Department of Computer Science and Engineering

Subject Name: **ENTERPRISE SOLUTIONS** Subject Code: **CS T61**

UNIT – V

Siebel Enterprise Applications - Siebel eBusiness Applications – Siebel Tools – Tables and Columns – Business Component – Business Objects – Applets – Joins – Links – Views – Screens – Configuring applications.

2 MARKS

1. What is Siebel?

Siebel is a prominent vendor of interoperable e-business software. They also call themselves Siebel Systems. The company's customer relationship management (CRM), enterprise resource management (ERM), and partner relationship management (PRM) applications are designed to automate those aspects of business and allow an enterprise to perform and coordinate associated tasks over the Internet and through other channels, such as retail or call center networks.

2. Name some customers of Siebel?

Siebel's customers include Chase Manhattan Bank, Deutsche Telecom, IBM, Lucent, Yahoo, and Microsoft.

3. What is Siebel Enterprise Application Integration?

*Siebel Enterprise Application Integration (*Siebel EAI*)* is the set of products on the Siebel Business Platform that includes tools, technologies, and prebuilt functional integrations that facilitate application integration.

4. What are the Features of Siebel EAI?

Some of the features provided by Siebel EAI include:

- Provides components to integrate Siebel Business Applications with external applications and technologies within your company
- Works with third-party solutions, such as solutions from IBM, TIBCO, WebMethods, and so forth
- Provides bidirectional, real time and batch solutions to integrate Siebel applications with other applications
- Provides a set of interfaces that interact with each other and with other components within a Siebel application.

5. What is Siebel e-Business applications?

Siebel e-Business applications synchronize the data through one central repository regardless of the contact points (e.g., web, e-mail, call center, field visits and resellers). By having everyone in the organization working with the same repository (data are stored only once), a complete 360-degree view of the customer can be provided to every department or function on an as-needed basis with consistent and up-to-date information available throughout the organization.

6. What is CRM?

Customer Relationship Management (CRM) processes is an approach to managing a company's interaction with current and future customers. It often involves using technology to organize, automate, and synchronize sales, marketing, customer service, and technical support.

7. Write the three levels of CRM?

- Operational CRM (communication strategy level)
- Analytical CRM (marketing strategy level)
- Collaborative CRM (business strategy level)

8. What is Siebel Tool?

Siebel Tools is an integrated environment for configuring all aspects of a Siebel application so a single configuration can be:

- Deployed across HTML and wireless clients
- Used to support multiple Siebel applications and languages
- Easily maintained
- Automatically upgraded to future Siebel product releases

9. Whether Siebel Tools is a programming environment?

Siebel Tools is not a programming environment

10.What are the two windows of Navigation in Siebel Tools ?

- Object Explorer window
- Object List Editor window

11. Define Tables?

A table object definition is the direct representation of a physical database table in a DBMS. Siebel applications provide a set of standard tables that are included in standard Siebel applications. These tables have predefined names and structures.

12. Define Base Table?

The term base table is used in two different contexts:

- The base table for an extension table is the table it extends. This is specified in the Base Table property of the extension table’s object definition.
- The base table for a business component is the table that provides most of its essential fields. This is specified in the Table property of the Business Component object definition.

13. Write the Properties of the Table Object Type?

- Name
- Type
- Base Table
- User Name
- Alias

14. Define Data tables?

Data tables comprise most of the tables in Siebel applications. They serve as base tables for business components, and their columns provide the data for fields. Data tables can be public or private.

15. Define an Extension table?

An extension table provides additional columns to a data table that cannot be directly added to the original table because the underlying DBMS may support only a limited number of columns, or will not allow adding a column to a table once it is populated with data. An extension table allows you to provide additional columns for use as fields in a business component without violating DBMS or Siebel application restrictions.

16. Define an Interaction Table?

An intersection table implements a many-to-many relationship between two business components.

17. What is Column object?

A Column object definition is the direct representation of a database column in a DBMS. The name, data type, length, primary key and foreign key status, alias, and other properties of the database column are recorded as properties in the corresponding column object definition.

18.What are the types of Column Objects ?

- Name
- Default
- Physical Type (Physical Type Name)

19.What are the types of Columns?

- Data Columns
- Extension Columns
- System Columns

20. Define Data Columns?

Data columns comprise most of the columns in Siebel applications. They are sometimes referred to as base columns. Data columns provide the data for fields, or serve as foreign keys that point to rows in other tables. The developer cannot modify the properties of data columns, unlike extension columns. Data columns can be public or private.

21. Define Extension Columns?

An extension column is a column that is not used by standard Siebel applications. There are three kinds of extension columns:

- Standard extension columns.
- Custom extension columns in an extension table
- Custom extension columns in a base table

22. Define System Column?

System columns have a value of System in their Type property. System columns appear in all tables in Siebel applications, although the same set of system columns does not appear in every table. You can use the data in system columns for various purposes; for example, the ROW_ID column in tables is used in the construction of joins.

23. What is meant by Business Component?

A business component is a logical entity that associates columns from one or more tables into a single structure.

24. Define Business Object?

A business object implements a business model (as represented in a logical database diagram), tying together a set of interrelated business components using links. The links provide the one-to-many relationships that govern how the business components interrelate in the context of this business object.

25. Define Applets?

An Applet is a data entry form, composed of controls, that occupies some portion of the Siebel application user interface. An applet can be configured to allow data entry, provide a table of data rows, or display business graphics, or a navigation tree. It provides viewing, entry, modification, and navigation capabilities for data in one business component.

26. Write the types of Applets?

- Form Applet
- List Applet

27. Define Form Applet?

A form applet displays data in a data entry form. Fields in the business component appear on the form applet as text boxes, check boxes, and other standard controls

28. Define List Applet?

A list applet allows the simultaneous display of data from multiple records. A list applet displays data in a list table format, much like a spreadsheet or word processor table.

29. Define Joins?

A Join object definition creates a relationship between a business component and a table other than its base table. The join allows the business component to use columns from that table. The join uses a foreign key in the business component to obtain rows on a one-to-one basis from the joined table, even though the two do not necessarily have a one-to-one relationship.

30. Define Links?

A link implements a one-to-many (or master-detail) relationship between business components based on their base tables. The Link object type makes master-detail views possible, in which one record of the master business component displays with many detail business component records that correspond to the master.

31. What is Multi value Link?

A multi-value link implements a special use of the Link object type, which is the maintenance by the user of a list of records attached to a control or list column in an applet. The group of attached detail records is called a multi-value group.

32. Define Views?

A view is a collection of applets that appear at the same time on the same screen. A view can be thought of as a single window's worth of data forms (applets). Generally, a Siebel application window displays one view at any one time. The currently active view is changed by selecting a different view from the view tabs.

33. Define List Form View?

A list applet and a form applet display data from the same business component. The list applet appears above the form applet. The form applet presents the same information as the currently selected record in the list applet, with a different arrangement that may include more fields.

34. Define Master-detail view?

In a master-detail view, a form applet and a list applet display data from two business components related by a link. The form applet appears above the list applet. The form applet displays one record from the master business component in the master-detail relationship. The list applet displays all of the records from the detail business component that have as their master record the record currently displayed in the form applet.

35. Define Thread Bar?

The thread bar is a navigational tool for the user. It provides the means to navigate from view to view among the views previously visited in the current screen.

36. What is Drilldown Behavior in a View?

It allows the user to drill down from a cell in a list applet (or using a pop-up menu in either a form or list applet) to a particular view. Drilldown controls or list columns in a list applet in Siebel applications consist of colored, underlined text, much like a hypertext link in a Web browser.

37. Give the types of Drilldown Behavior?

- Static Drilldown Behavior
- Dynamic Drilldown Behavior

38. Define screens?

A screen is a logical collection of views. It is not a visual construct in itself; rather, it associates views so that other visual constructs, such as the Site Map and tab bar, can reflect the list of views contained in the currently active screen.

39. Write any Configuring applications for deployment in High Interactivity mode.

- Browser scripting is fully supported in High Interactivity mode.
- You cannot modify the appearance of the rich text editor.
- You cannot modify the background and text color of list applets.
- You cannot place method-invoking controls.

11 MARKS

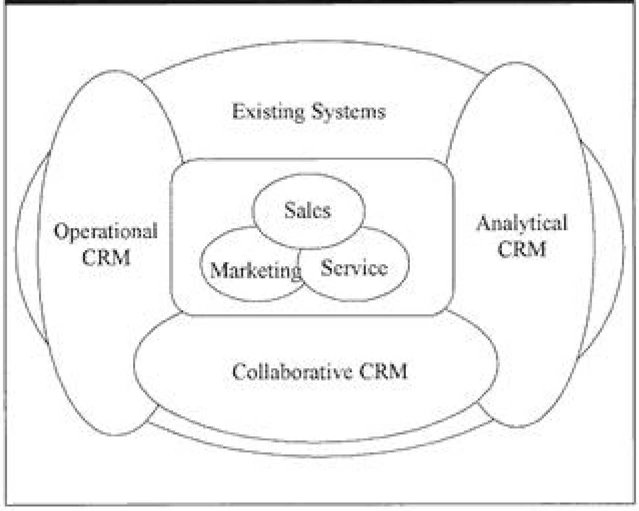
1. Explain CRM Processes in detail.

Companies, both large and small, are looking to gain resources like automating finance, manufacturing, distribution, human resources management and general office operations, many businesses through CRM strategies that enable them to manage all customer contact points, including person-to-person contact, the telephone, e-mail and the Internet.

Different levels of CRM3 (see **figure 1**) include:

- **Operational CRM** (communication strategy level) is the automation of the customer-oriented front-end marketing, sales and service processes and their integration with back-office systems. An example of operational CRM is a customer database maintained purely for direct mailing purposes.
- **Analytical CRM** (marketing strategy level) is the analyses of the data created with operational CRM to better meet the needs and desires of customers. For example, relations between events and buying behaviors are analyzed, typically in a data warehouse (possibly with neural network capabilities).
- **Collaborative CRM** (business strategy level) is the overall communication and coordination model of a customer life cycle between the channels and contact points. In this collaborative CRM approach, business processes and a central customer database are drivers for success.

Figure 1—CRM Overview



Siebel is designed to deal with all the different levels of CRM and supports CRM as a business strategy. It should be taken into account that the way CRM is implemented and effectively utilized will vary between organizations. It is dependent on many variants, including the way of doing business, buying habits of customers (e.g., convenience vs. specialty goods), industry (e.g., telecom with many consumers vs. forestry with a few business customers), company size, existing systems and organization structure.

Organization-specific aspects need to be incorporated in the process designs for marketing (e.g., campaign management, lead generation), sales (e.g., opportunity management, contact and account management and quotation management) and service (e.g., call center processes, service agreement management). These processes have to be related to the relevant business units and Siebel products. As a result of this mapping process, changes or improvements may be identified for the business units as well as gaps between the process design and the Siebel standard solution.

It is important for the IT auditor to be involved in the process design to ensure integrity, security auditability, controllability, availability, efficiency and effectiveness requirements are defined for the core marketing, sales and service processes as well as for the management control and support activities surrounding these processes (e.g., interfaces with the backbone ERP software).

2. Explain Siebel Tool in detail?

Siebel Tools is an integrated environment for configuring all aspects of a Siebel application so a single configuration can be:

- Deployed across HTML and wireless clients
- Used to support multiple Siebel applications and languages

- Easily maintained
- Automatically upgraded to future Siebel product releases

Siebel Tools is not a programming environment; it is a declarative application configuration tool. Standard Siebel applications provide a core set of object definitions that you can use as a basis for your tailored application. Using Siebel Tools and other configuration tools that are part of a Siebel solution, Siebel application developers, system administrators, and database administrators can customize a standard Siebel application without modifying source code or SQL. Some of the configuration tools are accessed through the Siebel applications. Siebel Tools, however, is a separate product with its own user interface.

Navigation in Siebel Tools is done mainly in two windows:

- Object Explorer window
- Object List Editor window

3. Explain in details about Tables.

Tables:

A table object definition is the direct representation of a physical database table in a DBMS. Siebel applications provide a set of standard tables that are included in standard Siebel applications. These tables have predefined names and structures, and typically begin with one of the following prefixes:

S_. Table names starting in S_ are standard tables for supplied with Siebel Sales and Siebel Service. For example, the S_CONTACT table stores contact information, and S_OPTY stores opportunity information. Nearly all standard tables are of this type.

