The 'Tables' tab is selected. Below the tabs, a table lists tables in the repository. One table, 'P_CONTACT_XM', is highlighted with a blue selection bar. The table has several columns: ATTRIB_46, ATTRIB_47, CONFLICT_ID, CREATED, CREATED_BY, LAST_UPD, LAST_UPD_BY, MODIFICATION_NUM, NAME, PAR_ROW_ID, ROW_ID, and TYPE. The first two columns, ATTRIB_46 and ATTRIB_47, are highlighted with red boxes. Three callout boxes with yellow backgrounds point to these columns:

- 'Contains many predefined ATTRIB columns of varying type'
- 'NAME column stores the name of the child entity'
- 'PAR_ROW_ID column stores foreign key to ROW_ID in main table'

Another callout box points to the 'TYPE' column:

- 'TYPE column identifies the child business component'

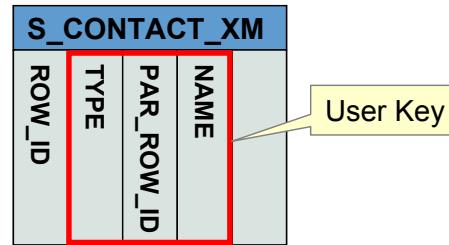
Copyright © 2007, Oracle. All rights reserved.

6 of 22

ORACLE®

User Key

- TYPE, PAR_ROW_ID, and NAME serve as a user key for the table
 - ▶ These three columns are required
 - Must provide values for all three columns when creating a new record
 - ▶ Combined value of the three columns must be unique



Copyright © 2007, Oracle. All rights reserved.

7 of 22

Ensuring that NAME Is Unique For BCs with data that cannot be stored in a Varchar 100 column, you must still provide a unique value for NAME

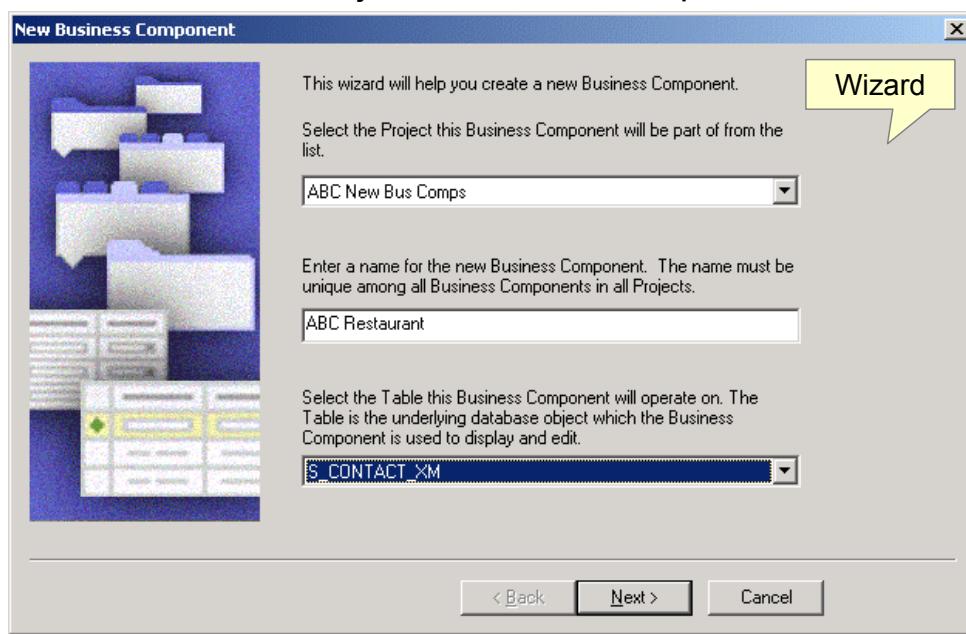
- NAME has physical type varchar 100
- If two rows in table share same TYPE & PAR_ROW_ID, then NAME must be unique
- Workaround is to store value of ROW_ID in NAME

30

ORACLE®

Business Component Wizard

- Use the Business Component Wizard to create a new business component
 - ▶ Select File > New Object, select BusComp, and then click OK



Copyright © 2007, Oracle. All rights reserved.

8 of 22



Business Component Wizard Inputs

- The Business Component Wizard requires the following inputs:
 - ▶ Project new business component will be part of
 - ▶ Business component name
 - Must be unique among all business components in all projects
 - ▶ The business component's table name
 - ▶ Fields that will be part of this BC
 - Selected in the wizard by specifying Table Column, then entering a field name
 - ▶ Specify TYPE column and relate it to Type field
 - ▶ Specify PAR_ROW_ID column and relate it to Id field
 - ▶ Specify NAME column and relate it to Name field
 - ▶ Use ATTRIB_ columns for additional data
 - Columns are limited to the table you select in wizard
 - Use a separate column for each field

Establishes
User Key

ORACLE®

Business Component Wizard Outputs

- The Business Component Wizard:
 - ▶ Creates a business component object and its child object definitions
 - ▶ Creates fields and relates them to table columns
 - Aside from what you specify in the wizard, uses default field and table values; for example, Owner Delete is set to FALSE

The screenshot shows the Siebel Object Explorer and the Single Value Field List windows.

Object Explorer: Shows the project as "All Projects" and the types as "Types". Under Siebel Objects, the "Business Component" node is expanded, showing sub-items like BusComp View Mode, Field, Join, Multi Value Field, Multi Value Link, Single Value Field, and Contact Id.

Single Value Field List: This window has two tabs: "Single Value Field List" and "Single Value Fields".

- Business Components:** A table with columns Name, Table, Project, and Search Specification. It contains one row: "ABC Restaurant" with Table "S_CONTACT_XM" and Project "Asset Management".
- Single Value Fields:** A table with columns Name, Column, and Type. It contains three rows:

Name	Column	Type
Contact Id	PAR_ROW_ID	DTYPE_ID
Restaurant Name	NAME	DTYPE_TEXT
Type	TYPE	DTYPE_TEXT

Annotations explain the actions:

- A callout points to the "ABC Restaurant" row in the Business Components table: "Creates new fields and relates them to columns"
- A callout points to the "ABC Restaurant" row in the Business Components table: "Creates the business component object definition"

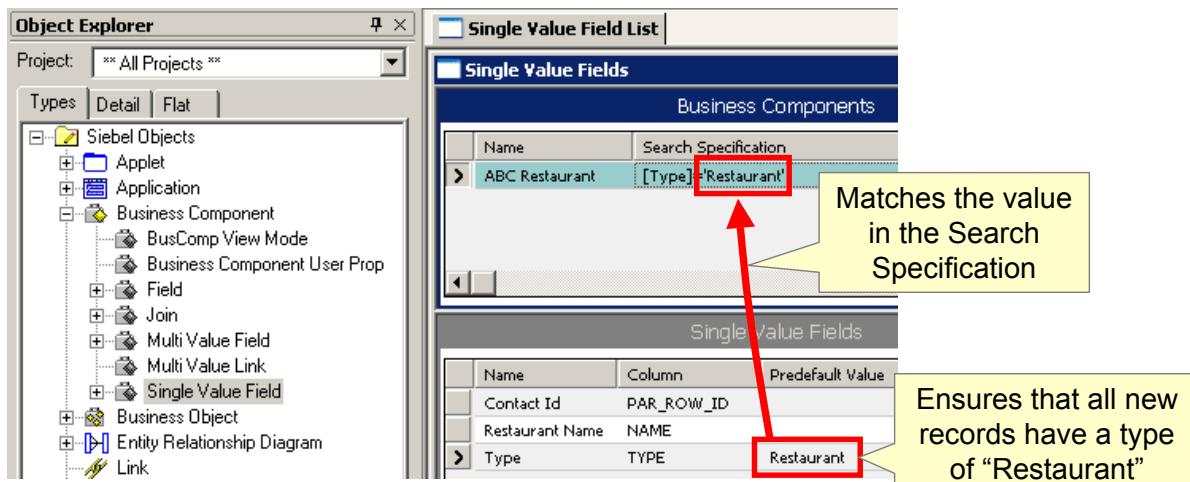
Copyright © 2007, Oracle. All rights reserved.

10 of 22

ORACLE®

Set the Pre-Default Value

- Automatically assigns a value to a field for a new record
 - ▶ For extension tables, set Predefault Value property of the Type field to the value used in the BC search specification
 - ▶ Can be used to ensure that the user key is populated



ORACLE®

Set Search Specification

- Manually set the search specification to match the unique TYPE value for the business component
 - ▶ Specifies the records to be retrieved by the business component
- Typically used with extension tables
 - ▶ Supports the case where multiple BCs are based on the same extension table

The screenshot shows the Siebel Object Explorer and the Single Value Field List windows.

Object Explorer:

- Project: ** All Projects **
- Types tab selected.
- Siebel Objects tree:
 - Applet
 - Application
 - Business Component** (selected)
 - BusComp View Mode
 - Business Component User Prop
 - Field
 - Join
 - Multi Value Field
 - Multi Value Link
 - Single Value Field
- Business Object
- Entity Relationship Diagram
- Link

Single Value Field List:

Single Value Fields table (Business Components row):

Name	Search Specification
ABC Restaurant	[Type] = 'Restaurant'

A callout box points to the search specification cell for 'ABC Restaurant' with the text: "Search retrieves only records with a Type of 'Restaurant'".

Single Value Fields table:

Name	Column	Predefault Value
Contact Id	PAR_ROW_ID	
Restaurant Name	NAME	
Type	TYPE	Restaurant

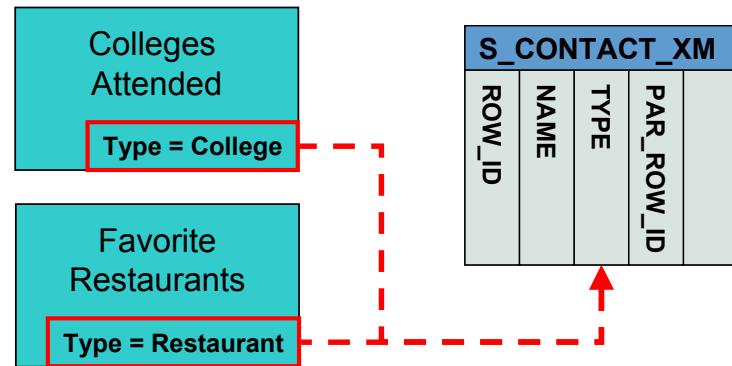
Copyright © 2007, Oracle. All rights reserved.

12 of 22

ORACLE®

Multiple Business Components

- You can use one table to map multiple user-defined child business components
- Each BC has a unique TYPE value
 - ▶ Stored in the TYPE column of the table
- Each BC retrieves only those rows with its TYPE value
- Specify properties on business components and fields
 - ▶ Search Specification
 - ▶ Pre Default Value



Copyright © 2007, Oracle. All rights reserved.

13 of 22

30

0/3

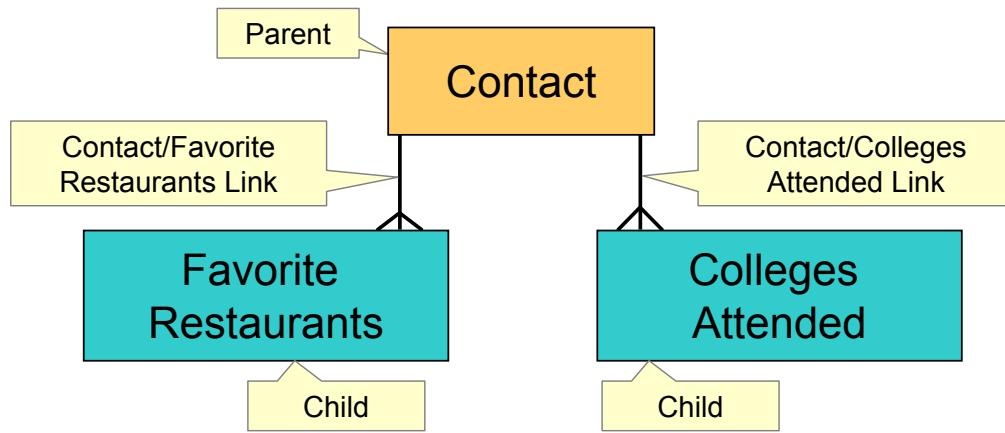


Post-Creation Tasks

1. Relate Child and Parent BCs
2. Reference Business Component in Business Object
3. Enable Display Data in UI

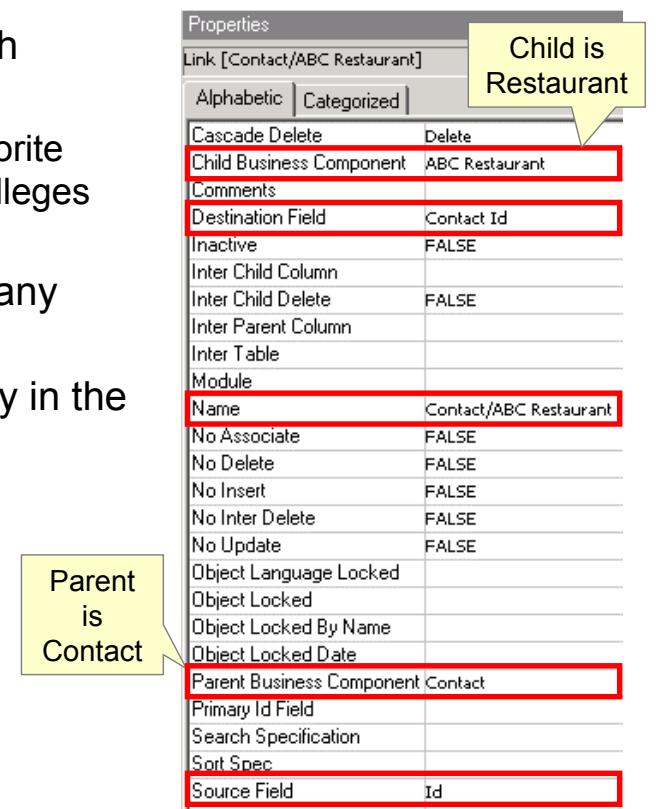
Relate Child and Parent BCs

- For each child/parent relationship, associate the newly configured child business component to the parent:
 - ▶ Create a link definition to relate child and parent records



Relate Child and Parent BCs Continued

- Create a separate link for each child/parent relationship
 - ▶ In our scenario, Contact/Favorite Restaurants and Contact/Colleges Attended
- Link creation is same as with any standard BC
- This will be the link you specify in the business object



Properties	
Link [Contact/ABC Restaurant]	
Alphabetic Categorized	
Cascade Delete	Delete
Child Business Component	ABC Restaurant
Comments	
Destination Field	Contact Id
Inactive	FALSE
Inter Child Column	
Inter Child Delete	FALSE
Inter Parent Column	
Inter Table	
Module	
Name	Contact/ABC Restaurant
No Associate	FALSE
No Delete	FALSE
No Insert	FALSE
No Inter Delete	FALSE
No Update	FALSE
Object Language Locked	
Object Locked	
Object Locked By Name	
Object Locked Date	
Parent Business Component	Contact
Primary Id Field	
Search Specification	
Sort Spec	
Source Field	Id

Copyright © 2007, Oracle. All rights reserved.