W_. Table names starting in W_ are Data Warehouse tables used in Siebel Analytics to demoralize data used in the S_ tables.

Base Tables

The term base table is used in two different contexts:

■ The base table for an extension table is the table it extends. This is specified in the Base Table property of the extension table's object definition.

■ The base table for a business component is the table that provides most of its essential fields. This is specified in the Table property of the Business Component object definition. The set of fields supplied by the base table is supplemented by non updateable fields that are obtained from joins.

Properties of the Table Object Type:

The following are the key properties in a table object definition:

1. Name: Provides the name of the table in the DBMS.

2. Type: Indicates which of the following styles describes the table.

- **Data (Public).** Public data tables are among the original set of tables implemented in Siebel applications. They hold data that is made available through business components to developers and users. Public data tables can be extended using extension tables and extension columns. These extension are subject to database restrictions. Data tables are discussed in "Data Tables" on page 196.
- **Data (Private).** Private data tables are similar to public data tables, but cannot have extension columns.
- **Data (Intersection).** Identifies an intersection table. An intersection table implements a many-to-many relationship between two data tables. Intersection tables are discussed in "Intersection Tables" on page 204.
- **Extension**. An extension table adds additional columns to a data table that the original data table is unable to hold due to DBMS platform or Siebel application design restrictions. Extension table names have an _X suffix, or _XM or one-to-many, or _T for TAS extension tables. NOTE: Extension tables that have _XM suffix have table type of "Data (Public)." Extension tables are discussed in "Extension Tables" on page 196.
- **Interface.** Interface tables are used by Siebel Enterprise Integration Manager (EIM) to import initial data for populating one or more base tables and subsequently to perform periodic batch updates between Siebel applications and other enterprise applications. Interface table names end in _IF or _XMIF. Interface tables are discussed in "Column Objects" on page 215.
- **Database View**, Dictionary, Journal, Log, Repository, Virtual Table, and Warehouse styles. These are all table types that are reserved for Siebel internal use.
- **Extension (Siebel).** These tables are reserved for Siebel use only. They are usually extensions from S_PARTY. If customers want to extend person- and organization-related tables they need to extend from S_PARTY. For example, S_CONTACT is an extension table of S_PARTY. Because S_CONTACT is of type Extension (Siebel), you cannot use it as a parent table for an extension table. You must use S_PARTY. For a business component based on your new table to show data from S_CONTACT, you must create a Join object that references S_CONTACT and has a Join Specification child object with a Source Field property set to Parent Id and Destination Column property set to ROW_ID. The row ID of an S_CONTACT record will be the same as the row ID of the corresponding S_PARTY record.

3. Base Table. Identifies the base table if the table in the object definition is an extension table. If the table in the object definition is a base table, this property is blank. An extension table always identifies a base table.

4. **User Name**. A longer, descriptive name that aids in identifying the table when used in configuration.

5. **Alias**. A name that can be used as a synonym for the table name to make the name more understandable. For example, an alias such as S_Organization_External could be specified for the S_ORG_EXT table.

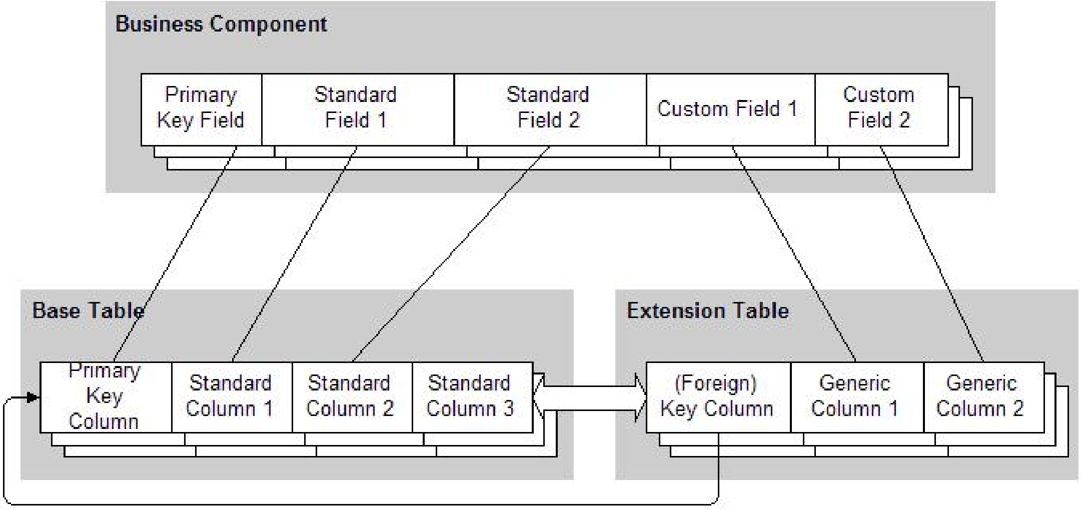
Data Tables :

Data tables comprise most of the tables in Siebel applications. They serve as base tables for business components, and their columns provide the data for fields. Data tables can be public or private.

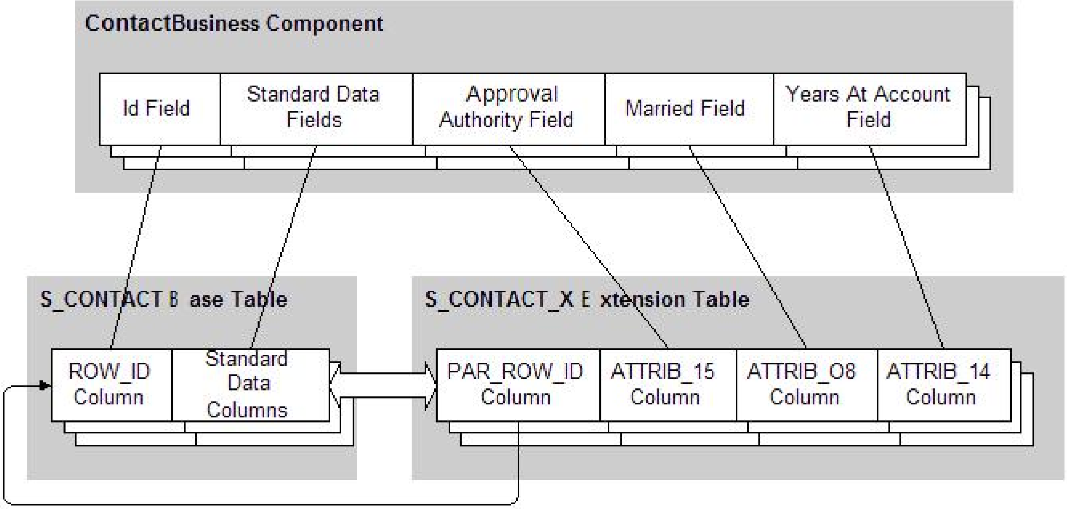
4. Explain in detail about Extension Tables.

An extension table provides additional columns to a data table that cannot be directly added to the original table because the underlying DBMS may support only a limited number of columns, or will not allow adding a column to a table once it is populated with data. An extension table allows you to provide additional columns for use as fields in a business component without violating DBMS or Siebel application restrictions.

An extension table is a logical augmentation of an existing table. Its columns are provided mostly for developers, and are generally not used by standard Siebel applications. An extension table extends a base table in the sense that it effectively adds additional columns. These columns are not physically part of the base table, but are available for use in a business component alongside the base table columns as if they were. When columns in a base table are updated, the time stamps of its extension tables are not updated unless columns in those extension tables are also updated. When records in an extension table are changed, system columns in a parent table are updated. This is done because the associated record in an extension table is considered by the object manager to be logically a part of its parent record. The relationships between a base table, an extension table, and the business component that uses them are given below in the figure.



Business Component, Base Table and Extension Table



Extension Table Example

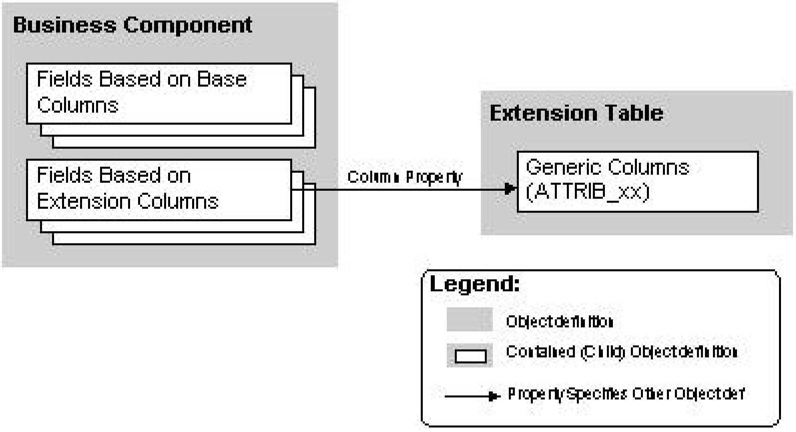
Extension tables can be of the one-to-one or one-to-many style:

Rows in *one-to-one* extension table have a one-to-one relationship with corresponding rows in the base table. A one-to-one extension table extends the base table horizontally

In a ***one-to-many*** extension table, there are multiple extension table rows for each base table row. There are standard one-to-many extension tables for certain of the major business components, including Opportunity, Contact and Account. These are used primarily to create multi-value groups based on user-created business components.

One-to-One Extension Tables

One-to-one extension tables have the _X suffix on their names (with the exception of TAS tables, which have the suffix _T). The details of the object definition relationships (excluding the implied join) are given in the figure.


Business Component
Fields Based on Base Columns
Fields Based on Extension Columns
Column Property
Extension Table
Generic Columns (ATTRIB_xx)
Legend:
Object definition
Contained (Child) Object definition
Property Specifies Other Object def

Extension Table Details (Excluding Implied Join)

The object definitions in the above figure are:

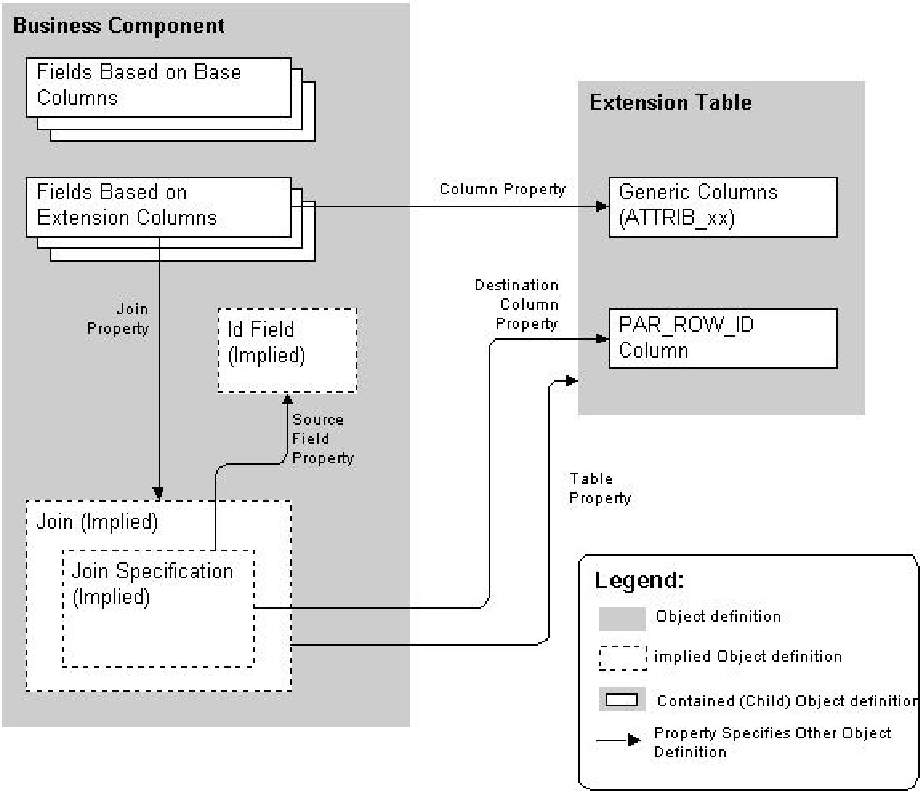
- **Business component.** Business component being extended.
- **Fields based on base columns.** Fields that represent data from columns in the business component's base table. They are unaffected by the extension table.
- **Fields based on extension columns.** Represent data from columns in the extension table.
- **Extension table.** Provides columns that may be used to add developer-defined fields to the business component.

Implied Joins

Underlying the one-to-one extension table's relationships with the base table and business component is a set of hidden relationships called an *implied* or *implicit join*. The implied join makes the extension table rows available on a one-to-one basis to the business component that uses the extension table. Every extension table has an implied join with the business component it extends. This join always has the name of the extension table.

An implied join is different from joins defined as object definitions. Data can be updated through an implied join. Data can be displayed only through other joins. This update capability is important for extension table functionality.

When a field in the business component is based on a column in the extension table, the Column property of the Field object is set to the name of the column, and the Join property is set to the name of the extension table. For example, the Birthday field in the Contact business component has a Column property value of ATTRIB_26 and a Join property value of S_CONTACT_X.


Business Component
Fields Based on Base Columns
Fields Based on Extension Columns
Join Property
Id Field (Implied)
Source Field Property
Join (Implied)
Join Specification (Implied)
Column Property
Destination Column Property
Table Property
Extension Table
Generic Columns (ATTRIB_xx)
PAR_ROW_ID Column
Legend:
Object definition
implied Object definition
Contained (Child) Object definition
Property Specifies Other Object Definition

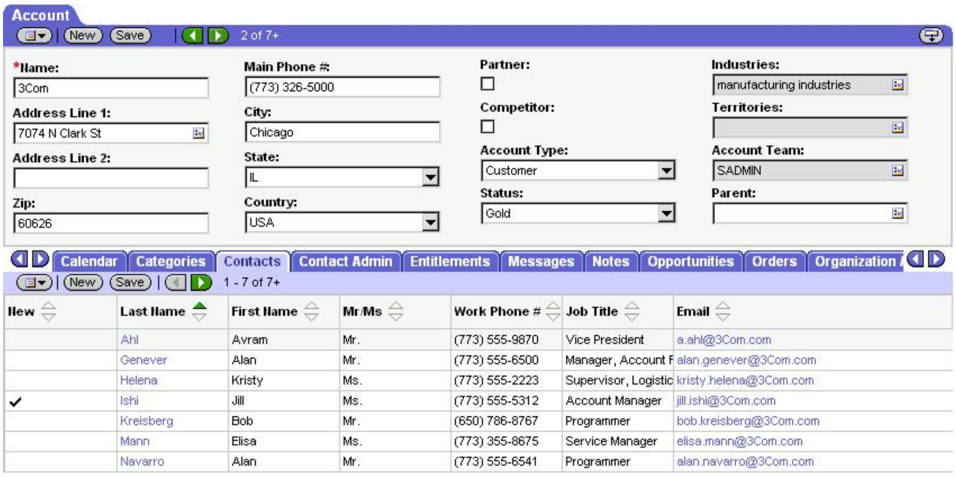
Extension Table Details with Implied Join

5. Explain Intersection Table with neat architecture.

An *intersection table* implements a many-to-many relationship between two business components.

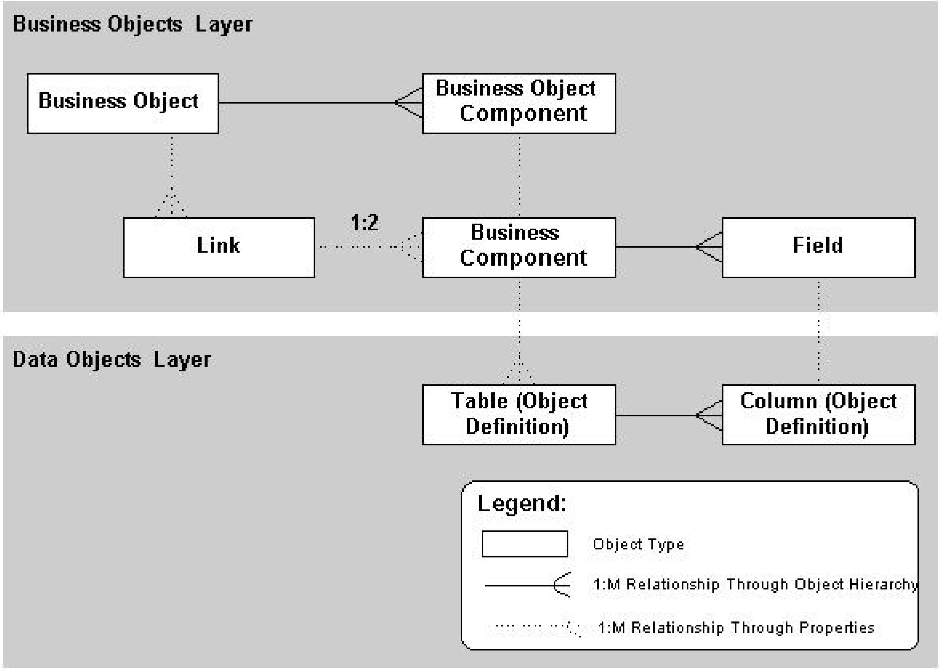
A *many-to-many relationship* is one in which there is a one-to-many relationship from either direction. For example, there is a many-to-many relationship between Opportunities and Contacts. One Opportunity can be associated with many Contact people, and one Contact person can be associated with many Opportunities.

The below figure shows the Account Detail - Contacts View, in which one account is displayed with multiple detail contacts.



*Name: 3Com	Main Phone #: (773) 326-5000	Partner: □	Industries: manufacturing industries
Address Line 1: 7074 N Clark St	City: Chicago	Competitor: □	Territories:
Address Line 2:	State: IL	Account Type: Customer	Account Team: SADMIN
Zip: 60626	Country: USA	Status: Gold	Parent:

New	Last Name	First Name	Mr/Ms	Work Phone #	Job Title	Email
	Ahl	Avram	Mr.	(773) 555-9870	Vice President	a.ahl@3Com.com
	Genever	Alan	Mr.	(773) 555-6500	Manager, Account F	alan.genever@3Com.com
	Helena	Kristy	Ms.	(773) 555-2223	Supervisor, Logistic	kristy.helena@3Com.com
✓	Ishi	Jill	Ms.	(773) 555-5312	Account Manager	jill.ishi@3Com.com
	Kreisberg	Bob	Mr.	(650) 786-8767	Programmer	bob.kreisberg@3Com.com
	Mann	Elisa	Ms.	(773) 355-8675	Service Manager	elisa.mann@3Com.com
	Navarro	Alan	Mr.	(773) 555-6541	Programmer	alan.navarro@3Com.com


Business Objects Layer
Business Object
Business Object Component
Link
1:2
Business Component
Field
Data Objects Layer
Table (Object Definition)
Column (Object Definition)
Legend:
Object Type
1:M Relationship Through Object Hierarchy
1:M Relationship Through Properties

Intersection Table Architecture

The following are descriptions of the object definitions:

Business object. The business object references the link (indirectly through the business object's child business object component) that uses the intersection table. It also contains the two business components included in the link.

Business object components. Business Object Component object definitions are used to include business components in the business object. Business Object Component is a child object type of Business Object. The detail business object component references both the detail business component, by means of the Business Component property, and the link, by means of the Link property. The master business object component only references its corresponding business component.

Link. The link object definition establishes a one-to-many relationship between the two business components in a particular direction. That is, the property settings in the link specify that one business component is the master and the other is the detail in the master-detail relationship.

Master and detail business components. The two business components are specified in the link. They provide data to the user interface object definitions that display the master-detail relationship. The base table of each business component contains the ROW_ID column referenced by the Inter Child Column (detail) and Inter Parent Column (master) properties of the Link object type.

Intersection table. The intersection table holds the associations between rows in the base tables of the master and detail business components. Each row in the intersection table represents one association between the two business components. Two columns in the intersection table serve as foreign keys to the base tables of the two business components.

These columns are identified in the Inter Parent Column and Inter Child Column properties of the link.

Inter Parent column. This column in the intersection table holds the pointer to the associated row in the master business component's base table. It is identified in the Inter Parent Column property of the Link object.

Inter Child column. This column in the intersection table holds the pointer to the associated row in the detail business component's base table. It is identified in the Inter Child Column property of the Link object.

ROW_ID columns. The base table of each business component has a unique identifier column for the rows in that table. This is the ROW_ID column

Column Objects

A Column object definition is the direct representation of a database column in a DBMS. The name, data type, length, primary key and foreign key status, alias, and other properties of the database column are recorded as properties in the corresponding column object definition.

A column has one of several styles based on the value in the Type property. These styles include Data, Extension, IFMGR, System, and others.

Column Object Type

The column object corresponds to one database column in the database table that is represented by the parent table object definition. Each database column in the database table needs to have a corresponding column object definition. The important properties of the Column object type are as follows:

Name. Provides the name of the database column in the database table.

Default. Provides a default value when new rows of this table are added.

Physical Type (Physical Type Name). Identifies the data type of the column in the database. The following data types are supported:

Character. Used for fixed-length text. Also used for Boolean columns, which are character columns with a length of 1.

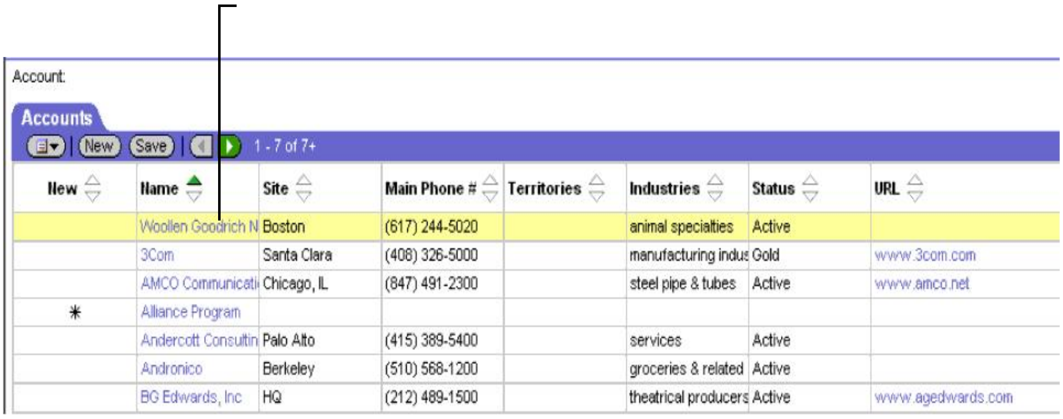
Long. Long text. You can store approximately 16K worth of data in long columns. By default, you cannot have char greater than 1. If you want, you need to set the preference under Options/Database.

Varchar. Variable-length text. Used for memo-type fields and to store row-ID and foreign key values.

Number. Any numeric data. Typical numeric columns in Siebel applications are 22,7 for general-purpose numbers, and 10,0 for integers.

6. Describe Business Components in detail.

A *business component* is a logical entity that associates columns from one or more tables into a single structure. Business components provide a layer of wrapping over tables, so that applets reference business components rather than the underlying tables. This creates convenience (all associated columns are together in one bundle), developer-friendly naming, and the separation of the developer role from the database administrator role. A business component can also have a default sort or search specification, providing records to applets in a predetermined sort order and according to a selection criterion. When instantiated in a Siebel application, a business component is comparable to a recordset. Its definition in Siebel Tools provides the foundation for controlling how data is selected from, inserted, and updated within the tables it references.



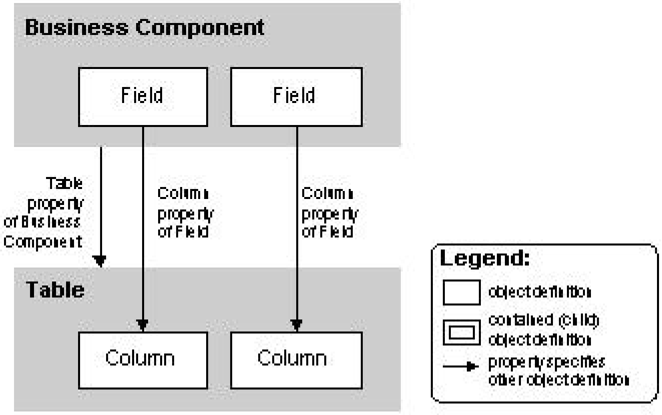
Account:

Accounts

New	Name	Site	Main Phone #	Territories	Industries	Status	URL
	Woollen Goodrich N	Boston	(617) 244-5020		animal specialties	Active	
	3Com	Santa Clara	(408) 326-5000		manufacturing indus	Gold	www.3com.com
	AMCO Communicati	Chicago, IL	(847) 491-2300		steel pipe & tubes	Active	www.amco.net
*	Alliance Program						
	Andercott Consultin	Palo Alto	(415) 389-5400		services	Active	
	Andronico	Berkeley	(510) 568-1200		groceries & related	Active	
	BG Edwards, Inc	HQ	(212) 489-1500		theatrical producers	Active	www.agedwards.com

Business Component Records in a List Applet

A business component contains fields. Each column whose data is included in a business component is represented with a field. The Field object type is a child of the Business Component object type. The Column object type is a child of the Table object type. The set of field child object definitions of a business component maps a corresponding set of columns into the business component.