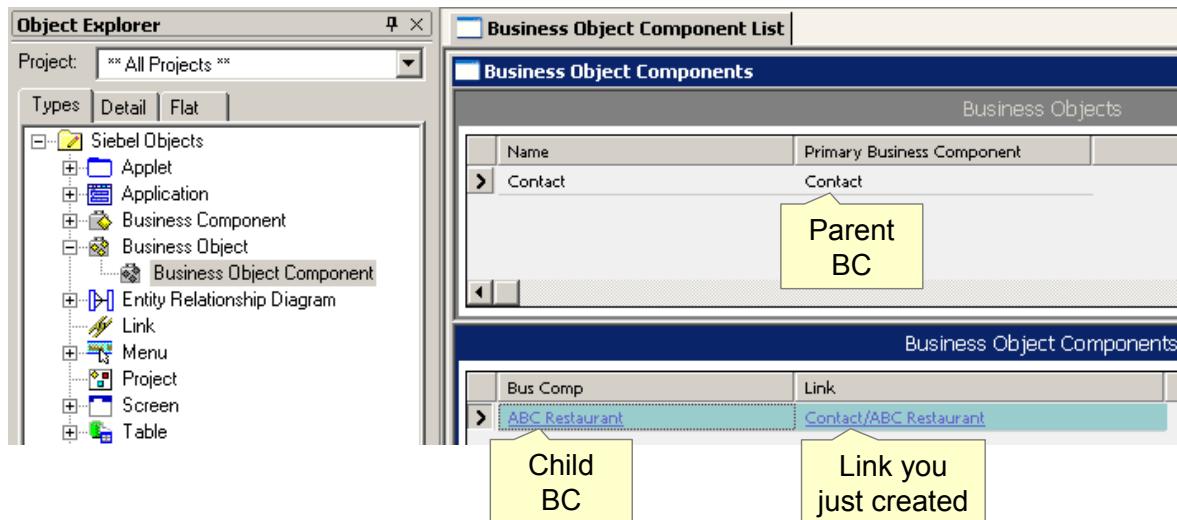
16 of 22

2/3

ORACLE®

Reference Business Component in Business Object

- Create a Business Object Component definition for each new business component
 - ▶ The business object defined for the parent is the one to use
 - ▶ The business component is the child you just created
 - ▶ Set the Link property



Copyright © 2007, Oracle. All rights reserved.

17 of 22

New Parent BCs and Business Objects If you create a *parent* business component you must also create an entirely new business object.

30

3/3

ORACLE

Enable Display Data in UI

- Build applets and views as required to display data from the business component
 - ▶ To prevent users from changing the value, do not display the Type field



Copyright © 2007, Oracle. All rights reserved.

18 of 22



Creating a Standalone Business Component

- May be necessary when your company's business logic requires a new table be added to the database
 - ▶ In most cases, Siebel's extensive database should meet most of your needs. There could be exceptions.
- To create a standalone BC:
 - ▶ Create a new table to hold BC data
 - ▶ Add appropriate columns to this table
 - ▶ Create a new BC and add BC fields
 - ▶ Create a new Business Object and reference the BC in this BO
 - ▶ Enable data display in the UI
 - ▶ Standalone BCs are based on standalone tables and have no relationships to other BCs or tables

Copyright © 2007, Oracle. All rights reserved.

19 of 22

Creating a Table

The new steps of creating a table and adding columns is covered in a later module.

30



Module Highlights

- Create a new BC as child of an existing BC to capture new information
- When possible, use predefined _XM tables to store data
- Use the BC Wizard to create a new BC
- When using extension tables, use one extension table to map to multiple child BCs
 - ▶ Each BC has a unique TYPE value
 - ▶ Specify properties on BC and fields
 - Include required fields, such as Type and NAME
- Create a link definition to associate new child BCs with parent
- Add new BC to a BO, then create applets and views to display data
- Create a standalone BC, when necessary



Lab

- In the lab you will:
 - ▶ Create and display a new business component based on a 1:M extension table

Copyright © 2007, Oracle. All rights reserved.

21 of 22

30

ORACLE®

Siebel 8.0 Essentials

Module 31: Business Layer Configuration: Picklists

31

Copyright © 2007, Oracle. All rights reserved.

Oracle Internal & Oracle Academy Use Only



Module Objectives

- After completing this module you should be able to:
 - ▶ Describe the differences between dynamic and static picklists
 - ▶ Administer a list of values
 - ▶ Configure a static or dynamic picklist
- Why you need to know:
 - ▶ Enables you to add picklists to your Siebel application



Picklists

- Allow users to populate one or more single-value fields by selecting a value from a list
 - ▶ Enforces business rules and policies
 - ▶ Makes data entry faster
 - ▶ Reduces errors
- Can be either static or dynamic

Copyright © 2007, Oracle. All rights reserved.

3 of 22

References

Configuring Siebel Business Applications: Configuring Picklists and Pick Applets

Configuring Siebel Business Applications: Creating and Administering Lists of Values

ORACLE®

Static Picklist

- Displays values in a drop-down list for user selection
- Copies the selected value into destination field
 - ▶ No link to the original picklist data
- Can be bounded or unbounded
 - ▶ Bounded picklist forces users to enter only a value in the picklist
 - ▶ Unbounded picklist permits users to enter any value into the field
- Draws values from picklist data managed by an administrator
 - ▶ Values displayed in static picklists do not change during run time

The screenshot shows a Siebel CRM interface for the 'Contacts' module. At the top, there's a navigation bar with tabs like 'More Info', 'Activities', 'Contacts' (which is selected), 'Notes', 'Opportunities', 'Revenues', and 'Service Prof'. Below the navigation bar is a toolbar with buttons for 'Menu', 'Add', 'New', 'Delete', and 'Query'. The main area is a table with columns: 'Mr/Ms', 'First Name', 'Last Name', 'Middle Name', and 'Job Title'. A dropdown menu is open under the 'Mr/Ms' column for a row where 'First Name' is 'Gloria' and 'Last Name' is 'Beasley'. The dropdown menu contains options: 'Ms', 'Miss', 'Mr.', 'Mrs.', 'Dr.', 'Mr.', and 'Mr.'. The option 'Ms' is highlighted. A yellow callout box labeled 'Drop-down list' points to the dropdown icon. Another yellow callout box labeled 'UI visual cue for static picklist' points to the dropdown menu itself.

Mr/Ms	First Name	Last Name	Middle Name	Job Title
> Ms	Gloria	Beasley		Manager, Call Center
Miss	James	Conway		Regional Sales Manager
Mr.	JB			Area Vice President, I
Mrs.	Mahesh			Chief Financial Officer
Dr.				VP Sales
Mr.	Carl	Thornton	M	Vice President Business
Mr.	Jung	Wang		

Copyright © 2007, Oracle. All rights reserved.

4 of 22

ORACLE

Static Picklist Values

- Are stored for *all* static picklists in the S_LST_OF_VAL table
- Appear in List of Values or LOV Explorer administrative views
 - ▶ Administration – Data > List of Values (or LOV Explorer)
- Each have a type, which indicates which static picklist the value belongs to

LOV:

Home Accounts Contacts Opportunities Quotes Sales Orders Service Administration - Data

Units of Measurement Administration | Order Action Types | Order Types | Activity Templates | Competitors | Decision Issues | List of Values

Type	Display Value	Language-Indep	Language Name	Parent LIC	Order	Active
MRGUSR_FRIENDLY_NOTIFICATION	Conflicts	Conflicts	English-American			✓
MRGUSR_FRIENDLY_NOTIFICATION	FALSE	FALSE	English-American			✓
MRGUSR_FRIENDLY_NOTIFICATION	TRUE	TRUE	English-American			✓
MR_MS	Miss		English-American	1		✓
MR_MS	Mr.		English-American	1		✓
MR_MS	Ms.		English-American	2		✓
MR_MS	Mrs.		English-American	3		✓
MR_MS	Dr.	Dr.	English-American	4		✓
MSGBOARD_MSG_TYPE	Info	Info	English-American	1		✓
MSGBOARD_MSG_TYPE	Summary	Summary	English-American	2		✓

Copyright © 2007, Oracle. All rights reserved. 5 of 22

ORACLE®

Administering a Static Picklist

- Use Administration – Data > List of Values Explorer view
- Select an existing picklist or create a new picklist type in the List of Values - Types applet
- Expand the type and select the child Values folder
- Edit the picklist values in the List of Values (LOV) applet

The screenshot shows the Siebel List of Values Explorer interface. At the top, there's a navigation bar with links like Home, Accounts, Contacts, Opportunities, Quotes, Sales Orders, Service, and Administration - Data. The Administration - Data link is highlighted with a red box. Below the navigation bar is a toolbar with buttons for New and Query. The main area is titled "List of Values" and shows a table of data with columns: Code, Display Value, Language Name, Active, Translate, and Multiling. The data rows are: Miss., Miss, English-American, checked, checked; Mr., Mr., English-American, checked, checked (highlighted with a yellow background); Ms., Ms., English-American, checked, checked; Mrs., Mrs., English-American, checked, checked; Dr., Dr., English-American, checked, checked. On the left, there's a tree view of types and values. A red box highlights the "Values" folder under the "MR_MS" type. A red arrow points from this highlighted folder to the "Values" column in the table. The bottom of the screen has a footer with copyright information and page numbers.

Code	Display Value	Language Name	Active	Translate	Multiling
Miss.	Miss	English-American	✓	✓	
Mr.	Mr.	English-American	✓	✓	
Ms.	Ms.	English-American	✓	✓	
Mrs.	Mrs.	English-American	✓	✓	
Dr.	Dr.	English-American	✓	✓	

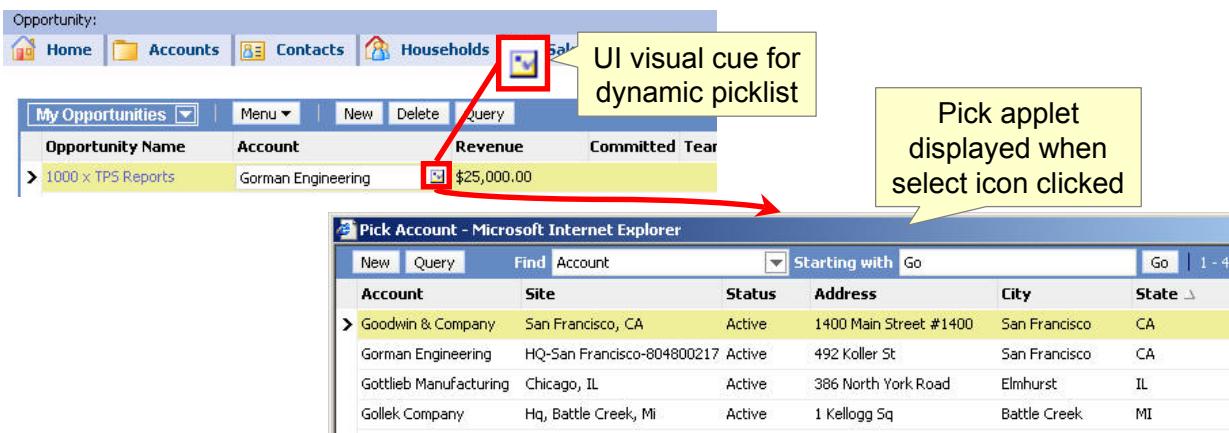
Copyright © 2007, Oracle. All rights reserved.

6 of 22

ORACLE

Dynamic Picklist

- Displays values in a pick applet
- Draws values from a business component (BC) with records edited by users
 - ▶ Values are dynamic and depend on current BC records
- Is used to update joined fields
 - ▶ Copies foreign key reference to the selected value into destination field



Copyright © 2007, Oracle. All rights reserved.

7 of 22



Picklist Terms

- Siebel picklists:
 - ▶ Are associated with a field in the *originating* business component
 - ▶ Draw values from a *pick* business component

Opportunity:

Opportunity Name	Account	Revenue	Committed	Tear
> 1000 x TPS Reports	Gorman Engineering	\$25,000.00		

Opportunity: originating business component

Account: pick business component

Pick Account - Microsoft Internet Explorer

New	Query	Find	Account	Starting with	Go	Go	1 - 4
			Account				
			Goodwin & Company	San Francisco, CA	Active	1400 Main Street #1400	San Francisco CA
			Gorman Engineering	HQ-San Francisco-804800217	Active	492 Koller St	San Francisco CA
			Gottlieb Manufacturing	Chicago, IL	Active	386 North York Road	Elmhurst IL
			Gollek Company	Hq, Battle Creek, Mi	Active	1 Kellogg Sq	Battle Creek MI

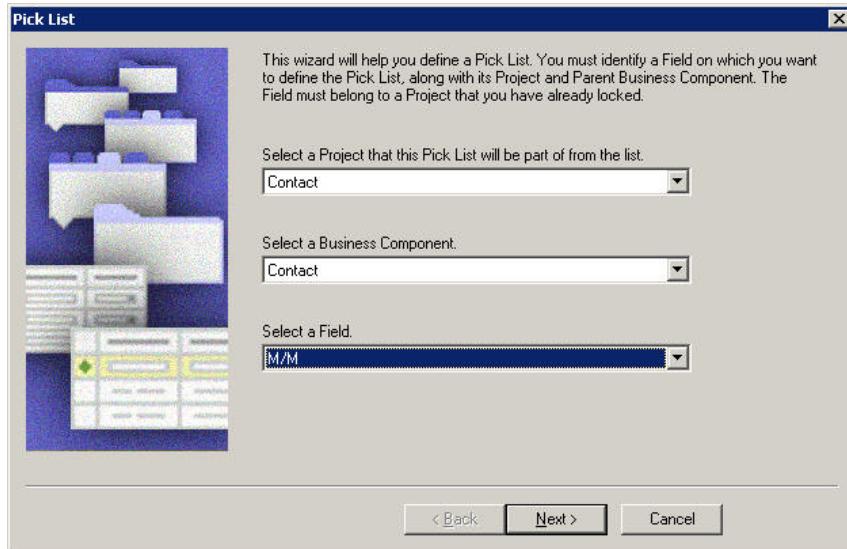
Copyright © 2007, Oracle. All rights reserved.

8 of 22

ORACLE®

Create a New Picklist

- Use the Pick List Wizard to create a new static or dynamic picklist
 - ▶ Select File > New Object > Pick List



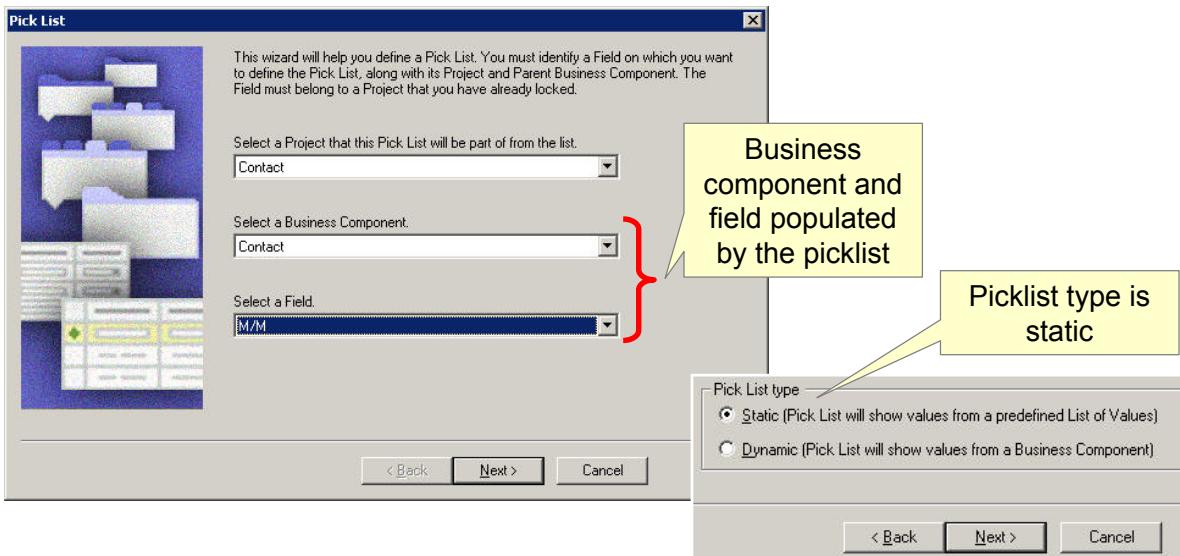
Copyright © 2007, Oracle. All rights reserved.

9 of 22

ORACLE®

Wizard Inputs for a Static Picklist

- The Pick List Wizard requires the following inputs:
 - ▶ Project the created picklist object will be part of
 - ▶ Business component and field populated by the picklist
 - ▶ Picklist name and type (Static)



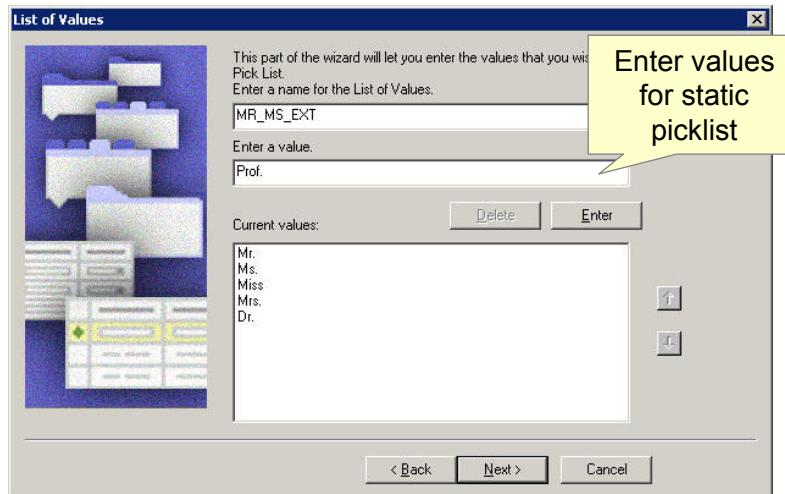
Copyright © 2007, Oracle. All rights reserved.

10 of 22

ORACLE

Wizard Inputs for a Static Picklist Continued

- The Pick List Wizard requires the following inputs (continued):
 - ▶ Bounded or unbounded picklist
 - ▶ Type value (example: MR_MS)
 - ▶ Pick list values
 - May use existing type and/or values in S_LST_OF_VAL



Copyright © 2007, Oracle. All rights reserved.

11 of 22



Pick List Wizard Outputs for a Static Picklist

- The Pick List Wizard creates:
 - ▶ A picklist object
 - ▶ A pick map that specifies values copied to the S_LST_OF_VAL table
 - ▶ Values in S_LST_OF_VAL for the picklist
 - ▶ Activated drop-down list column or control in applets containing the originating field

The screenshot shows three windows from the Siebel interface:

- Picklists**: A list of picklists. One entry is highlighted: "PickList MrMs" (Type: Picklist, Business Component: PickList Generic). A yellow callout labeled "Picklist" points to this entry.
- Pick Maps**: A list of pick maps. One entry is highlighted: "M/M" (Field: M/M, Picklist Field: Value). A yellow callout labeled "Pick map for originating field" points to this entry.
- Drop-down control**: A screenshot of a dropdown menu titled "Mr/Ms". The menu lists "Miss", "Mr.", "Ms.", "Activ Mrs.", and "Dr.". A yellow callout labeled "Drop-down control" points to this menu.

Copyright © 2007, Oracle. All rights reserved.

12 of 22

PickList Generic

Static picklists all share a common pick business component, PickList Generic. This very simple, Siebel-provided business component provides a connection between the originating business component and field and the values that appear in S_LST_OF_VAL. PickList Generic contains fields corresponding to each of this table's columns, such as Type and Value.

Notice in the middle picture above that the pick map is mapping the originating field M/M to the PickList Generic field Value, which links the MR_MS entries in S_LST_OF_VAL with the M/M field of Contact.

ORACLE®

Activating a Drop-Down List Column or Control

- The Pick List Wizard may activate drop-down control for picklist field
 - ▶ The wizard displays all applets displaying the originating field
 - ▶ Only the applets locked by developer will be activated
- Wizard sets list column or control Runtime property to TRUE
 - ▶ A drop-down arrow appears as a cue

The screenshot shows the Siebel Business Components interface. On the left, there are two panels: 'Applets' and 'Controls'. The 'Applets' panel lists 'Contact Form Applet' under the 'Business Component' column. The 'Controls' panel lists several fields with their names, runtime status (indicated by a checkmark), and captions. A red box highlights the row for 'M/M' (Runtime checked, Mr/Ms caption). To the right, a detailed view of a contact form for 'Shashi Aamot' is shown. The form includes fields for Last Name, First Name, Middle Initial, Mr/Ms (with a dropdown menu showing 'Mr.' selected), and Job Title. A yellow callout bubble points to the 'Mr/Ms' dropdown with the text 'Runtime set to TRUE'.

	Name	Business Component
>	Contact Form Applet	Contact

	Name	Runtime	Caption
>	M/M	✓	Mr/Ms
	MailStop		Mail Stop
	ManagerFirstName		Manager First Name
	ManagerLastName	✓	Manager Last Name
	MiddleName		Middle Initial

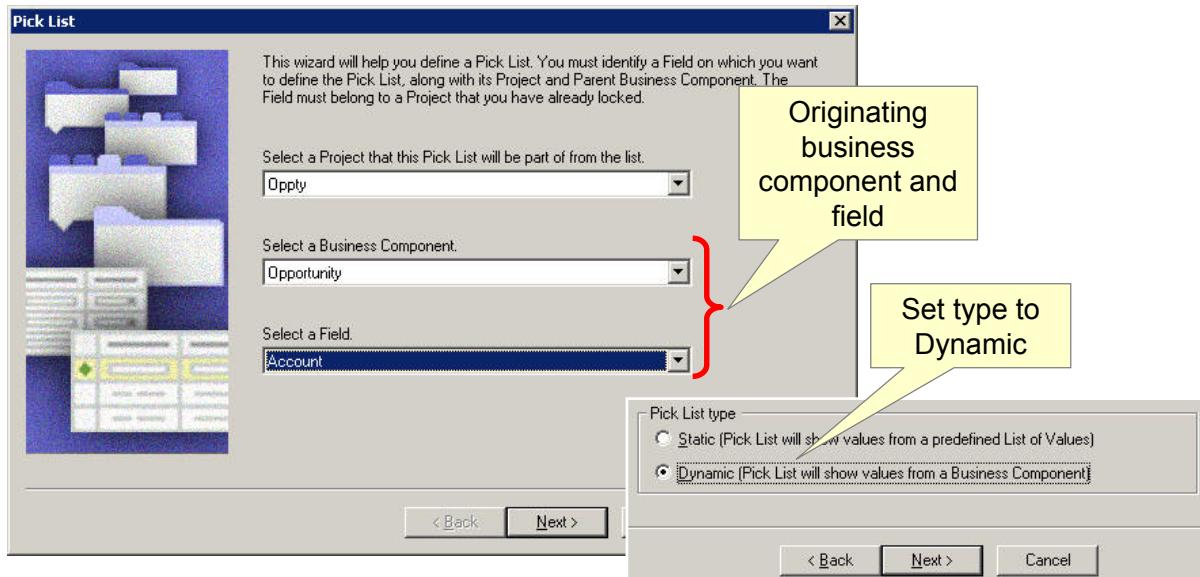
Copyright © 2007, Oracle. All rights reserved.

13 of 22

ORACLE®

Wizard Inputs for Dynamic Picklists

- The Pick List Wizard requires the following inputs:
 - ▶ Project the created picklist object will be part of
 - ▶ Originating business component and field for the picklist
 - ▶ Picklist name and type (Dynamic)



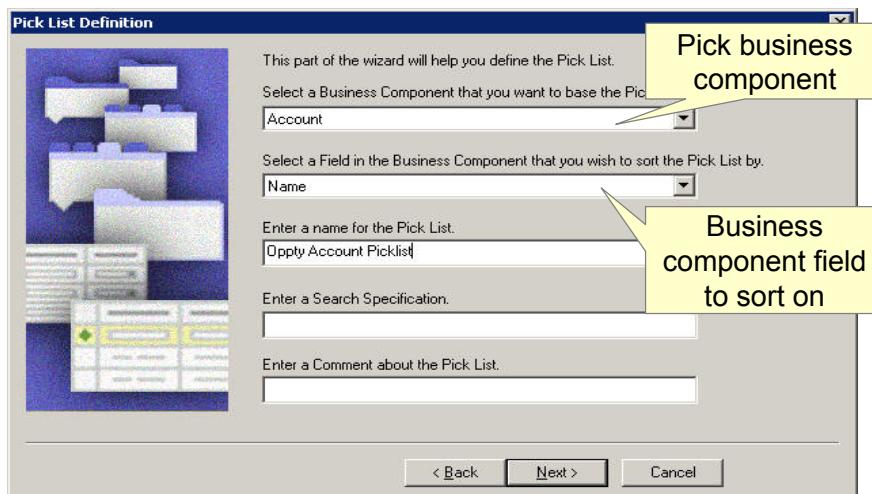
Copyright © 2007, Oracle. All rights reserved.

14 of 22

ORACLE®

Wizard Inputs for Dynamic Picklists continued

- The Pick List Wizard requires the following inputs (continued):
 - ▶ A pick business component
 - ▶ A field in the pick business component to sort on
 - ▶ A picklist name
 - ▶ A search specification



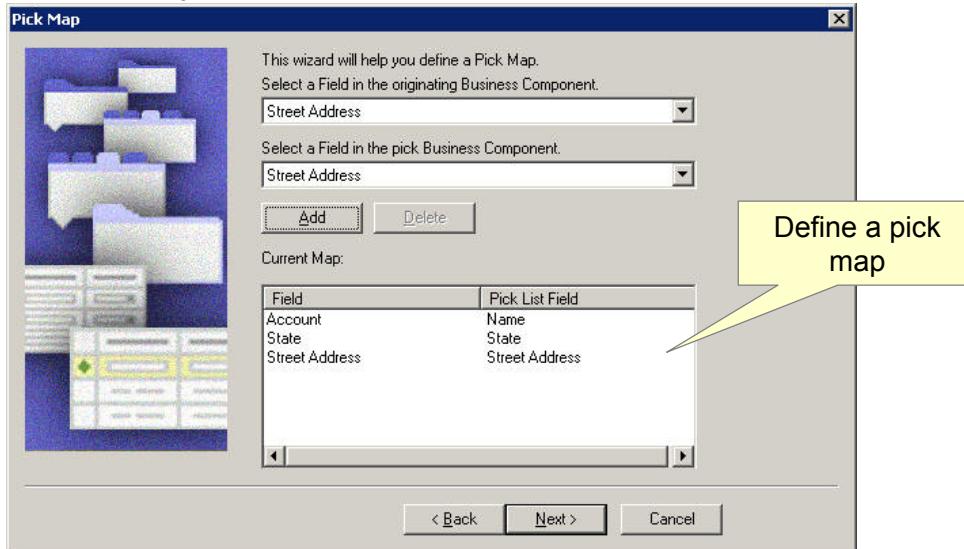
Copyright © 2007, Oracle. All rights reserved.

15 of 22

ORACLE®

Wizard Inputs for Dynamic Picklists Continued

- The Pick List Wizard requires the following inputs (continued):
 - ▶ Pick applet properties (No Delete, No Insert, No Update, No Merge)
 - ▶ Values for a pick map, mapping fields between originating and pick business components



Copyright © 2007, Oracle. All rights reserved.

16 of 22

ORACLE®

Pick List Wizard Outputs for a Dynamic Picklist

- The Pick List Wizard creates:
 - ▶ A picklist object
 - ▶ A pick map constructed in the wizard
 - ▶ A new pick applet, if needed
 - ▶ Activated list column or control in applets containing the originating field
 - List column or control for originating field must have Runtime property set to TRUE

The screenshot shows two tables from the Siebel Business Layer Configuration interface.

Picklists:

Name	Project	Changed	Bounded	Business Component	Icon
Oppty Account Picklist	Oppty	✓		Account	

A yellow callout labeled "Picklist" points to the "Account" icon in the last column.

Pick Maps:

Field	Changed	Constrain	No Clear	Picklist Field
Account	✓			Name
State	✓			State
Street Address	✓			Street Address

A yellow callout labeled "Pick map for originating field" points to the "Name" entry under the "Picklist Field" column.