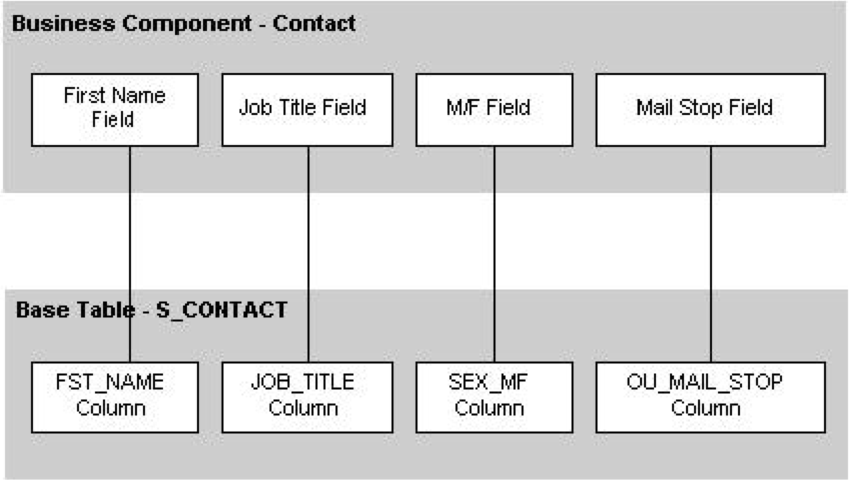
Business Component Field and Column Relationships

In addition to connecting a column to the business component, the Field object type also allows you to specify a meaningful name that refers to that column. Columns in tables are often cryptically named to match the names in the DBMS, whereas fields can have more meaningful and longer names than the columns they represent (75 characters long as opposed to, typically, 30 characters).

Base Tables of Business Components

A *base table* of a business component is assigned to the business component to provide the most important columns for use as fields in the business component. Fields built on the base table can be edited, whereas fields built on joined tables can only be displayed.

The base table is assigned to the business component with the Table property in the Business Component's object definition. The below an example of some fields in the Contact business component that map corresponding columns from the business component's base table, S_CONTACT.



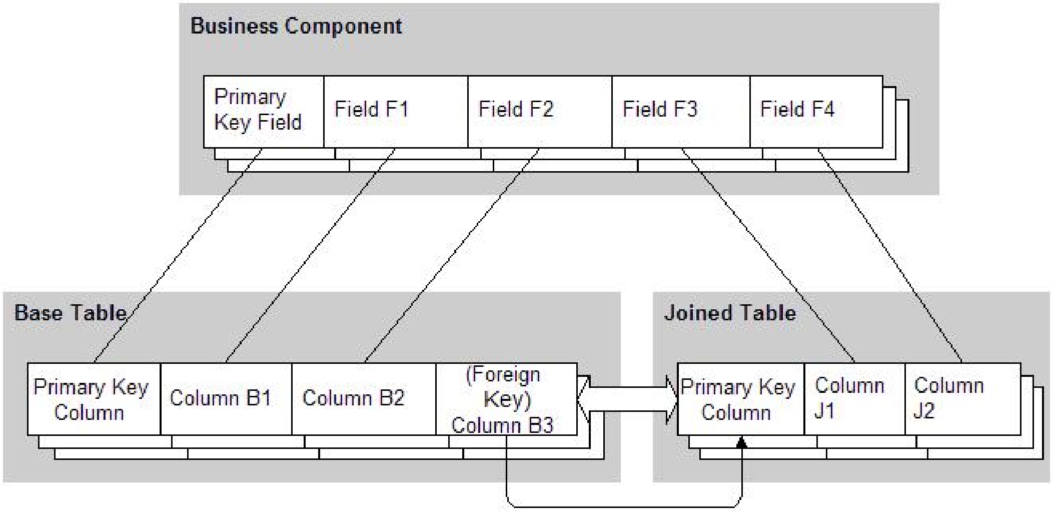
Examples of Fields Representing Columns

Every business component has a base table assigned to it. It is not essential that the business component include all of the columns in the base table, although typically it will include most of them. In particular, system columns in the base table such as ROW_ID, CREATED_BY and LAST_UPD_BY are automatically represented in the business component through implied fields. System columns do not require field object definitions in the business component.

Joined Tables

A *joined table* provides rows on a one-to-one basis to the business component as a result of a foreign key relationship between the joined table and the business component's base table. That is, for every record in the business component (which corresponds to a row in the base table) there can be a corresponding row in the joined table. However, not every record in the base table will have a record in the joined table.

The data obtained by a business component through a join (other than to an extension table) is read-only in that business component.



Fields from the Base Table and a Joined Table

Extension Tables

Extension tables are a special kind of joined table. Like other joined tables, extension tables provide rows on a one-to-one basis in parallel with base table rows. Extension tables are identified by the _X suffix in the table name, such as S_ORG_EXT_X, which extends S_ORG_EXT.

Extension tables are provided specifically to allow columns to be virtually added to a base table rather than physically added. This provides the means to expand base tables without violating DBMS or database design restrictions, and without the need to perform complicated database restructuring operations. Extension table data, unlike the data in other joined tables, can be updated in the business component.

Intersection Business Components

An *intersection business component* is a business component based on an intersection table. It provides the means to display all of the combinations of data in a many-to-many relationship, instead of only one or the other one-to-many relationship of which it is composed.

Intersection tables implement many-to-many relationships. Some (such as S_OPTY_CON and S_ACCNT_POSTN) also provide intersection data through a join to one or the other master business component that uses the intersection table. Intersection data is data that resides in columns other than the two required foreign key columns in the intersection table, and is specific to the intersection of the two master business components.

7. Describe in detail about Virtual Business Components:

Business components based on external data are called *virtual business components*. Virtual business components are used when the business component has to obtain data from a location other than a database table in the Siebel database, but the information has to be presented in the standard Siebel user interface (applets and views). This is typically real-time information from another database, such as from the Report Encyclopedia in Actuate, or from an SAP table, although anything that can supply data in response to a SQL query is a candidate.

Virtual business components allow you to:

- Represent external data (for example, data in an SAP R/3 database) as a virtual business component within a Siebel application—the business component configuration specifies the DLL to use to access the data
- Use business services to transfer data

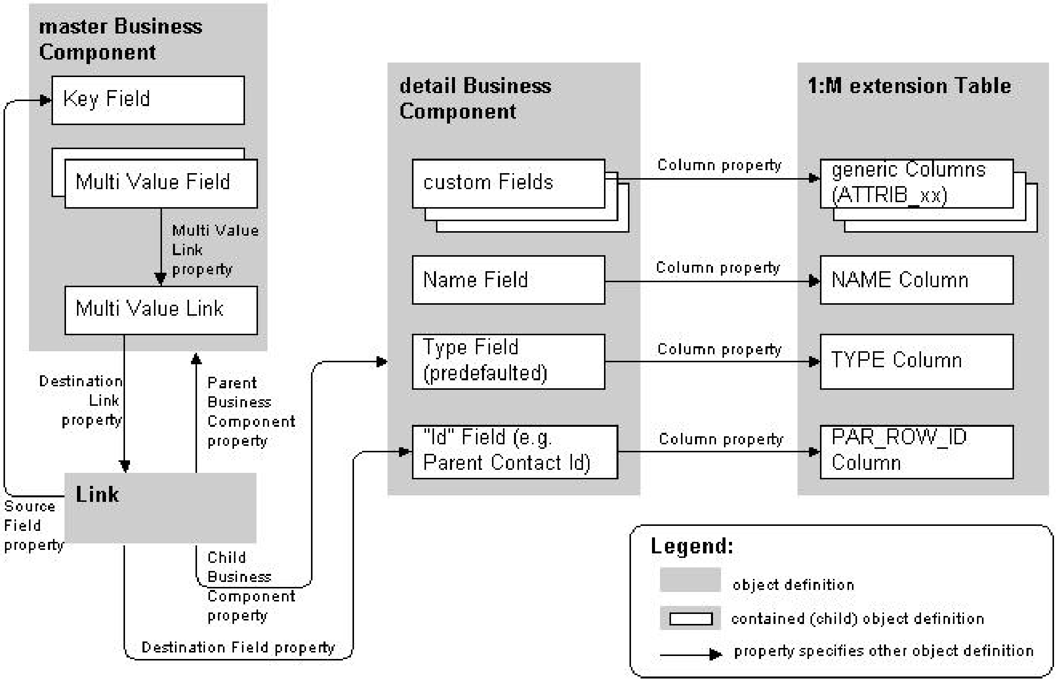
Virtual business components support properties such as:

- Single-value fields
- Field-level validation
- Standard business component event model (for example, PreNewRecord, PreDelete, and so on)
- Insert, delete, query, and update operations

Additional information about virtual business components:

- Applets can be based on virtual business components.
- Virtual business components can be accessed through object interfaces.
- All business component events are available for scripting.
- Virtual business components cannot be docked.
- Virtual business components can be used as stand-alone or children business components in a business object.
- Virtual business components support dynamic applet toggles. For more information about applet toggles.

8. Explain Master-Detail Business Components in detail.


master Business Component
Key Field
Multi Value Field
Multi Value Link property
Multi Value Link
Destination Link property
Parent Business Component property
Link
Source Field property
Child Business Component property
Destination Field property
detail Business Component
custom Fields
Name Field
Type Field (predefaulted)
"Id" Field (e.g. Parent Contact Id)
Column property
Column property
Column property
Column property
1:M extension Table
generic Columns (ATTRIB_xx)
NAME Column
TYPE Column
PAR_ROW_ID Column
Legend:
object definition
contained (child) object definition
property specifies other object definition

One-to-Many Extension Table Details

Master Business Components

The master business components will hold the new multi-value fields. Master business component contain the following important object definitions:

■ **Key field.** This is the key field in the master business component; it is used to reference individual records. Typically, it is named Id. The Source Field property of the Link object definition points to this field. The property value may be blank because, by default, a blank Source Field value refers to the Id field.

Multi-value field. The multi-value field provides access to a corresponding field value in the current record of the detail business component. The control or list column that displays the multi-value field normally will be able to invoke a multi-value group applet for display and maintenance of the detail records.

■ **Multi-value link.** The multi-value link provides access to the set of records in the detail business component. One multi-value link is created for each multi-value group that is created using the one-to-many extension table.

Link

The link object definition creates the one-to-many relationship between the master and detail business components. There are no special link configuration issues related to one-to-many extension tables.

Detail Business Component

The detail business component represents the one-to-many extension table used by the multi-value link and multi-value group applet. Multiple custom business components can be created using the same one-to-many extension table. Each custom business component presents a different type of data for use in a different multi-value group.

The detail business component contains custom fields that represent generic (ATTRIBxx) extension columns, and hold whatever data is required for the application. For example, an Area Of Expertise business component might have a Subject Area field, a Years Of Experience field, and a Licensed field. Each field is a mapping of a different generic extension column.

The following three fields are part of the User Key (U1), which uniquely identifies a row for EIM:

Name field. The name field represents the NAME column from the extension table. It provides the means for the user to enter an identifying value in each record. For example, in a Hobbies business component, the name field might be called Hobby. The user would enter the name of a hobby into each record in this field.

Type field. The type field is usually named Type, and represents the TYPE column. It contains the same value for all records in one multi-value group, and distinguishes the records of that multi-value group from others. It should be set in the Predefault property to some identifying word or phrase, such as HOBBY, EXPERTISE or PRIOR JOB, and should not be exposed in the user interface.

Parent ID field. The parent ID field represents the PAR_ROW_ID column. Generally it is named Parent Contact Id, Parent Account Id, or something similar. It identifies the row ID of the base table row corresponding to the parent record in the master business component The parent ID field is specified in the Destination Field property of the Link object.

The detail business component contains one important property for use with a one- to-many extension table:

- **Search Specification.** The Search Specification property should be set to restrict the records retrieved to only those with a specific value in the Type field. This is the same value that is specified in the Pre Default Value property for that field.