Below the table, there is a form field labeled "Account:" with a red box around it and a small "x" icon to its right. A yellow callout labeled "Field with select icon" points to this field.

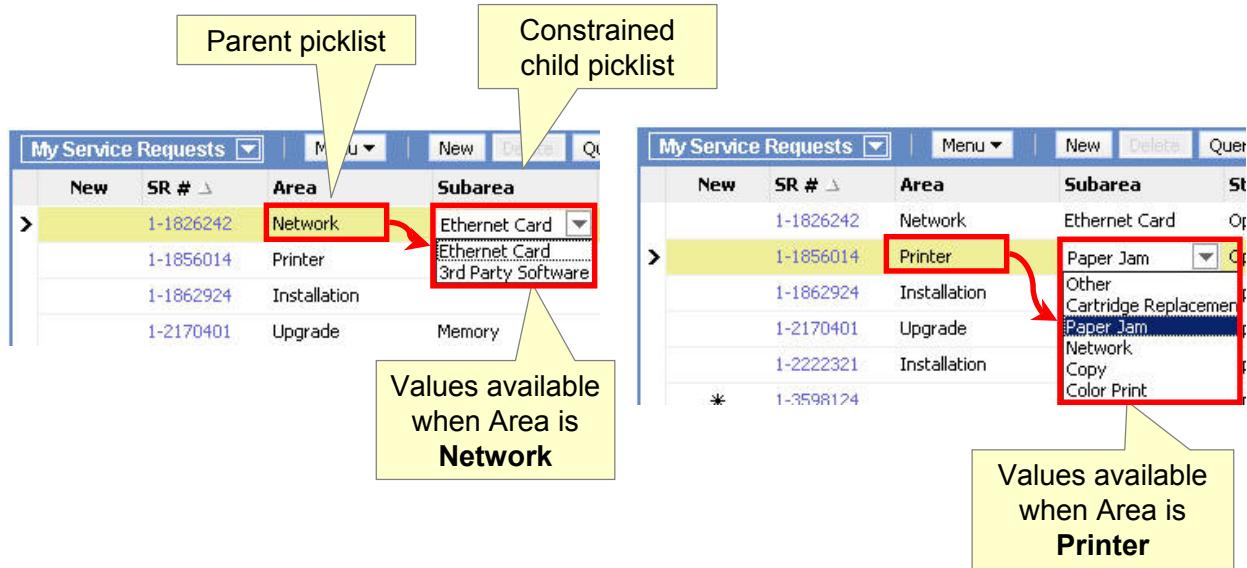
Copyright © 2007, Oracle. All rights reserved.

17 of 22

ORACLE®

Constrained Picklist

- Filters values dynamically based on value in parent picklist



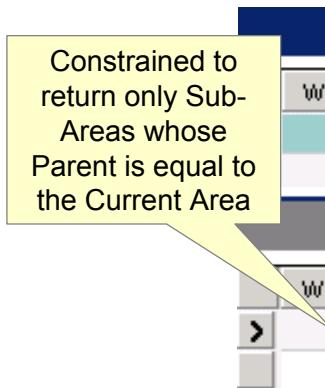
Copyright © 2007, Oracle. All rights reserved.

18 of 22

ORACLE®

Constraining a Picklist

- Create a pick map definition for each field that must match
- Set the Constrain property to TRUE for each of the matching fields
 - ▶ Filters the pick business component records for matches
 - ▶ Does not copy values for the field



Single Value Fields			
W	Name	Column	
	Sub-Area	SR_SUB_AREA	

SVF Pick Maps			
W	Field	Constrain	Picklist Field
>	CurrentArea	✓	Parent
	Sub-Area		Value



Module Highlights

- Picklists allow selection of values from a list for one or more single-value fields in an originating business component
- Static picklists:
 - ▶ Display values in a drop-down list
 - ▶ Contain static values, which are managed through List of Values administrative views
 - ▶ Store values in S_LST_OF_VAL table
- Dynamic picklists:
 - ▶ Display values in a pick applet
 - ▶ Contain dynamic data, which is typically the result of user transactions
 - ▶ Access data in pick business component using a foreign key
- Picklists are created using Siebel Tools' Pick List Wizard



Lab

- In the lab you will:
 - ▶ Create a new static picklist
 - ▶ Create a new dynamic picklist

Copyright © 2007, Oracle. All rights reserved.

21 of 22

ORACLE®

Siebel 8.0 Essentials

Module 32: Configuring Multi-Value Groups

32

Copyright © 2007, Oracle. All rights reserved.





Module Objectives

- After completing this module you should be able to:
 - ▶ Describe multi-value groups and their benefits
 - ▶ Use Siebel Tools to configure a multi-value group
- Why you need to know:
 - ▶ Enables you to incorporate child data directly in an applet
 - More child data can be available within a view
 - More effective use of screen space

ORACLE®

Multi-Value Group

- A multi-value group (MVG) is a set of detail (child) records associated with a parent record
 - ▶ The Parent applet displays only one of the child records
 - ▶ The MVG applet opens on demand to display all child records

The screenshot shows a Siebel application interface. At the top, there's a navigation bar with links like Home, Accounts, Contacts, Households, Sales Orders, Service, Assets, and Service Orders. Below it is a toolbar with buttons for My Accounts, Menu, New, Delete, Query, Collaborate, and Create Team Space. The main area displays account details for 'Knoll Pharmaceutical Co' (Edison, NJ, phone: (973) 426-2600) with an 'Industries' field containing 'drugs, proprietaries & sundries'. A red box highlights this field. A callout bubble labeled 'MVG applet' points to a small window titled 'Industries - Microsoft Internet Explorer' which is overlaid on the main screen. This window shows a list of industries under 'Available' and 'Selected' categories. The 'Available' list includes items like U.S. Postal Service, X-ray apparatus & tubes, abrasive products, accident & health insurance, accounting, auditing & bookkeepir, adhesives & sealants, adjustment & collection services, administration of educational prog, and administration of general economi. The 'Selected' list contains 'drugs, proprietaries & sundries' with a checked checkbox. Red arrows point from the 'Selected' list back to the 'Industries' field in the main account view.

Copyright © 2007, Oracle. All rights reserved.

3 of 22

Reference

Configuring Siebel Business Applications: Configuring Multi-Value Group and Association Applets

ORACLE®

Multi-Value Group Continued

- Is an alternative to a detail view for managing parent and related child records
 - ▶ Makes effective use of space
 - Does not require dedicated space on a view
 - ▶ Allows for multiple sets of detail records to be available from a single view

Knoll Pharmaceutical Co

Menu New Delete Query

Account Name: * Knoll Pharmaceutical Co Site: Edison, NJ Account Team: CCHENG Status: Current Customer

Address: 3000 Continental Dr. Address Line 2: Main Phone #: (973) 426-2100 Account Type: Commercial

City: Budd Lake State: NJ Main Fax #: Country: USA URL: www.knolph.

Zip Code: 07828-1202 Territory: Industries: drugs, proprietaries

Business Address MVG

Account Team MVG

Territory MVG

Industries MVG

Copyright © 2007, Oracle. All rights reserved.

4 of 22

ORACLE®

Multi-Value Group Continued

- Allows users to access child records for multiple parent records in a single list view

The diagram illustrates the concept of Multi-Value Groups (MVG). Three yellow callout boxes labeled "Business Address MVG", "Territory MVG", and "Industries MVG" point to a single list view of accounts. The list view shows columns for Account Name, Site, Address Line 1, Territories, and Industries. Each account row contains a link to its details, such as "Marriott International HQ" which links to "10400 Fernwood Road". The Territories and Industries columns show the categories associated with each account.

Account Name	Site	Address Line 1	Territories	Industries
Marriott International HQ	HQ	10400 Fernwood Road		hotels & motels
AT&T	Edison, NJ	350 Main Street		communications equipment
Ace Properties	Headquarters	100 El Sabino Road		investment offices
Akamai Technologies, Inc.	Cambridge, MA	118 Turnpike Rd		computer related services
Alliance Program		1855 S. Grant St		
American General Corporation	Houston, TX	2929 Allen Parkway		
Andrews Manufacturing	Fresno	5807 Lampas Verde		automobiles & other motor vehicles
British American Tobacco	Hamburg, Germany	Alsterufer 4		tobacco & tobacco products

Copyright © 2007, Oracle. All rights reserved.

5 of 22

ORACLE®

Multi-Value Group Continued

- Allows for creating queries that include values for fields in both parent and child records

The screenshot shows the 'Enter Query' dialog box from the Siebel interface. It displays various query parameters grouped into sections:

- Parent:** A dropdown menu.
- Parent Site:** A dropdown menu.
- Partner:** A checkbox.
- Competitor:** A checkbox.
- DUNS #:** A dropdown menu.
- Domestic Ultimate DUNS:** A dropdown menu.
- Parent/HQ DUNS:** A dropdown menu.
- Global Ultimate DUNS:** A dropdown menu.
- Assignment Area Code:** A dropdown menu.
- Assignment Country Code:** A dropdown menu.
- Account Type:** A dropdown menu.
- Organization:** A dropdown menu.
- Industries:** A dropdown menu.
- Synonyms:** A dropdown menu.
- Region:** A dropdown menu.
- Territory:** A dropdown menu.
- Status:** A dropdown menu.
- Stage:** A dropdown menu.
- Expertise:** A dropdown menu.

Three specific dropdown menus are highlighted with yellow callout boxes:

- Organization:** Labeled "Query on Organization".
- Industries:** Labeled "Query on Industries".
- Territory:** Labeled "Query on Territory".

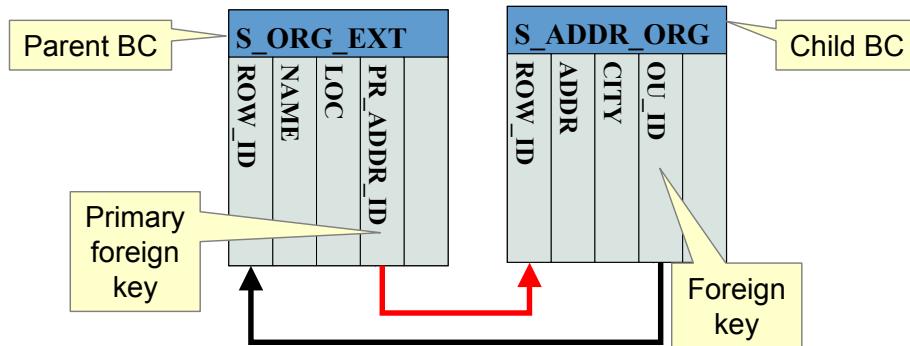
Copyright © 2007, Oracle. All rights reserved.

6 of 22



Primaries

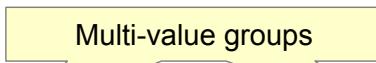
- Primary key allows fast retrieval of the primary child record for display in the parent applet
 - ▶ A primary refers to a designated child record
 - ▶ Designating a primary allows one child record to be retrieved quickly for display
- Supported by a primary foreign key field in the parent business component to reference the child's primary key (ROW_ID)
 - ▶ Siebel Data Model includes many primary foreign keys



ORACLE®

Primaries Performance Example

- Three MVGs on a list applet displaying 10 parent records requires 31 queries to populate without primaries
 - ▶ One query to populate parent fields in list applet
 - ▶ 30 queries (three per parent record) to populate the MVGs



Account Name	Address Line 1	City	State	Account Team	Industries
Akamai Technologies, Inc.	118 Turnpike Rd	Southborough	MA	CCHENG	computer related services
Andrews Manufacturing	5807 Lampos Verde	Fresno	CA	CCHENG	automobiles & other motor vehicles
British American Tobacco	Alsterufer 4	Hamburg		CCHENG	tobacco & tobacco products
Cap Gemini Ernst & Young	2727 Paces Ferry Rd SE Bldg	Atlanta	GA	CCHENG	services
Caterpillar	1550 N Milwaukee Ave	Chicago	IL	CCHENG	engine electrical equipment
Chase Manhattan Bank	95 Wall St Lbby 3rd	New York	NY	CCHENG	bank holding companies
Country Companies Services	1711 G E Rd	Bloomington	IL	CCHENG	insurance carriers
Cymer Inc.	3457 South Van Ness Avenue	San Francisco	CA	CCHENG	computer peripheral equipment
Danney K. Foundation	100 Beecham Dr	Pittsburgh	PA	CCHENG	administration of general economic programs
FleetBoston Financial	205 Newbury St	Framingham	MA	CCHENG	commercial banks

Copyright © 2007, Oracle. All rights reserved.

8 of 22

ORACLE®

Primaries Performance Example Continued

- Using primaries, one query will populate all the fields on the list applet
 - ▶ A query using SQL joins populates each MVG with its primary child record

Multi-value groups

Account Name	Address Line 1	City	State	Account Team	Industries
Akamai Technologies, Inc.	118 Turnpike Rd	Southborough	MA	CCHENG	computer related services
Andrews Manufacturing	5807 Lampos Verde	Fresno	CA	CCHENG	automobiles & other motor vehicles
British American Tobacco	Alsterufer 4	Hamburg		CCHENG	tobacco & tobacco products
Cap Gemini Ernst & Young	2727 Paces Ferry Rd SE Bldg	Atlanta	GA	CCHENG	services
Caterpillar	1550 N Milwaukee Ave	Chicago	IL	CCHENG	engine electrical equipment
Chase Manhattan Bank	95 Wall St Lbby 3rd	New York	NY	CCHENG	bank holding companies
Country Companies Services	1711 G E Rd	Bloomington	IL	CCHENG	insurance carriers
Cymer Inc.	3457 South Van Ness Avenue	San Francisco	CA	CCHENG	computer peripheral equipment
Danney K. Foundation	100 Beecham Dr	Pittsburgh	PA	CCHENG	administration of general economic programs
FleetBoston Financial	205 Newbury St	Framingham	MA	CCHENG	commercial banks

Copyright © 2007, Oracle. All rights reserved.

9 of 22

0/3



Object Types Supporting MVGs

- MVGs are implemented in Siebel applications using the following object types:

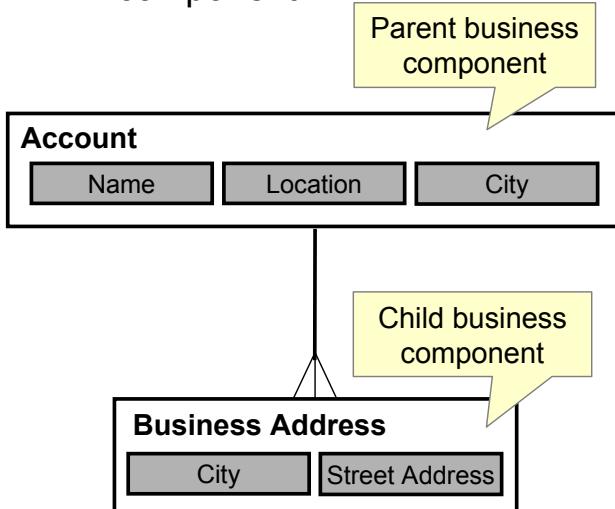
Link

Multi-Value Field (MVF)

MVG Applet

Link

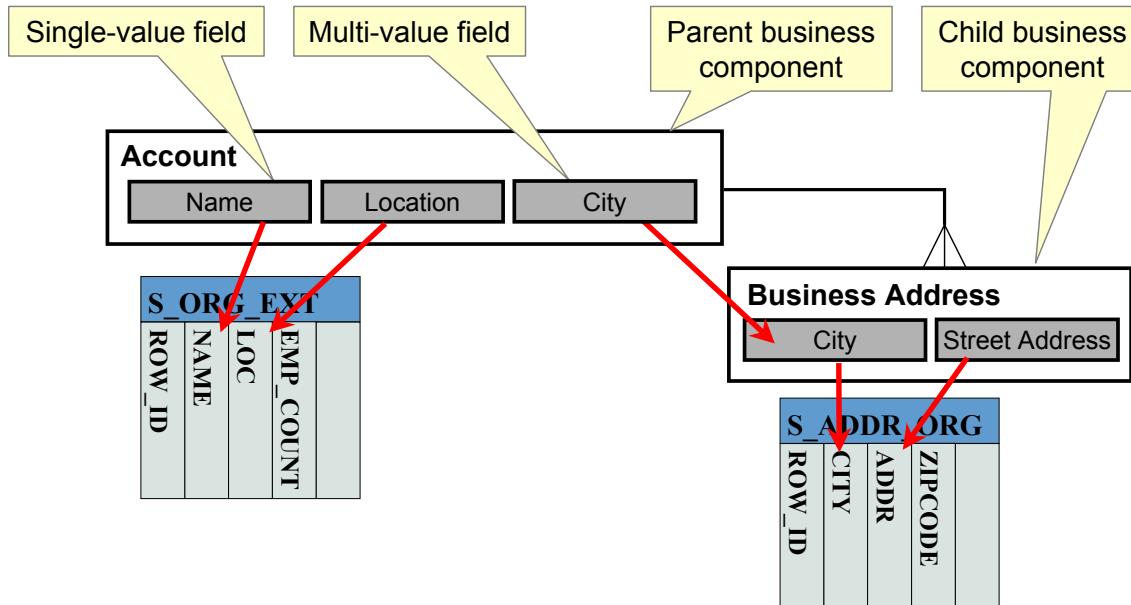
- Must exist between parent and child business components in the MVG
 - ▶ Specifies how to get data from child business component



Properties	
Link [Account/Business Address]	
Alphabetic Categorized	
Cascade Delete	Delete
Child Business Component	Business Address
Comments	
Destination Field	Account Id
Inactive	FALSE
Inter Child Column	
Inter Child Delete	FALSE
Inter Parent Column	
Inter Table	
Module	
Name	Account/Business Address
No Associate	FALSE
No Delete	FALSE
No Insert	FALSE
No Inter Delete	FALSE
No Update	FALSE
Object Language Locked	
Object Locked	FALSE
Object Locked By Name	
Object Locked Date	
Parent Business Component	Account
Primary Id Field	Primary Address Id
Search Specification	

Multi-Value Field (MVF)

- Is a field in the parent business component that references a field in the child business component
- Is required for an MVG



Copyright © 2007, Oracle. All rights reserved.

12 of 22

MVG Applet

- Displays child business component records
 - ▶ Displays list columns that allow the user to distinguish the desired record
- MVG applets' type is MVG

W	Name	Changed	Type	Project	Business Component
	Account - Oracle 10.7 Credit Profile MVG Appl		MVG	Oracle Account 10.7	Account Credit Profile
	Account - Oracle 11i Credit Profile MVG Apple		MVG	Oracle Account 11i	Account Credit Profile
>	Account Address Mvg Applet		MVG	Account (SSE)	Business Address
	Account Credit Profile MVG Applet		MVG	Oracle Account 10.7	Account Credit Profile
	Account Mvg Applet		MVG	Account (SSE)	Account
	Account Mvg Applet (Delegated Admin)		MVG	Admin	Account (Delegated Admin)
	Account Mvg Applet - No Primary		MVG	Admin (SCW)	Account

Copyright © 2007, Oracle. All rights reserved.

13 of 22

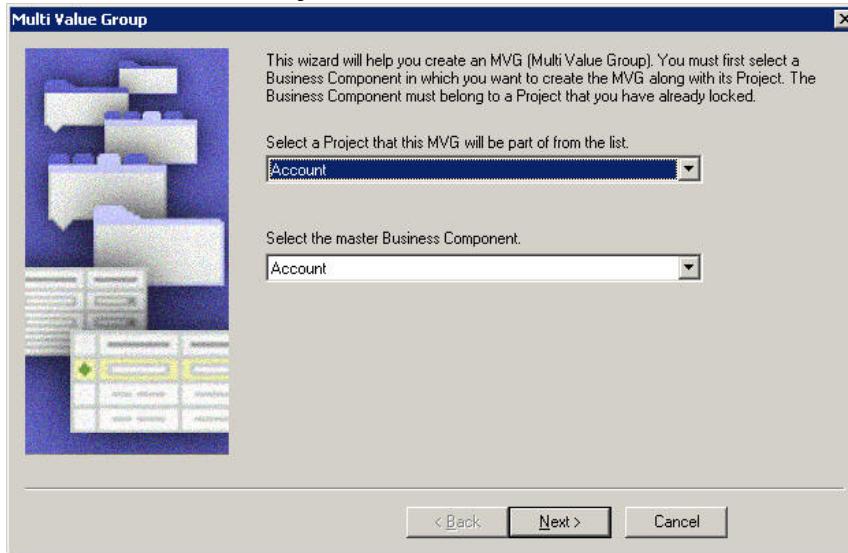
MVG Applet

There is no requirement that the MVG applet display only those multi-value fields that are displayed on the parent applet. This allows an MVG applet to be used in other MVGs throughout the application.

ORACLE®

Creating a Multi-Value Group

- Use the MVG Wizard in Siebel Tools to create a multi-value group
 - ▶ Verify that a link between the parent and child business components exists
 - ▶ Select File > New Object > MVG



Copyright © 2007, Oracle. All rights reserved.

14 of 22



Inputs to the MVG Wizard

- Provide the following information to the MVG Wizard:
 - ▶ The project that the MVG will belong to
 - ▶ The master (parent) and detail (child) business components
 - ▶ The name of the multi-value link
 - ▶ The Auto Primary property (discussed later)
 - ▶ The link properties

Copyright © 2007, Oracle. All rights reserved.

15 of 22

Link Properties

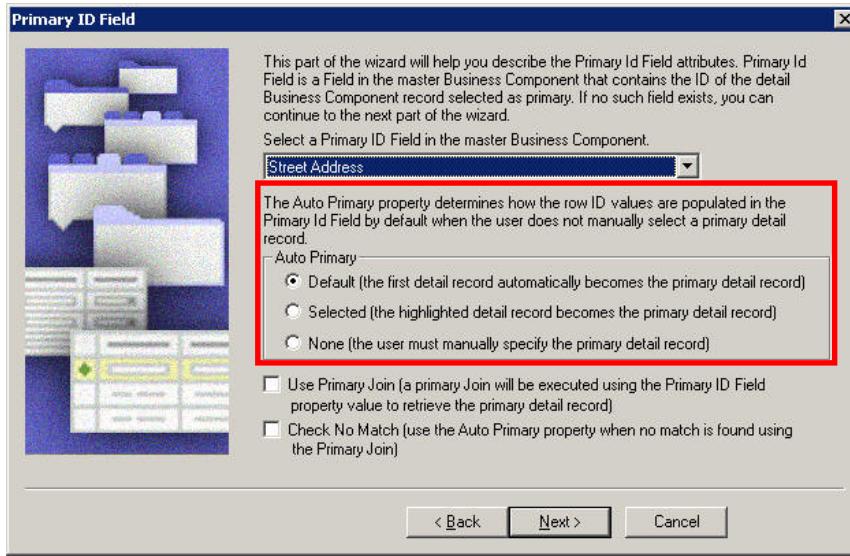
The following properties can be set to TRUE or FALSE for a link: No Associate, No Copy, No Delete, No Insert, No Update. No Associate equals TRUE specifies that no new associations can be created through the link. New child records may be added.



ORACLE®

Setting Auto Primary Property in the MVG Wizard

- Set the Auto Primary property to either:
 - ▶ Default (the first child record becomes the primary)
 - ▶ Selected (the highlighted record becomes the primary)
 - ▶ None (the user must specify the primary)



Copyright © 2007, Oracle. All rights reserved.

16 of 22



MVG Wizard Output

- The MVG Wizard creates an MVG
- If no suitable MVG applet exists suitable for the choice of parent and child business components and MVFs, the wizard will open the MVG Applet Wizard

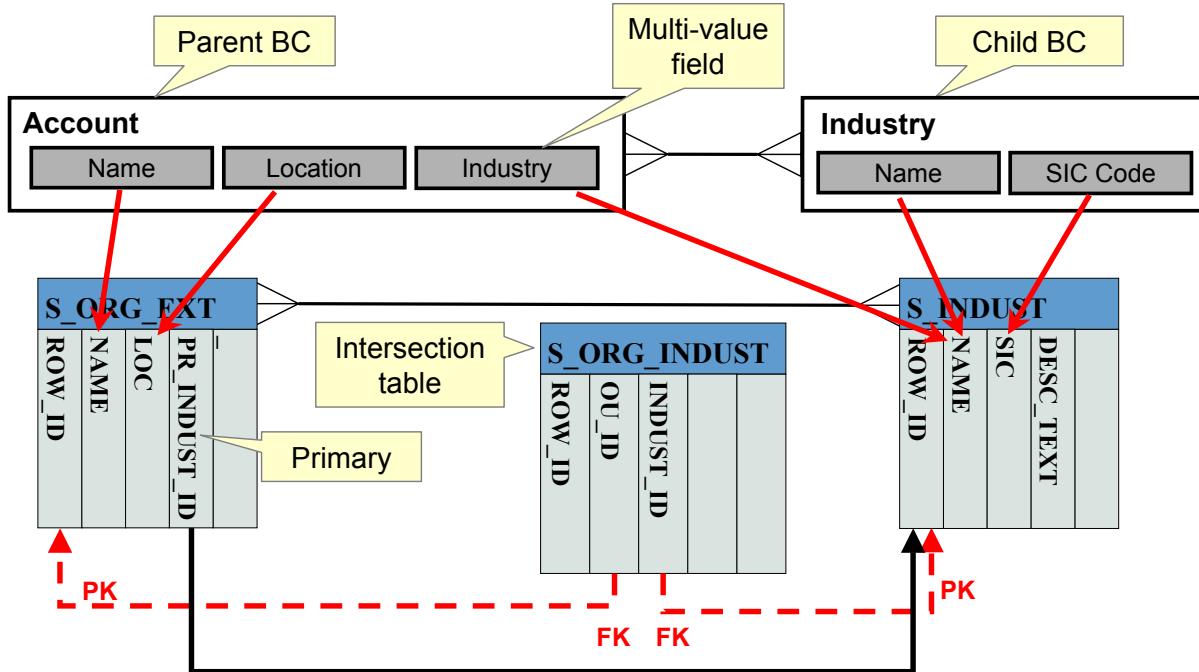
Copyright © 2007, Oracle. All rights reserved.

17 of 22

ORACLE®

Multi-Value Groups for M:M Relationships

- MVGs can be built on M:M relationships as well
 - Link must specify the Inter Table property



Copyright © 2007, Oracle. All rights reserved.

18 of 22

ORACLE®

Configuring MVGs for M:M Relationships

- Create object definitions required for 1:M MVGs
- Specify an association applet for the MVG applet
 - ▶ Allows users to select other child records to add to the MVG

The screenshot illustrates the configuration of an Association applet for a Multi-Value Group (MVG). At the top, a table titled "Applets" shows the "Industry Mvg Applet" entry, which is identified as an "MVG" and associated with the "Industry" Business Component. The "Associate Applet" field contains the value "Industry Assoc Applet", which is highlighted with a red box and has a red arrow pointing from it to the "Available" list in the Siebel interface below.

The Siebel interface displays two lists: "Available" and "Selected".

Available		
Industry Name	Type	SIC Code
U.S. Postal Service	4-digit SIC	4311
X-ray apparatus & tr	4-digit SIC	3844
abrasive products	4-digit SIC	3291
accident & health ins	4-digit SIC	6321
accounting, auditing	4-digit SIC	8721

Selected

Primary	Industry Name	SIC Code
	computer peripheral	3577
	electrical equipment	3699
> ✓	manufacturing indus	3999

Buttons between the lists include "Add >" and "< Remove".

Copyright © 2007, Oracle. All rights reserved. 19 of 22



Module Highlights

- A multi-value group (MVG) is a set of detail (child) records associated with a parent record
- A primary is a designated child record that is displayed in the parent's applet
 - ▶ Speeds retrieval of parent record with primary child record for display in UI
 - ▶ Supported by a primary foreign key
- Objects needed to support an MVG are:
 - ▶ Multi-value fields (MVF)
 - ▶ A multi-value link
 - ▶ A MVG applet
- Use the MVG Wizard to create an MVG in Siebel Tools



Lab

- In the lab you will:
 - ▶ Create a new MVG
 - ▶ Create an MVG applet to display the MVG

Copyright © 2007, Oracle. All rights reserved.

21 of 22



Siebel 8.0 Essentials

Module 33: Data Layer Configuration

33

Copyright © 2007, Oracle. All rights reserved.