9. Explain Business Objects in Detail.

Business Objects A business object represents a major functional area of the enterprise—every major entity has a business object. Examples of business objects are Opportunity, Account, and Contact. A business object is a collection of related business components; for example (as shown in Figure 6), the Opportunity business object consists of Opportunities plus related Contacts, Activities, Products, and Issues. Each business object has one business component that serves as the master or driving business component. This master business component provides focus for the business object, and they both have the same name (the name is Opportunity).

Business Object Types:

Business Component. A business component is a logical entity that associates columns from one or more tables into a single structure. Business components provide a layer of wrapping over tables, causing applets to reference business components rather than the underlying tables. This design creates convenience (all associated columns together in one bundle), developer-friendly naming, and the isolation of the developer role from the database administrator role. Multiple users can instantiate copies of the same business component. Data changes made by any one user are reflected in all instances of the business component.

Field. A field object definition associates a column to a business component. This is how columns from tables are assigned to a business component and provided with meaningful names that the customer developer can easily change. Alternately, a field's values can be calculated from the values in other fields in the business component. Fields supply data to controls and list columns in the Web interface.

Business Object. A business object implements a business model (logical database diagram), tying together a set of interrelated business components using links. The links provide the one-to-many relationships that govern how the business components interrelate in the context of this business object.

Business Object Component. A business object component object definition is used to include a business component and, generally, a link in the business object. The link specifies how the business component is related to another business component in the context of the same business object. **Link.** A link implements a one-to-many relationship between business components. The Link object type makes master-detail views possible. A masterdetail view displays one record of the master business component with many detail business component records corresponding to the master. A pair of links also may be used to implement a many-to-many relationship.

Multi-Value Link. A multi-value link is used in the implementation of a multi-value group. A multi-value group is a user-maintainable list of detail records associated with a master record. The user invokes the list of detail records from the master record when it is displayed in a list or form applet. For example, in an applet displaying the Account business component, the user can click the Select button to the right of the Address text box to see a pop-up window displaying multiple Address records associated with the currently displayed account.

Join. A join object definition creates a relationship between a business component and a table that is not the business component's base table. The join allows the business component to build fields using columns from the non-base (joined) table. The join uses a foreign key in the business component to obtain rows on a one-to-one basis from the joined table, even though the two do not necessarily have a one-to-one relationship.

Join Specification. A join specification is a child object type of join that provides details about how the join is implemented within the business component.

User Property. A user property is a temporary storage field in a business component that is not tied to the database. When a user sets a value in a business component user property, it is not visible to other users of the same business component. User properties are not saved across sessions.

Business Service. A business service is a reusable module containing a set of methods. It provides the ability to call its C++ or script methods from customer-defined scripts and object interface logic, through the invoke-method mechanism.

10. Explain Applet and types of applet?

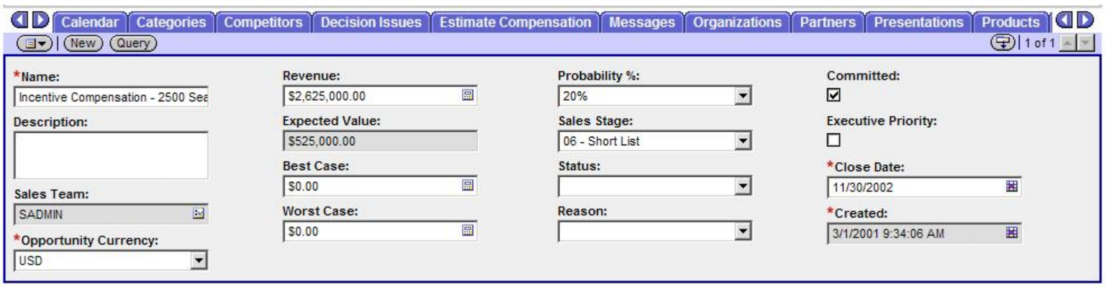
An *applet* is a data entry form, composed of controls, that occupies some portion of the Siebe application user interface. An applet can be configured to allow data entry, provide a table of data rows, or display business graphics, or a navigation tree. It provides viewing, entry modification, and navigation capabilities for data in one business component.

An applet is always associated with a business component. Although the same business component can be associated with multiple applets, an applet is associated with only one business component.

Applets are associated with one or more Siebel Web templates. Web templates are files that contain HTML and proprietary Siebel tags that define the layout and format of the applet in the user interface.

Types of Applets:

Form applet. A *form applet* displays data in a data entry form. Fields in the business component appear on the form applet as text boxes, check boxes, and other standard controls. A form applet appears in below figure.



*Name: Incentive Compensation - 2500 Sea

Description:

Sales Team: SADMIN

*Opportunity Currency: USD

Revenue: $2,625,000.00

Expected Value: $525,000.00

Best Case: $0.00

Worst Case: $0.00

Probability %: 20%

Sales Stage: 06 - Short List

Status:

Reason:

Committed: ☑

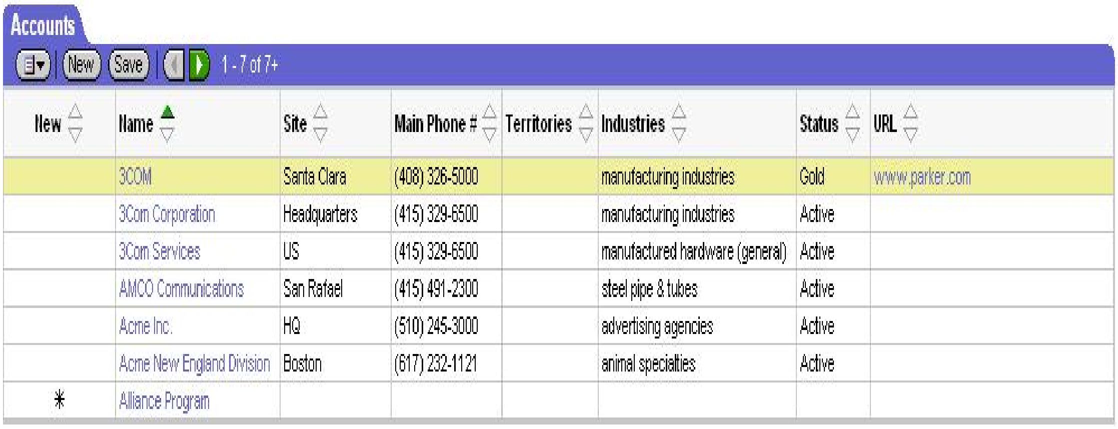
Executive Priority: □

*Close Date: 11/30/2002

*Created: 3/1/2001 9:34:06 AM

Form Applet

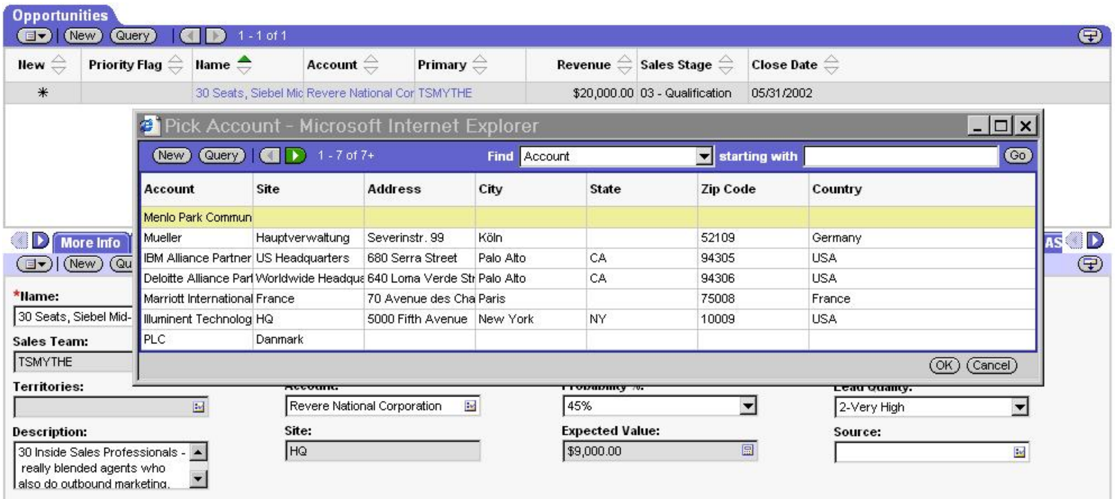
List applet. A *list applet* allows the simultaneous display of data from multiple records. A list applet displays data in a list table format, much like a spreadsheet or word processor table. Rows in the list applet correspond to records in the business component; list columns in the list applet correspond to fields in the business component. In addition to textual data, lists also support images in JPEG and GIF formats and edit controls such as check boxes, drop-down lists, MVGs, and text fields. A list applet appears in below figure.



New	Name	Site	Main Phone #	Territories	Industries	Status	URL
	3COM	Santa Clara	(408) 326-5000		manufacturing industries	Gold	www.parker.com
	3Com Corporation	Headquarters	(415) 329-6500		manufacturing industries	Active	
	3Com Services	US	(415) 329-6500		manufactured hardware (general)	Active	
	AMCO Communications	San Rafael	(415) 491-2300		steel pipe & tubes	Active	
	Acme Inc.	HQ	(510) 245-3000		advertising agencies	Active	
	Acme New England Division	Boston	(617) 232-1121		animal specialties	Active	
*	Alliance Program						

List Applet

Pick applet. A *pick applet* is a dialog box window that appears when a selection is to be made in a control or list column that has the check mark icon to its right. The pick applet provides a list or table of selection values, from which the user selects a value or record. A pick applet displays data that has a M:1 relationship to the data in the parent applet. A pick applet appears in below figure.



New	Priority Flag	Name	Account	Primary	Revenue	Sales Stage	Close Date
*		30 Seats, Siebel Mic	Revere National Cor	TSMYTHE	$20,000.00	03 - Qualification	05/31/2002

Account	Site	Address	City	State	Zip Code	Country
Menlo Park Commun						
Mueller	Hauptverwaltung	Severinstr. 99	Köln		52109	Germany
IBM Alliance Partner	US Headquarters	680 Serra Street	Palo Alto	CA	94305	USA
Deloitte Alliance Part	Worldwide Headqua	640 Loma Verde Str	Palo Alto	CA	94306	USA
Marriott International	France	70 Avenue des Cha	Paris		75008	France
Illuminent Technolog	HQ	5000 Fifth Avenue	New York	NY	10009	USA
PLC	Danmark					

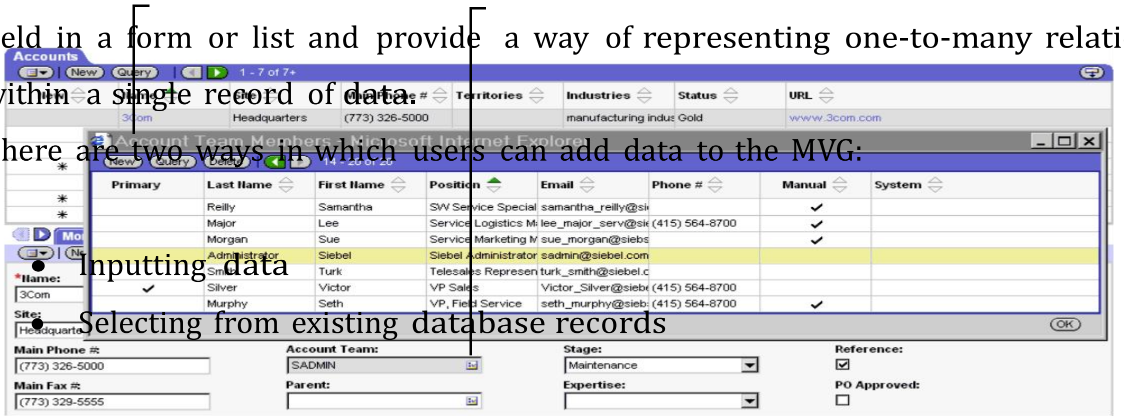
*Name: 30 Seats, Siebel Mid-
Sales Team: TSMYTHE
Territories:
Description: 30 Inside Sales Professionals - really blended agents who also do outbound marketing.
Account: Revere National Corporation
Site: HQ
Probability %: 45%
Expected Value: $9,000.00
Lead Quality: 2-Very High
Source:

Pick Applet

Multi-value group applet. A *multi-value group applet* is used for entry, maintenance, and viewing of a list of detail records associated with one or more fields in the currently displayed master record. MVGs allow the user to associate multiple records to a single field in a form or list and provide a way of representing one-to-many relationships within a single record of data.