33

Oracle Internal & Oracle Academy Use Only



Module Objectives

- After completing this module you should be able to:
 - ▶ Create extension columns in a table
 - ▶ Create custom tables
 - Standalone table
 - 1:1 extension table
 - 1:M extension table
 - Intersection table
- Why you need to know:
 - ▶ Enables you to incorporate additional attributes and business entities into a Siebel application



Incorporating Additional Data

- Your business requirements may include:
 - ▶ Adding new fields to capture additional data
 - ▶ Creating new business components to capture additional business entities
- Extending the Siebel database can satisfy these requirements
 - ▶ Adding one or more columns to an existing table
 - ▶ Creating new database tables to support new business components

0/2



Evaluate the Existing Database Tables

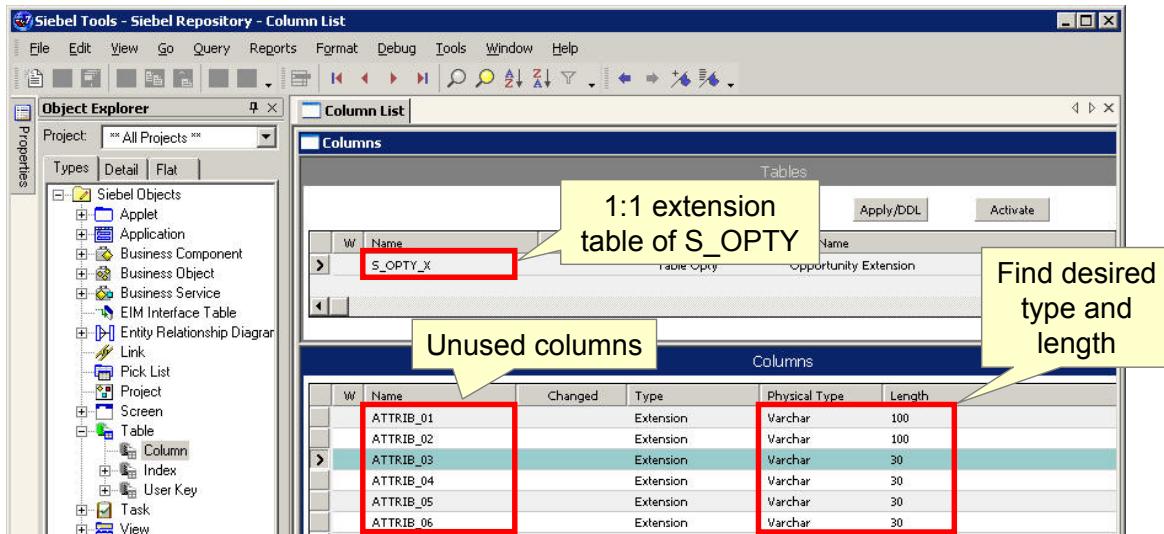
- Prior to extending the Siebel database, consider using:

Unused Columns in an Existing Table

Existing 1:M Extension Table

Unused Columns in an Existing Table

- Consider mapping new fields to unused columns in an existing base or 1:1 extension table
 - ▶ Verify that the candidate column has the desired type
 - ▶ Consider possible upgrade conflicts in future releases
 - ▶ Consider performance impact of a join to the 1:1 extension table

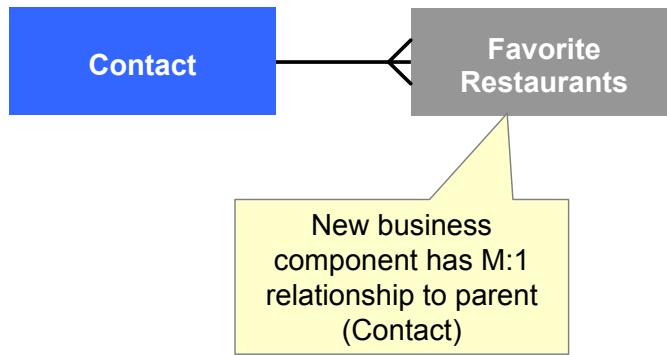


Copyright © 2007, Oracle. All rights reserved.

5 of 24

Existing 1:M Extension Tables

- Consider mapping the new business component to an appropriate 1:M extension table (as discussed in an earlier module)
 - ▶ Verify that the business component has the correct M:1 relationship to the candidate parent business component





Extending the Database

- Changes the database schema and requires propagating the changes to:
 - ▶ Other developers during development
 - ▶ Mobile users after development for applications in production
 - ▶ Production enterprise at completion of development and testing
- Requires creating additional object definitions to:
 - ▶ Map columns in the EIM tables needed to import and export data to the extension columns and tables
 - ▶ Specify how data for these extension columns and tables are to be routed to remote users (Dock Objects)



Using Siebel Tools to Extend the Database

- Supports creating new:
 - ▶ Extension columns on tables
 - ▶ Standalone tables
 - ▶ 1:1 extension tables
 - ▶ 1:M extension tables
 - ▶ Intersection tables
- Creates new object definitions for the database extension
 - ▶ Invokes a wizard to build new tables
- Makes the corresponding physical database changes
 - ▶ Developers do not create, use, or maintain SQL scripts



Best Practices for Extending the Siebel Database

- General suggestions:

- ▶ When adding a new column:
 - If column is not often populated, use an existing column in an extension table if available
 - If field often populated, avoid join overhead by using adding an extension column to base table
- ▶ If data for column appears in:
 - A form applet, prefer an existing column in an extension table
 - A list applet, prefer an extension column in the base table
 - ▶ Join overhead can be significant when displaying many records

Copyright © 2007, Oracle. All rights reserved.

9 of 24

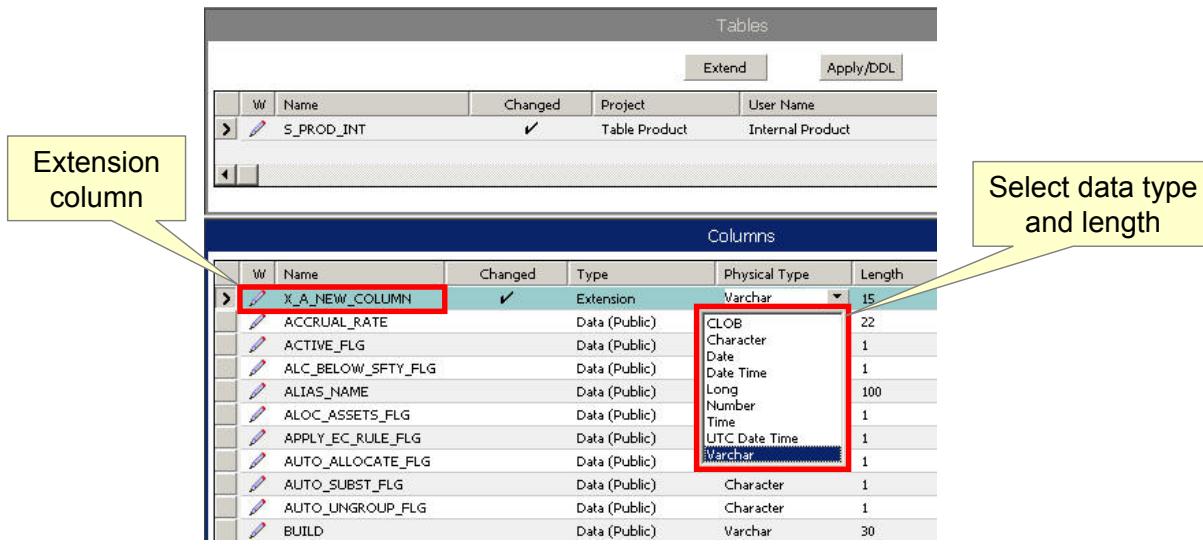
Reference

Siebel Bookshelf, Configuring Siebel Business Applications, "Configuring Tables and Columns"

ORACLE®

Creating a Custom Extension Column

- Make logical changes to the Data layer
 - ▶ Check out the appropriate project
 - ▶ Select the table to be extended
 - ▶ Create a new column record with the desired properties
 - Name automatically prefixed with X_



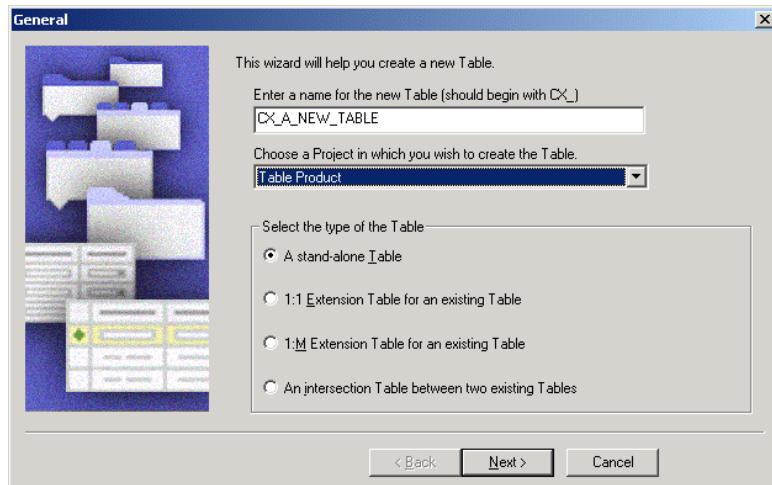
Copyright © 2007, Oracle. All rights reserved.

10 of 24

ORACLE®

Creating a New Table

- Use the Table Wizard to create a new table
 - ▶ Select File > New Object > Table
- Four types of table can be created:
 - ▶ Standalone table
 - ▶ 1:1 extension table
 - ▶ 1:M extension table
 - ▶ Intersection table



Copyright © 2007, Oracle. All rights reserved.

11 of 24

Table Name

A table name must start with CX_ and cannot exceed 15 characters.

ORACLE®

Creating a Standalone Table

- Table wizard creates a standalone table with:
 - ▶ Data (Public) as its type
 - ▶ Nine system columns
 - ▶ One index P1 on ROW_ID

Tables

	W	Name	Changed	Project	User Name	Alias
>	<input checked="" type="checkbox"/>	CX_NEW_SA_TABLE	✓	Table Product	CX_NEW_SA_TABLE	
<input type="button" value="◀"/> <input type="button" value="▶"/>						

Columns

	W	Name	Changed	User Name	Alias	Type	Primary Key
>	<input checked="" type="checkbox"/>	CONFLICT_ID	✓	Conflict Id		System	
	<input checked="" type="checkbox"/>	CREATED	✓	Created		System	
	<input checked="" type="checkbox"/>	CREATED_BY	✓	Created By		System	
	<input checked="" type="checkbox"/>	DB_LAST_UPD	✓	DB Last Updated		System	
	<input checked="" type="checkbox"/>	DB_LAST_UPD_SRC	✓	DB Last Updated By		System	
	<input checked="" type="checkbox"/>	LAST_UPD	✓	Last Updated		System	
	<input checked="" type="checkbox"/>	LAST_UPD_BY	✓	Last Updated By		System	
	<input checked="" type="checkbox"/>	MODIFICATION_NUM	✓	Modification Number		System	
	<input checked="" type="checkbox"/>	ROW_ID	✓	Row Id		System	

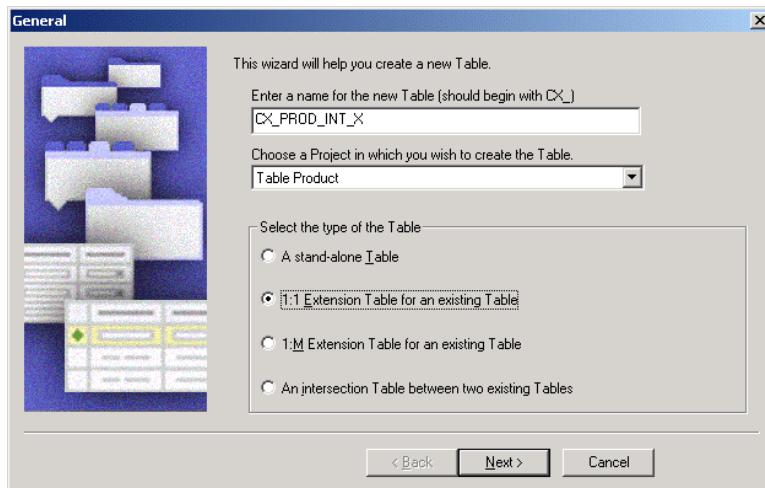
Copyright © 2007, Oracle. All rights reserved.

12 of 24

ORACLE®

Creating a 1:1 Extension Table

- Provide a base table as input to the Table Wizard
 - ▶ Choice restricted to the Data (Public) type
 - ▶ Multiple extension tables relate directly to the base table, and not to each other



Copyright © 2007, Oracle. All rights reserved.

13 of 24

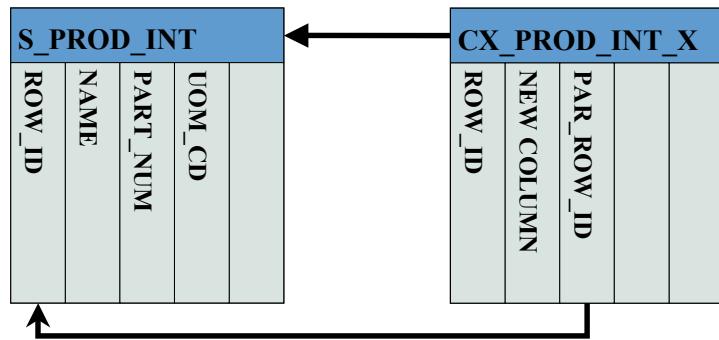
Extension Tables

1:1 extension tables cannot be created for tables such as S_ORG_EXT and S_CONTACT, which are already extension tables of S_PARTY. Rather, create the new table as an extension table of S_PARTY.

ORACLE®

Creating a 1:1 Extension Table Continued

- The Table Wizard creates an extension table with:
 - ▶ Ten system columns
 - Required nine system columns
 - PAR_ROW_ID column as the foreign key column to the base table
 - ▶ Two indexes:
 - P1 index on ROW_ID
 - U1 index on PAR_ROW_ID and CONFLICT_ID



Copyright © 2007, Oracle. All rights reserved.

14 of 24

Indexes

Indexes allow fast retrieval and sorting of records using one or more designated columns. Inserts and updates on indexed columns are costlier than on un-indexed ones.

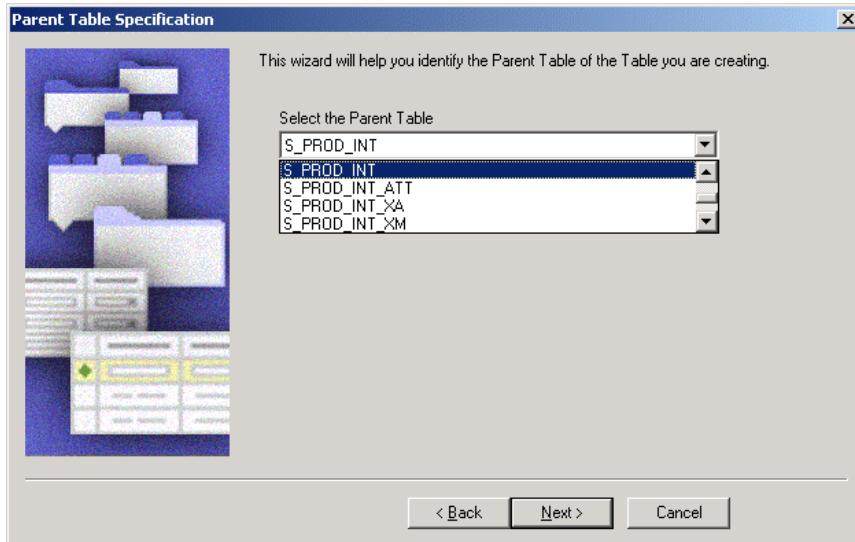
CONFLICT_ID

Is used by Siebel Remote to resolve synchronization conflicts between remote clients. See Siebel Bookshelf, Remote and Replication Manager Administration Guide for further details.

ORACLE®

Creating a 1:M Extension Table

- Create if the parent table does not have an existing 1:M table
- Provide a parent table as input to the Table Wizard
 - ▶ Choice restricted to the Data (Public) type



Copyright © 2007, Oracle. All rights reserved.

15 of 24



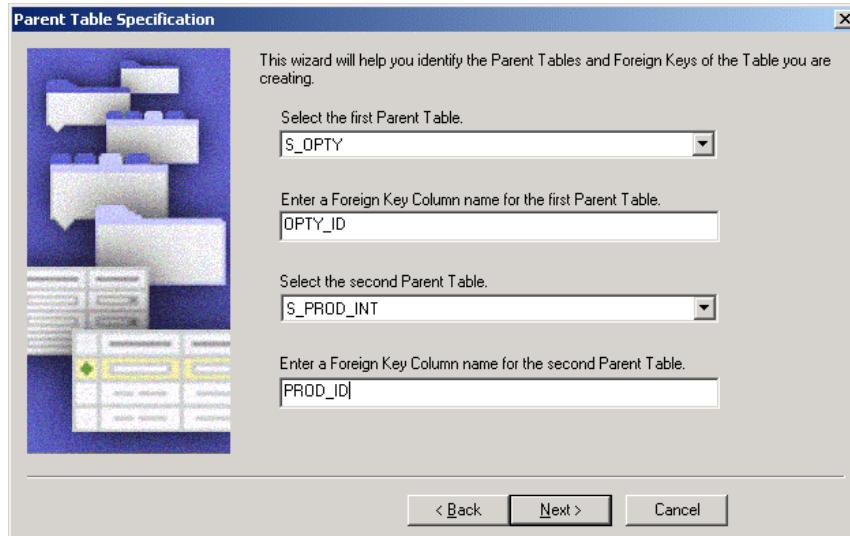
Creating a 1:M Extension Table Continued

- Table wizard creates an 1:M extension table with:
 - ▶ Data (Public) as its type
 - ▶ Ten system columns
 - Nine required columns
 - PAR_ROW_ID column as the foreign key column to the base table
 - ▶ TYPE and NAME columns
 - ▶ Three indexes:
 - P1 index on ROW_ID
 - U1 index on PAR_ROW_ID, TYPE, NAME, and CONFLICT_ID
 - M1 index on TYPE and NAME

ORACLE®

Creating an Intersection Table

- Select both parent tables
 - ▶ Choices restricted to the Data (Public) type
- Specify the foreign key column name for each parent table



Copyright © 2007, Oracle. All rights reserved.

17 of 24



Creating an Intersection Table continued

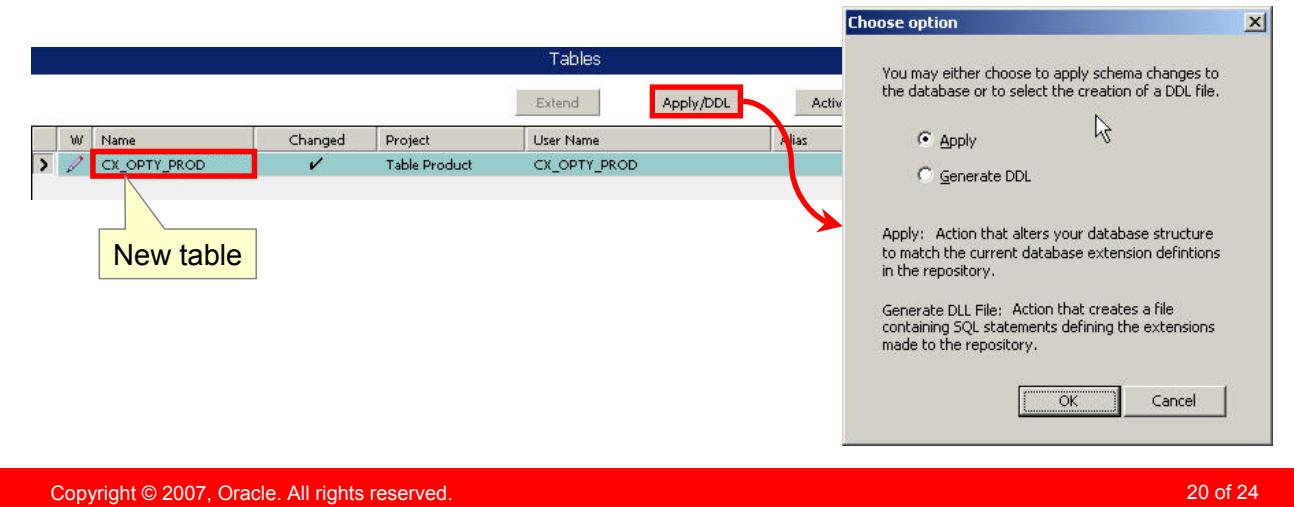
- Table Wizard creates an intersection table with:
 - ▶ Data (Intersection) as its type
 - ▶ Nine system columns
 - ▶ Two foreign key columns as specified
 - ▶ Three indexes:
 - P1 index on ROW_ID
 - U1 index on two foreign key columns, TYPE, and CONFLICT_ID
 - F1 index on foreign key to second parent table

Applying and Propagating Database Changes

- Test changes locally before applying them to the server database
 - ▶ Reduces the likelihood of undesired changes to the server schema
- Best practices for changing the schema:
 1. Apply Changes to the Local Database
 2. Propagate Changes to the Server Database
 3. Propagate Changes to Other Developers

1. Apply Changes to Local Database

- Click Apply/DDL to make the physical database changes
 - ▶ Choice to apply schema changes or generate DDL script
 - ▶ Changes are preserved across Siebel application upgrades
- Compile relevant objects and projects
- Test changes locally before checking in to the server
 - ▶ Query tables/columns using a database SQL utility



Copyright © 2007, Oracle. All rights reserved.

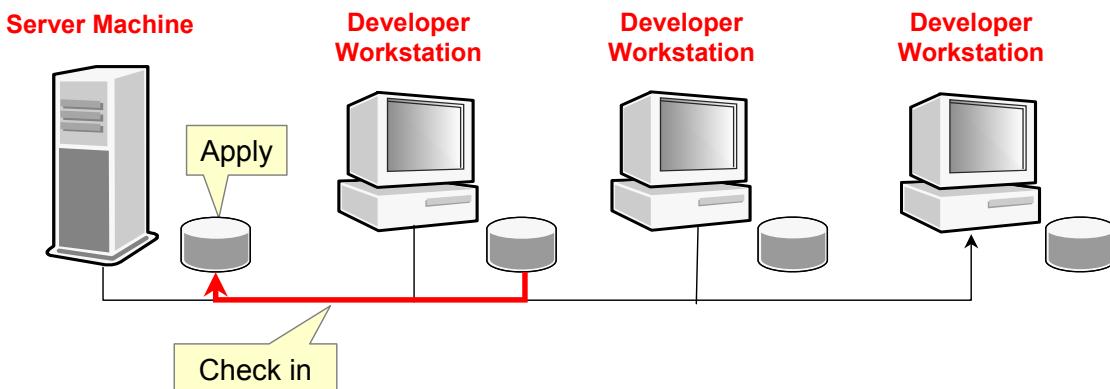
20 of 24

DDL

Stands for Data Definition Language, the subset of SQL statements used to define and manipulate database objects. Typical SQL commands used in DDL are CREATE, DELETE, ALTER, and so on.

2. Propagate Changes to Server Database

- Check project into the server
 - ▶ Copies the table and column object definitions
 - ▶ Does not apply those changes to the server database schema
- Apply changes to the server database
 - ▶ To make changes visible, press the Activate button on the Table object on the server machine
- Compile and test against the server database

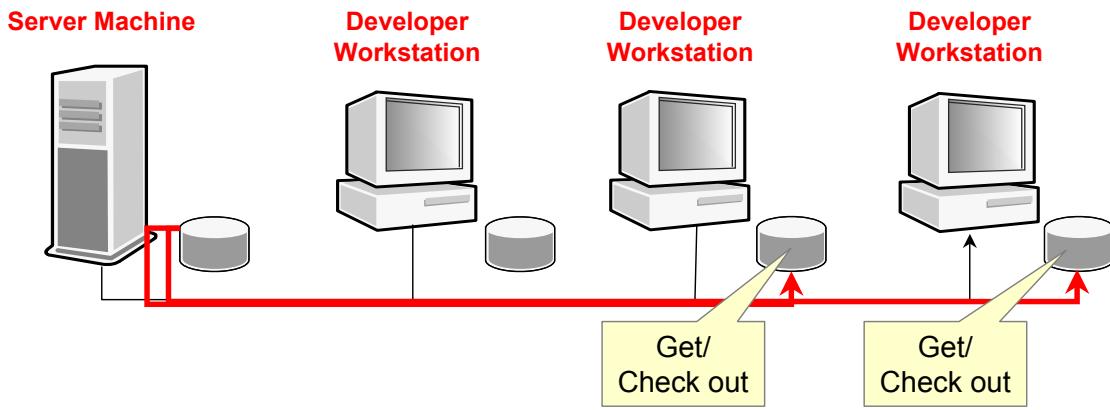


Copyright © 2007, Oracle. All rights reserved.

21 of 24

3. Propagate Changes to Other Developers

- Other developers need to apply changes to their local databases
 - ▶ Have other developers Get or check out the project and apply changes locally
 - ▶ Alternatively re-extract developers and have them get all projects





Module Highlights

- Alternatives to extending Siebel database schema:
 - ▶ Utilize unused columns in an existing table
 - ▶ Use an existing 1:M extension table if appropriate
- Extend the Siebel database in Siebel Tools:
 - ▶ Add an extension column to an existing table
 - ▶ Use the Table wizard to create a new table:
 - Standalone
 - 1:1 extension table
 - 1:M extension table
 - Intersection table
- Best practices to modify Siebel database schema:
 - ▶ Apply changes locally and test
 - ▶ Propagate changes to the server database
 - ▶ Propagate changes to other developers

Copyright © 2007, Oracle. All rights reserved.

23 of 24



Lab

- In the lab you will:
 - ▶ Create a custom extension column on a table
 - ▶ Check in configuration and apply to the server database



Siebel 8.0 Essentials

Module 34: Siebel Business Services

34

Copyright © 2007, Oracle. All rights reserved.

Oracle Internal & Oracle Academy Use Only

34



Module Objectives

- After completing this module you should be able to:
 - ▶ Describe a business service
 - ▶ Describe the structure and role of property sets
 - ▶ Use the business service simulator to test a business service
- Why you need to know:
 - ▶ Business services are an important building block for Siebel workflow



Automating Business Processes

- A business process is a series of activities executed to achieve a specific business objective
 - ▶ Example: the Quote to Cash business process (how an enterprise creates a quote and converts it to an order for submission)
- Automation options within the Siebel application can address such challenges as:
 - ▶ Maintaining and standardizing consistent business processes across all business units
 - ▶ Routing and assigning tasks accurately and efficiently
 - ▶ Responding in a timely, effective manner to customer inquiries and service requests
 - ▶ Assisting users with the implementation of best practices
 - ▶ Offering consistent and personalized service to customers



Siebel Workflow

- Is a set of capabilities to extend the functionality of Siebel applications by automating business processes
- Includes capabilities such as:
 - ▶ Workflow Processes
 - Automates steps in a business process
 - ▶ Workflow Policies
 - Invokes workflow process under specified conditions
 - ▶ Tasks
 - Guides users through a series of views to complete a step in business process
 - ▶ Assignment Manager
 - Automates assignment of data (such as opportunities and service requests) to the desired people
 - ▶ State Model
 - Enforces a limited life cycle for select business entities



Business Service

- Is a unit of functionality that is reusable and globally accessible
 - ▶ Example: The ISS Shipping Cost Service computes shipping charges corresponding to a company's shipping policies
- Enables business logic to be executed repeatedly in multiple different contexts
 - ▶ Business logic is not restricted to a specific object (business component, applet, and so forth)
- Can be invoked in a Siebel workflow process or a Siebel task

W	Name	Changed	Project	Cache	Class
>	ISS Credit Card Transaction Service		ISS Order Management		CSSISSCreditCardTransactionService
	ISS Credit Check Service		ISS Order Management		CSSISSCreditCheckService
	ISS Shipping Calculation Service		ISS Order Management		CSSISSShippingCalculationService
	ISS Shipping Cost Service		ISS Order Management		CSSISSShippingCostService
	ISS Shipping Validation Service		ISS Order Management		CSSISSCheckShippingMethod

Copyright © 2007, Oracle. All rights reserved.

5 of 18

Reference

Integration Platform Technologies: Siebel EAI : Business Services



Prebuilt Business Services

- Siebel repository contains many prebuilt business services to support processing in areas such as
 - ▶ Customer order management
 - ISS Credit Check Service
 - ISS Shipping Cost Service
 - ISS Tax Calculation Service
 - ▶ Enterprise application integration
 - EAI Siebel Adapter
 - EAI HTTP Transport
 - ▶ XML document processing
 - XML Hierarchy Converter
 - XML Converter
 - ▶ Enforcing customer business rules
 - Business Rule Service

ORACLE®

Repository-Stored Business Services

- Some business services are stored in the Siebel repository
 - ▶ Siebel-developed business services
 - Are written in C++
 - Cannot be modified by customers
 - ▶ Custom business services developed by users
 - Are written in Siebel Visual Basic or eScript
 - Are created and modified by customers using Siebel Tools

Business Services					
	Name	Changed	Project	Cache	Class
>	String Consolidation		Siebel Tools	CSSStringConsolidationService	
	String Conversion		Siebel Tools	CSSStringConversionService	
	StringManipulation	✓	Scripting Test	CSSService	

Custom business service



Client-Stored Business Services

- Some business services are stored in the client database
 - ▶ Siebel-developed business services
 - ▶ Custom business services developed by users
- Client-stored business services
 - ▶ Are written in Siebel Visual Basic or eScript
 - ▶ Are created and modified by customers using the Administration - Business Services screen
 - ▶ Are never executed if there is a repository-stored business service of the same name

The screenshot shows the Siebel Administration - Business Service interface. The top navigation bar includes Home, Accounts, Contacts, Opportunities, Quotes, Administration - Business Service, Details, Methods, Scripts, Simulator, and User Properties. The main content area displays a table titled 'Details' with columns for Name, Cache, and Comments. The table lists four entries:

Name	Cache	Comments
Auto Invoice Service		Will automatically create invoices for Time and Expenses projects - PSA demo workflow
CC Prod		Will get the current products the customer owns and look at related products to recom
CXInstanceServiceSuppliment		
Call Center		Will return a score for agent customer satisfaction surveys on SR's

Copyright © 2007, Oracle. All rights reserved.