There are two ways in which users can add data to the MVG:

- Inputting data
- Selecting from existing database records



Accounts

1 - 7 of 7+

Name	Site	Main Phone #	Territories	Industries	Status	URL
3Com	Headquarters	(773) 326-5000		manufacturing indus	Gold	www.3com.com

Account Team Members - Microsoft Internet Explorer

14 - 20 of 20

Primary	Last Name	First Name	Position	Email	Phone #	Manual	System
	Reilly	Samantha	SW Service Special	samantha_reilly@si		✓	
	Major	Lee	Service Logistics M	lee_major_serv@si	(415) 564-8700	✓	
	Morgan	Sue	Service Marketing M	sue_morgan@siebs		✓	
	Administrator	Siebel	Siebel Administrator	sadmin@siebel.com			
	Smith	Turk	Telesales Represen	turk_smith@siebel.c			
✓	Silver	Victor	VP Sales	Victor_Silver@siebe	(415) 564-8700		
	Murphy	Seth	VP, Field Service	seth_murphy@sieb	(415) 564-8700	✓	

*Name: 3Com			
Site: Headquarters			
Main Phone #: (773) 326-5000	Account Team: SADMIN	Stage: Maintenance	Reference: ☑
Main Fax #: (773) 329-5555	Parent:	Expertise:	PO Approved: □

Multi-Value Group Applet

Chart applet. A *chart applet* graphically displays data from a business component in a bar chart, line graph, pie chart, scatter diagram or other format. It summarizes and illustrates data relationships. Charts are usually accessed through a tab in the third-level navigation level and contain a number of sub- category views (multiple charts of data). These are displayed in an overview of miniature chart graphics (.gif images) with title text. Both the mini-graphics and the title text for the chart are hyperlinked to the detailed version of the chart. A chart applet appears below.

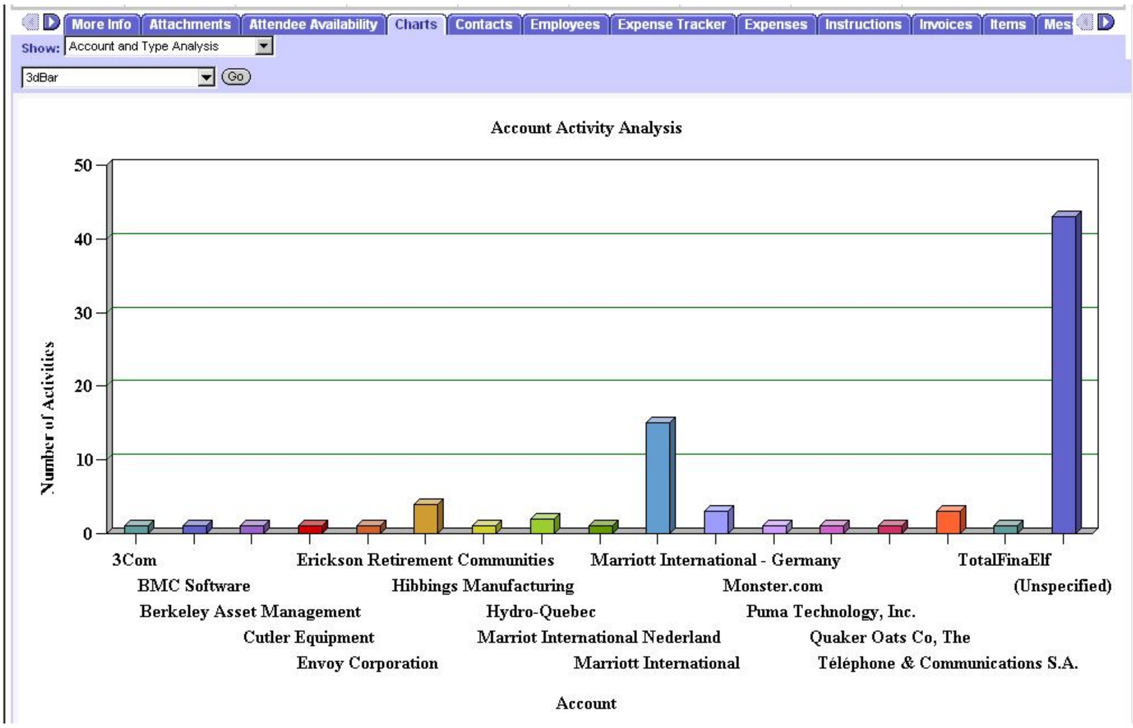
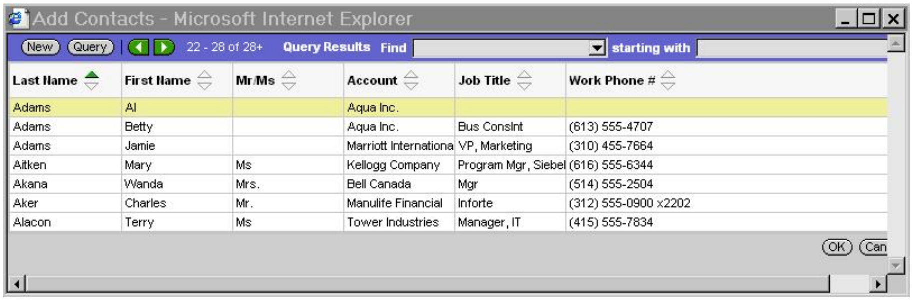


Chart Applet

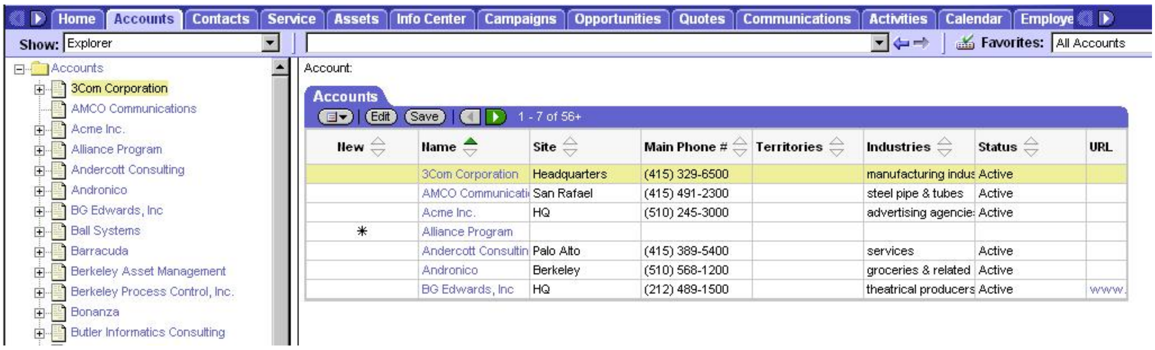
Association applet. An *association applet* provides the user with the ability to associate records of two business components that have a many-to-many relationship. It is invoked from the check mark icon in a multi-value group applet. An association applet appear below.


Add Contacts - Microsoft Internet Explorer

Last Name	First Name	Mr/Ms	Account	Job Title	Work Phone #
Adams	Al		Aqua Inc.		
Adams	Betty		Aqua Inc.	Bus Conslnt	(613) 555-4707
Adams	Jamie		Marriott Internationa	VP, Marketing	(310) 455-7664
Aitken	Mary	Ms	Kellogg Company	Program Mgr, Siebel	(616) 555-6344
Akana	Wanda	Mrs.	Bell Canada	Mgr	(514) 555-2504
Aker	Charles	Mr.	Manulife Financial	Inforte	(312) 555-0900 x2202
Alacon	Terry	Ms	Tower Industries	Manager, IT	(415) 555-7834

Association Applet

Explorer or Tree applet. A *tree applet* is used to create an explorer view that allows the user to navigate hierarchically through a structured list of object instances. A tree applet appears below.



Accounts
- 3Com Corporation
- AMCO Communications
- Acme Inc.
- Alliance Program
- Andercott Consulting
- Andronico
- BG Edwards, Inc
- Ball Systems
- Barracuda
- Berkeley Asset Management
- Berkeley Process Control, Inc.
- Bonanza
- Butler Informatics Consulting

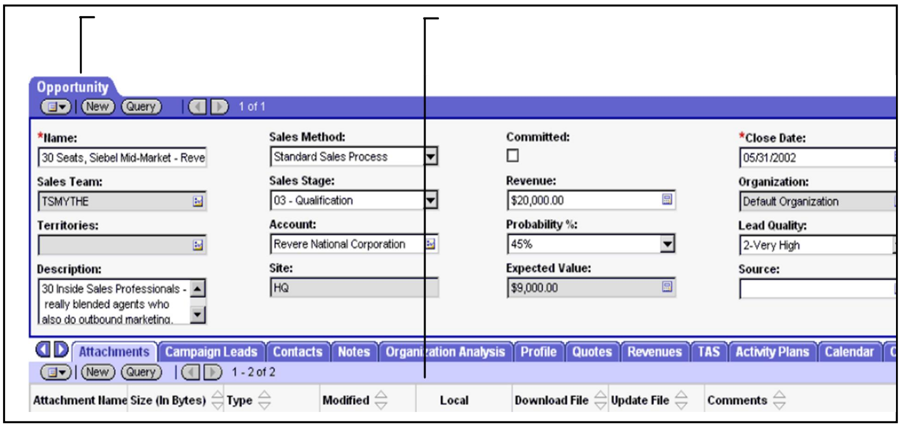
Account:

Accounts

New	Name	Site	Main Phone #	Territories	Industries	Status	URL
	3Com Corporation	Headquarters	(415) 329-6500		manufacturing indus	Active	
	AMCO Communicati	San Rafael	(415) 491-2300		steel pipe & tubes	Active	
	Acme Inc.	HQ	(510) 245-3000		advertising agencie:	Active	
*	Alliance Program						
	Andercott Consultin	Palo Alto	(415) 389-5400		services	Active	
	Andronico	Berkeley	(510) 568-1200		groceries & related	Active	
	BG Edwards, Inc	HQ	(212) 489-1500		theatrical producers	Active	www.

Explorer or Tree Applet

File attachment applet. *File attachment applets* provide access to external documents, such as spreadsheets, word processing documents, and presentations, that have been imported in compressed format into records in a Siebel application. A file attachment applet appears below.



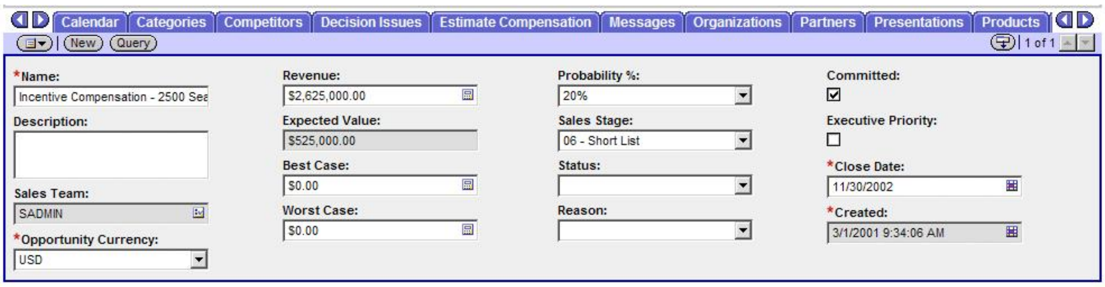
*Name: 30 Seats, Siebel Mid-Market - Reve
Sales Team: TSMYTHE
Territories:
Description: 30 Inside Sales Professionals - really blended agents who also do outbound marketing.
Sales Method: Standard Sales Process
Sales Stage: 03 - Qualification
Account: Revere National Corporation
Site: HQ
Committed: □
Revenue: $20,000.00
Probability %: 45%
Expected Value: $9,000.00
*Close Date: 05/31/2002
Organization: Default Organization
Lead Quality: 2-Very High
Source:

Attachment Name	Size (In Bytes)	Type	Modified	Local	Download File	Update File	Comments

File Attachment Applet

Form Applets

A *form applet* presents business component information in a data entry form layout. An example of a form applet in Siebel Call Center appears in below figure.