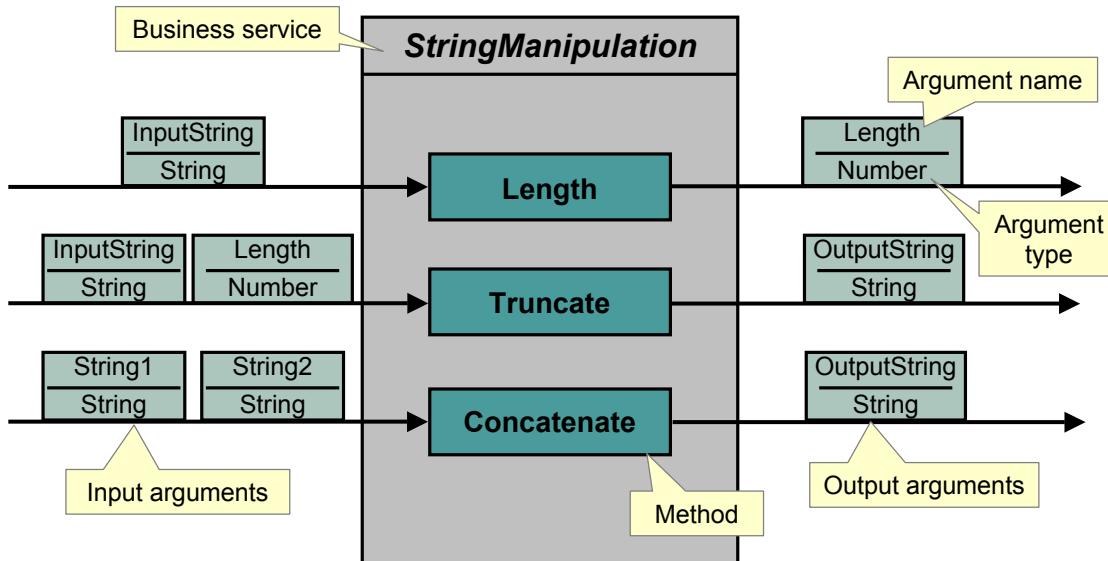
8 of 18

Client Database

The term “client database” refers to tables in the Siebel database that store user data.

Methods

- A business service consists of one or more operations called methods
 - ▶ Each method has a set of input and output arguments, each with a specified type



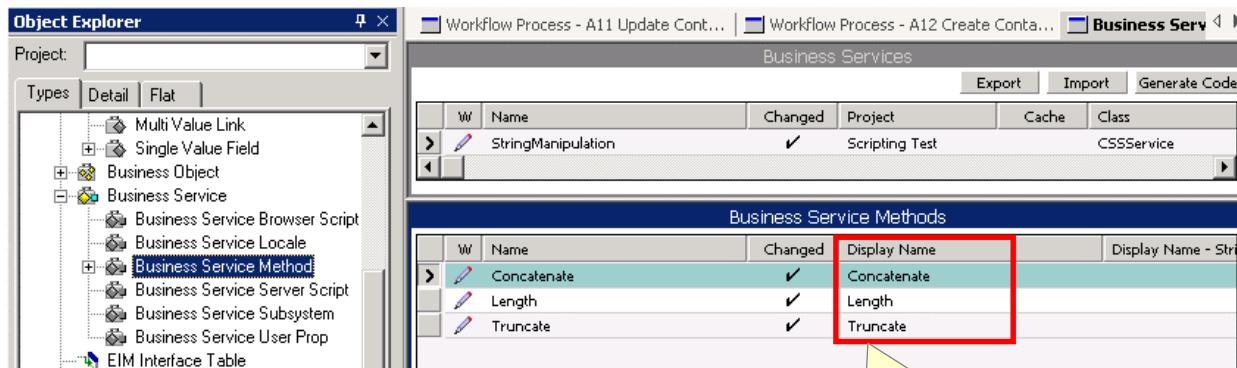
Copyright © 2007, Oracle. All rights reserved.

9 of 18

ORACLE

Identifying Methods for a Business Service

- In Siebel Tools, navigate to Business Service | Business Service Method



Copyright © 2007, Oracle. All rights reserved.

10 of 18

ORACLE

Identifying Arguments and Types for a Method

- In Siebel Tools, navigate to Business Service | Business Service Method | Business Service Method Arg

The screenshot shows two windows side-by-side. The top window is titled 'Business Service Methods' and lists three methods: 'Concatenate', 'Length', and 'Truncate'. The 'Truncate' method is selected and highlighted with a blue background. The bottom window is titled 'Business Service Method Arguments' and shows the arguments for the selected 'Truncate' method. It has columns for Name, Data Type, Type, and Optional. The arguments listed are 'InputString' (String, Input), 'Length' (Number, Input), and 'OutputString' (String, Output). The 'InputString' argument is also highlighted with a blue background.

W	Name	Changed	Display Name	Display Name - String Referer
	Concatenate		Concatenate	
	Length		Length	
>	Truncate		Truncate	
<				

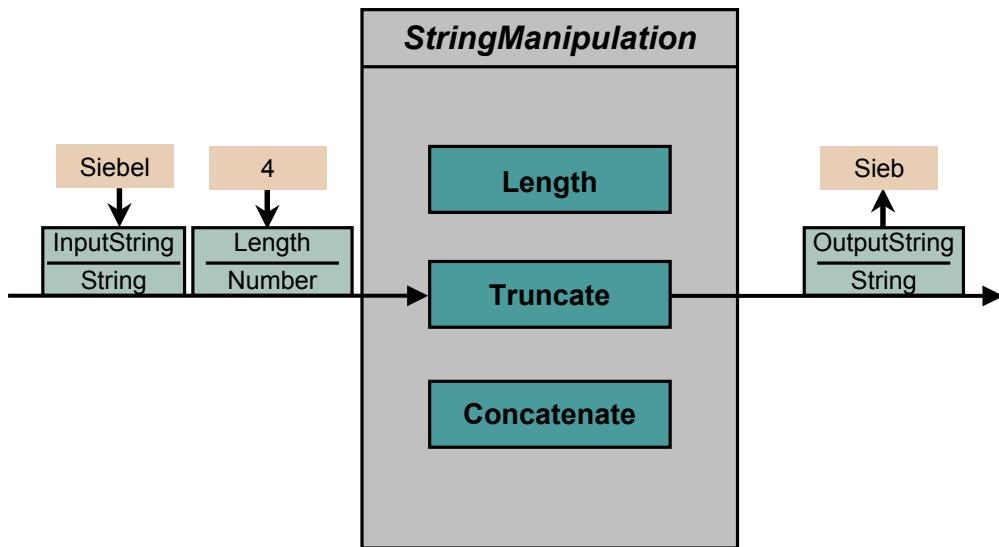
W	Name	Data Type	Type	Optional
>	InputString	String	Input	
	Length	Number	Input	
	OutputString	String	Output	

Copyright © 2007, Oracle. All rights reserved.

11 of 18

Invoking a Method

- Involves:
 - ▶ Assigning values to the input parameters
 - Not all input parameters are required to have values
 - ▶ Retrieving the values assigned to the output parameters

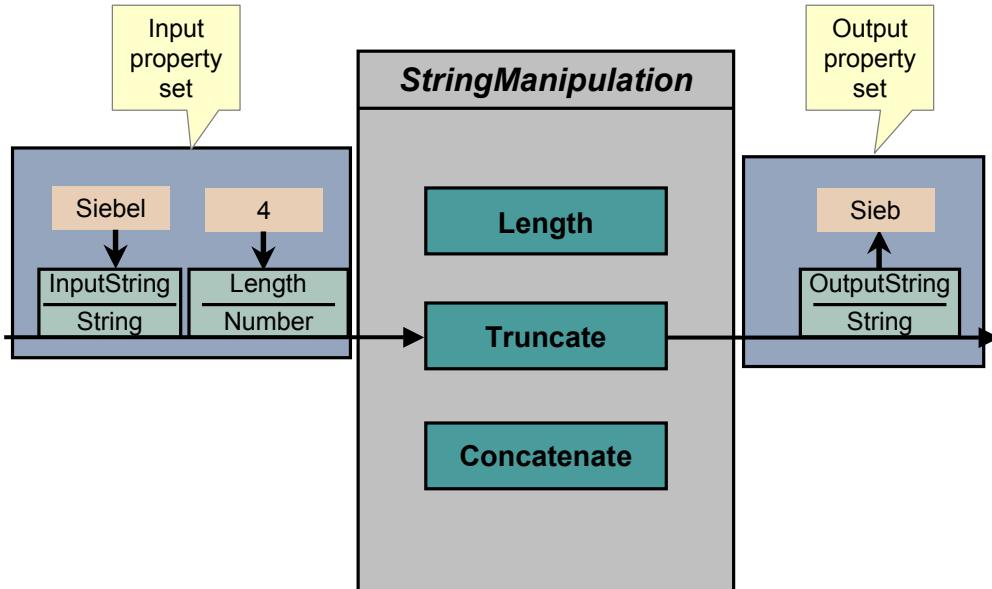


Copyright © 2007, Oracle. All rights reserved.

12 of 18

Property Set

- Is the in-memory data structure used to:
 - ▶ Pass a set of input arguments into a method
 - ▶ Receive a set of output arguments from a method

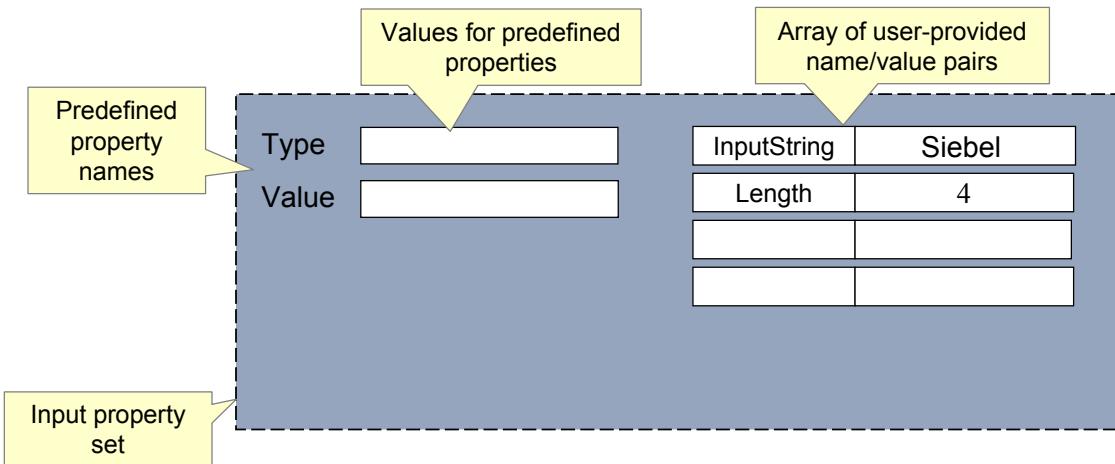


Copyright © 2007, Oracle. All rights reserved.

13 of 18

Property Set Continued

- Represents data using name/value pairs
- Has two predefined properties: Type and Value
- Has an array for storing user-provided name/value pairs
- Is automatically created and populated when invoking most business services from a Siebel workflow or task



Copyright © 2007, Oracle. All rights reserved.

14 of 18



Testing a Business Service

- Use the business service simulator in the Siebel Client
 - ▶ Navigate to Administration - Business Service > Simulator
 - ▶ Select the business service and method
 - ▶ Create the property set name/value pairs
 - Optionally load data from an input file

The screenshot shows the Siebel Business Service Simulator interface. At the top, there's a navigation bar with tabs like Home, Accounts, Contacts, Opportunities, Quotes, Sales Orders, and Administration - Business Service. Below the navigation bar is a toolbar with buttons for Simulator, Menu, New, Delete, Query, Run, Load From File..., Save To File..., and Run on One Input. A status bar at the bottom indicates "1 - 1 of 1".

The main area has two tables. The first table, titled "Input Arguments", has columns for Test Case #, Type, Value, Child Type, Child Value, Property Name, and Property Value. It contains one row with a value of "Length" under "Value" and "4" under "Property Value".

The second table, titled "Property Set Prop...", has columns for Property Name and Value. It contains two rows: "Length" with value "4" and "InputString" with value "Siebel". The "Length" row is highlighted with a red border.

Copyright © 2007, Oracle. All rights reserved.

15 of 18

ORACLE

Testing a Business Service Continued

- Use the business service simulator in the Siebel client
 - ▶ Click Run on One Input
 - ▶ Examine the output property set name/value pairs
 - ▶ Optionally save the output to a file

The screenshot shows the Siebel Business Service Simulator interface. It consists of three main tabs:

- Simulator**: Shows a list of service names, method names, and iterations. A red box highlights the "Run on One Input" button.
- Input Arguments**: Shows a table of input arguments with columns: Test Case #, Type, Value, Child Type, Child Value, Property Name, and Property Value. One row is shown: Length, 4.
- Output Arguments**: Shows a table of output arguments with columns: Test Case, Iteration, Type, Value, Child Type, Child Value, Property Name, and Property Value. One row is shown: 1, OutputString, Sieb.

Copyright © 2007, Oracle. All rights reserved.

16 of 18



Module Highlights

- A business service is a unit of functionality that is reusable and globally accessible
 - ▶ Can be stored in the repository or in user database tables
 - ▶ Consist of one or more methods
 - Each method is specified by a set of input and output arguments
- A property set is an in-memory data structure consisting of name value pairs
- A business service
 - ▶ Is invoked by passing in the input arguments in a property set
 - ▶ Returns the output arguments in a property set
- Use the business service simulator to test a business service

Copyright © 2007, Oracle. All rights reserved.

17 of 18



Lab

- In the lab you will:
 - ▶ Import a custom business service into the repository
 - ▶ Examine the methods and arguments for a business service
 - ▶ Use the business service simulator to test a business service and examine the output property sets