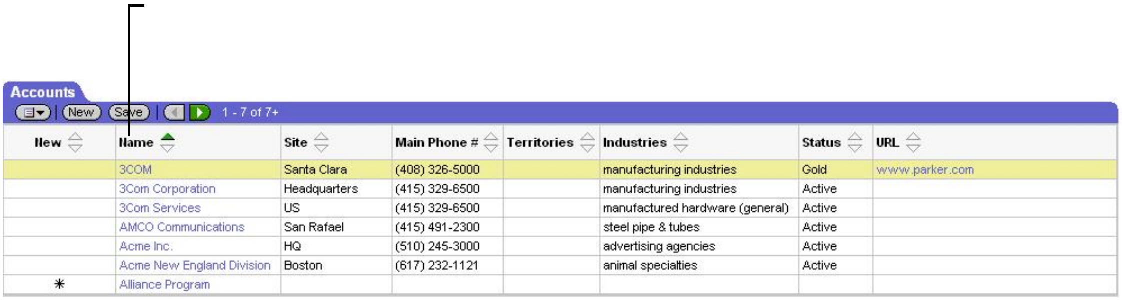


*Name: Incentive Compensation - 2500 Sea
Description:
Sales Team: SADMIN
*Opportunity Currency: USD
Revenue: $2,625,000.00
Expected Value: $525,000.00
Best Case: $0.00
Worst Case: $0.00
Probability %: 20%
Sales Stage: 06 - Short List
Status:
Reason:
Committed: ☑
Executive Priority: □
*Close Date: 11/30/2002
*Created: 3/1/2001 9:34:06 AM

Form Applet in Siebel Call Center

List Applets

A *list applet* allows simultaneous display of data from multiple records and presents business component information in a list table format with multicolumn layout with each record of data represented in a row. In addition to textual data, lists also support images in JPEG and GIF formats and edits control such as check boxes, drop-down lists, noneditable MVGs, and text fields.

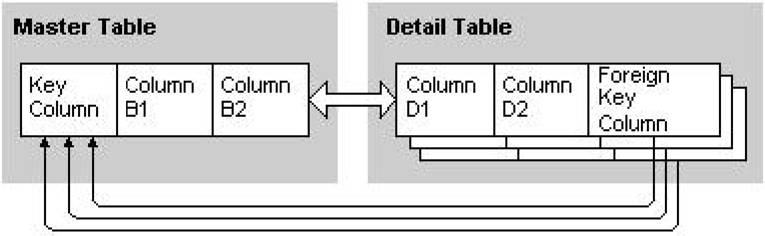


New	Name	Site	Main Phone #	Territories	Industries	Status	URL
	3COM	Santa Clara	(408) 326-5000		manufacturing industries	Gold	www.parker.com
	3Com Corporation	Headquarters	(415) 329-6500		manufacturing industries	Active	
	3Com Services	US	(415) 329-6500		manufactured hardware (general)	Active	
	AMCO Communications	San Rafael	(415) 491-2300		steel pipe & tubes	Active	
	Acme Inc.	HQ	(510) 245-3000		advertising agencies	Active	
	Acme New England Division	Boston	(617) 232-1121		animal specialties	Active	
✳	Alliance Program						

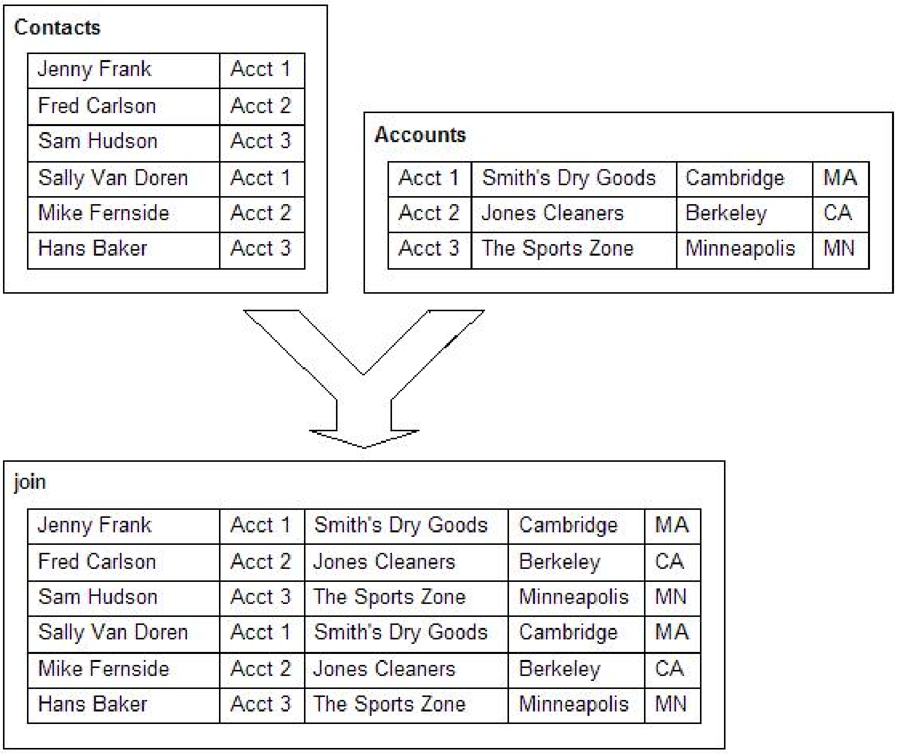
List Applet in Siebel Sales

11. Explain in detail about Joins?

A Join object definition creates a relationship between a business component and a table other than its base table. The join allows the business component to use columns from that table. The join uses a foreign key in the business component to obtain rows on a one-to-one basis from the joined table, even though the two do not necessarily have a one-to-one relationship.



Master-Detail Relationship in a Join



Contacts

Jenny Frank	Acct 1
Fred Carlson	Acct 2
Sam Hudson	Acct 3
Sally Van Doren	Acct 1
Mike Fernside	Acct 2
Hans Baker	Acct 3

Accounts

Acct 1	Smith's Dry Goods	Cambridge	MA
Acct 2	Jones Cleaners	Berkeley	CA
Acct 3	The Sports Zone	Minneapolis	MN

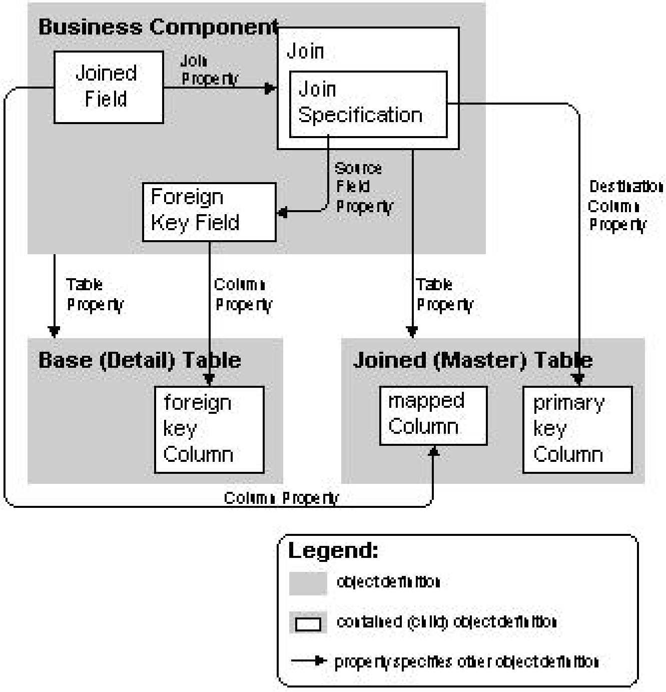
join

Jenny Frank	Acct 1	Smith's Dry Goods	Cambridge	MA
Fred Carlson	Acct 2	Jones Cleaners	Berkeley	CA
Sam Hudson	Acct 3	The Sports Zone	Minneapolis	MN
Sally Van Doren	Acct 1	Smith's Dry Goods	Cambridge	MA
Mike Fernside	Acct 2	Jones Cleaners	Berkeley	CA
Hans Baker	Acct 3	The Sports Zone	Minneapolis	MN

Set of Rows Resulting from a Join

A join is always one-to-one and it is always between a business component and a table. Once a join is created, you can create additional fields in the business component based on columns in the joined table. In the diagram, the account name, city and state are fields that can be added to the Contact business component because of this join.

How a Join is constructed:



Join Relationships

The roles of the object definitions in the diagram are summarized as follows:

Business Component object type. The business component is the parent object definition of the join. Because of the join, fields in the business component (called joined fields) can represent columns from the joined table.

Joined field. A joined field in the business component represents a column from a table other than the business component's base table. Therefore, a joined field must obtain its values through a join. A joined field has the name of the join in its Join property. Together the Join property and Column property identify the column and how to access it. When creating a joined field in a business component, you can change the Type property from the default DTYPE_TEXT to a more appropriate type. For example, if you are joining a table column that contains phone numbers, you can change the Type field to DTYPE_PHONE.

Join object type. Join is a child object type of the Business Component object type. The Join object definition uniquely identifies a join relationship for the parent business component and provides the name of the destination (joined) table. The join object definition identifies the joined table in the Table property. The name of the base table is already known to the business component. Typically, a join object definition is given the same name as the joined table.

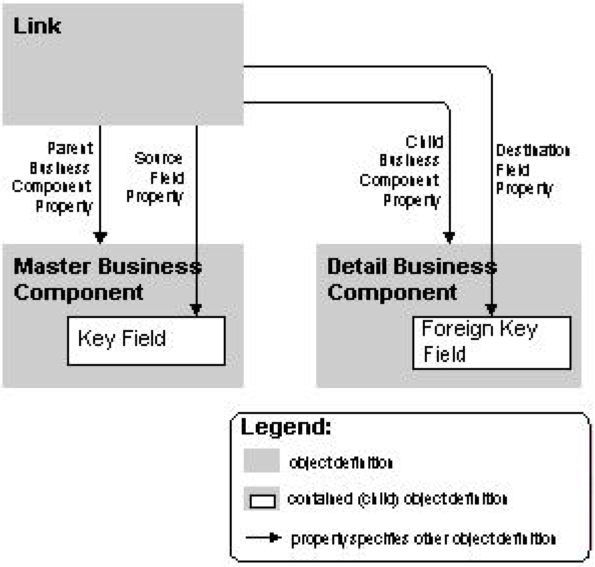
Join Specification object type. The join specification object definition is a child of the join object definition. It identifies the foreign key field in the business component and the primary key column in the joined table.

12. Describe Links in detail.

A link implements a one-to-many (or master-detail) relationship between business components based on their base tables. The Link object type makes master-detail views possible, in which one record of the master business component displays with many detail business component records that correspond to the master.

Link in a Master-Detail View

How a Link is constructed:


Link
Parent Business Component Property
Source Field Property
Child Business Component Property
Destination Field Property
Master Business Component
Key Field
Detail Business Component
Foreign Key Field
Legend:
object definition
contained (child) object definition
property specifies other object definition

Link. The Link object definition specifies a master-detail relationship between two business components. It identifies the master and detail business components, the key field in the master business component, and the foreign key field in the detail business component.

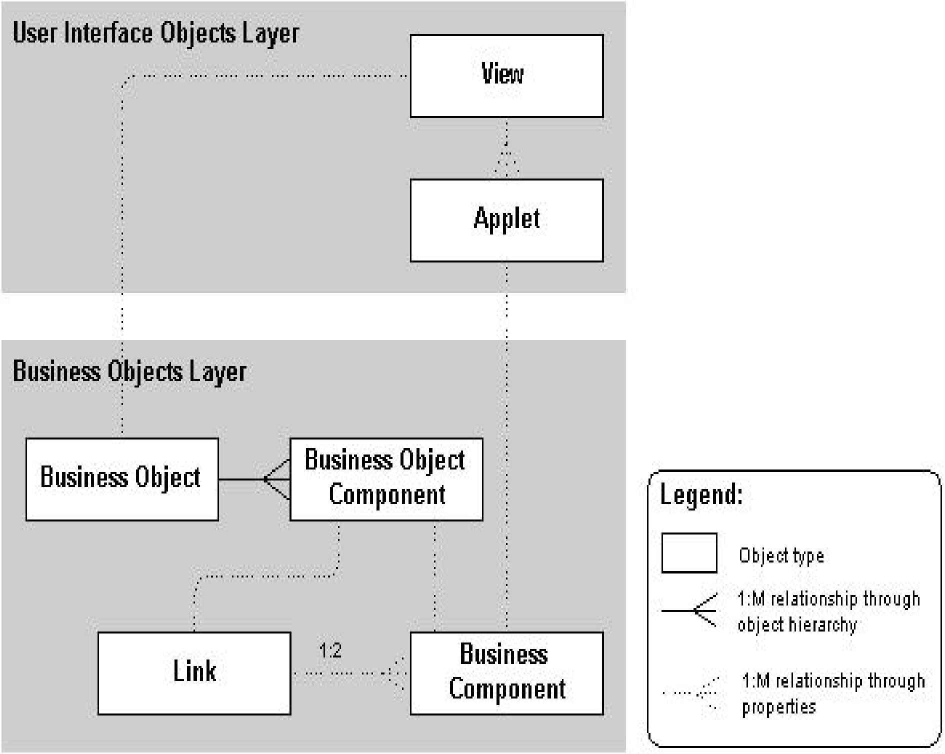
Master business component. The master business component is the "one" in the one-to-many relationship. The name of this object definition is specified in the Parent Business Component property in the Link object definition.

Detail business component. The detail business component is the "many" in the one-to-many relationship. The name of this object definition is specified in the Child Business Component property in the Link object definition.

Source (primary key) field. The source field, also known as the primary key field, is a field in the master business component that uniquely identifies records in the business component. It represents the ROW_ID column from the business component's base table. The name of this field is specified in the Source Field property in the Link object definition. Source field typically, but not necessarily, represents the row id column from the business component's base table.

Destination (foreign key) field. The destination field, also known as the foreign key field, is a field in the detail business component that points back to the master record in the business component. Account Id and Opportunity Id are typical foreign key fields.

Link Architecture


User Interface Objects Layer
View
Applet
Business Objects Layer
Business Object
Business Object Component
Link
1:2
Business Component
Legend:
Object type
1:M relationship through object hierarchy
1:M relationship through properties

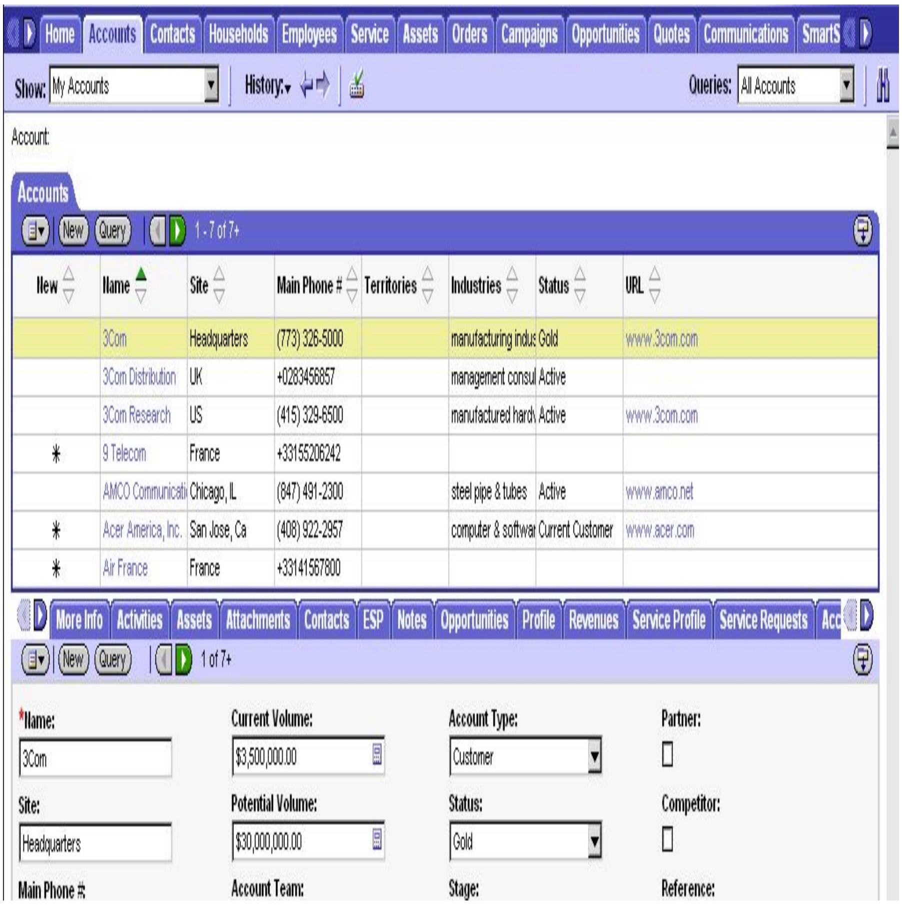
In a master-detail view, a Link object definition is incorporated into a business object (by means of a Business Object Component object definition) to establish the master-detail relationship. This relationship applies to any use of the two business components together within the context of the business object. Each view specifies the business object it uses in its Business Object property. This forces the view to operate as a master-detail view, as specified in the link, without any additional configuration of the view.

A *multi-value link* implements a special use of the Link object type, which is the maintenance by the user of a list of records attached to a control or list column in an applet The group of attached detail records is called a *multi-value group*.

13. Describe Views in detail.

Views are typically of the following styles:

List-form view. In a list-form view, a list applet and a form applet display data from the same business component. The list applet appears above the form applet. The form applet presents the same information as the currently selected record in the list applet, with a different arrangement that may include more fields.



Show: My Accounts

Queries: All Accounts

Account:

1 - 7 of 7+

New	Name	Site	Main Phone #	Territories	Industries	Status	URL
	3Com	Headquarters	(773) 326-5000		manufacturing indus	Gold	www.3com.com
	3Com Distribution	UK	+0283456857		management consul	Active	
	3Com Research	US	(415) 329-6500		manufactured hardv	Active	www.3com.com
*	9 Telecom	France	+33155206242				
	AMCO Communicati	Chicago, IL	(847) 491-2300		steel pipe & tubes	Active	www.amco.net
*	Acer America, Inc.	San Jose, Ca	(408) 922-2957		computer & softwar	Current Customer	www.acer.com
*	Air France	France	+33141567800				

1 of 7+

*Name: 3Com

Current Volume: $3,500,000.00

Account Type: Customer

Partner: □

Site: Headquarters

Potential Volume: $30,000,000.00

Status: Gold

Competitor: □

Main Phone #:

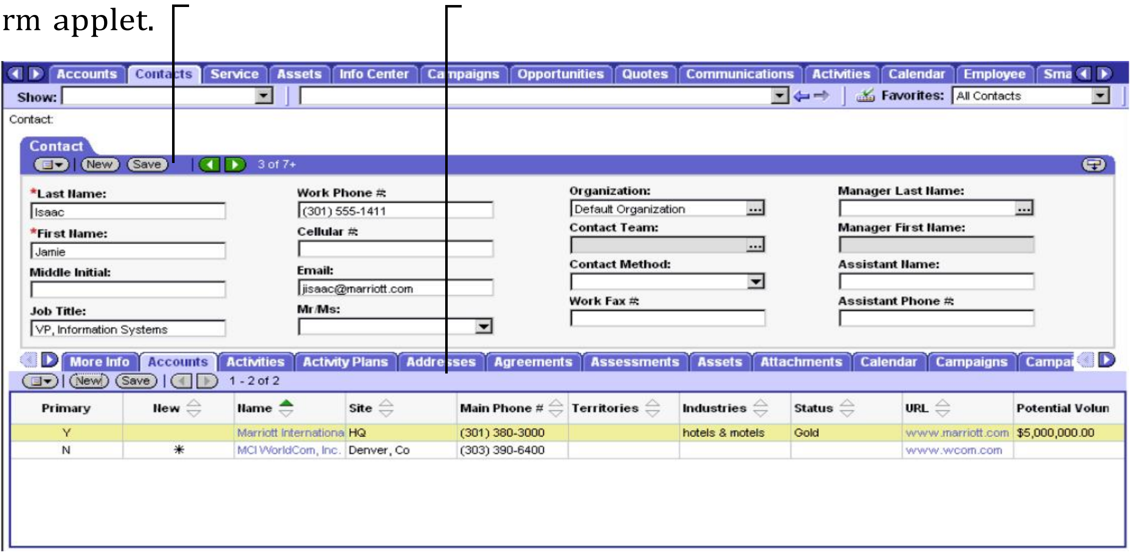
Account Team:

Stage:

Reference:

Master-detail view:

In a *master-detail view*, typically a form applet and a list applet display data from two business components related by a link. The form applet appears above the list applet. The form applet displays one record from the master business component in the master-detail relationship. The list applet displays all of the records from the detail business component that have as their master record the record currently displayed in the form applet.



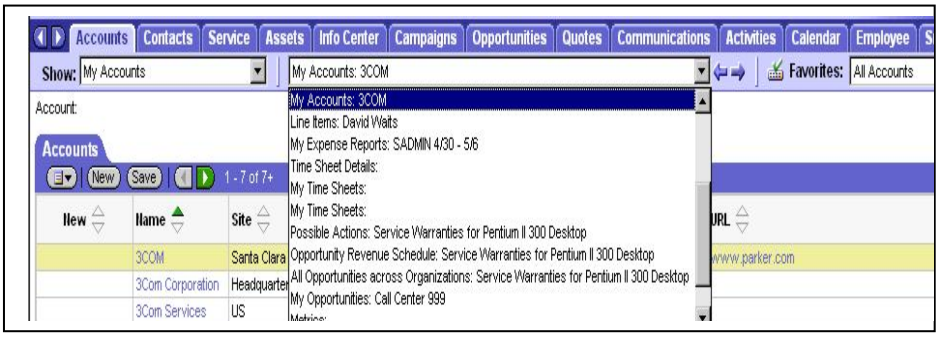
*Last Name: Isaac	Work Phone #: (301) 555-1411	Organization: Default Organization	Manager Last Name:
*First Name: Jamie	Cellular #:	Contact Team:	Manager First Name:
Middle Initial:	Email: jisaac@marriott.com	Contact Method:	Assistant Name:
Job Title: VP, Information Systems	Mr/Ms:	Work Fax #:	Assistant Phone #:

Primary	New	Name	Site	Main Phone #	Territories	Industries	Status	URL	Potential Volun
Y		Marriott Internationa	HQ	(301) 380-3000		hotels & motels	Gold	www.marriott.com	$5,000,000.00
N	*	MCI WorldCom, Inc.	Denver, Co	(303) 390-6400				www.wcom.com	

Master-Detail View in a Siebel Application

Thread bar

The *thread bar* is a navigational tool for the user. It provides the means to navigate from view to view among the views previously visited in the current screen.



New	Name	Site	URL
	3COM	Santa Clara	www.parker.com
	3Com Corporation	Headquarter	
	3Com Services	US	

Thread Bar in a Siebel Application

Drilldown Behavior in a View

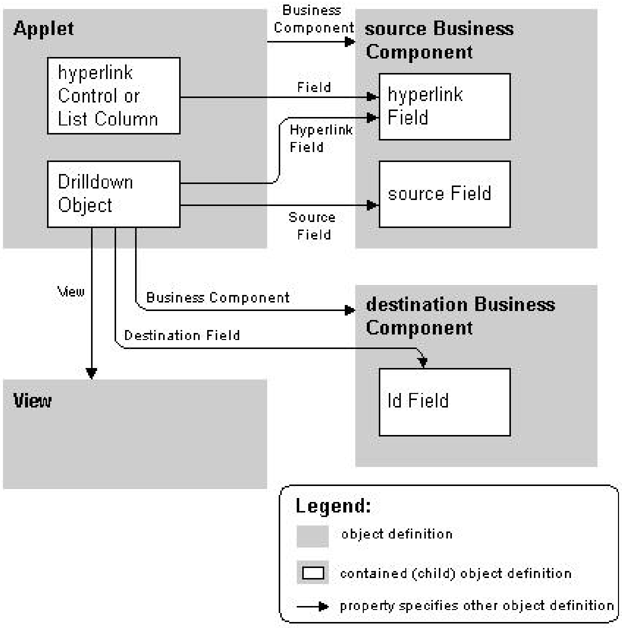
The *Drilldown Object* object type is a child of Applet, used primarily in list applets. It allows the user to drill down from a cell in a list applet (or using a pop-up menu in either a form or list applet) to a particular view. Drilldown controls or list columns in a list applet in Siebel applications consist of colored, underlined text, much like a hypertext link in a Web browser.

Drilldown List Columns in a List-Form View

Types of Drilldown Behavior:

1. Static Drilldown Behavior:

In the example, underlined account name appears in the list column labeled Name. If the user clicks the account in the Name list column, a master-detail view appears, with the selected account in a form applet above an applet displaying the corresponding list of contacts.

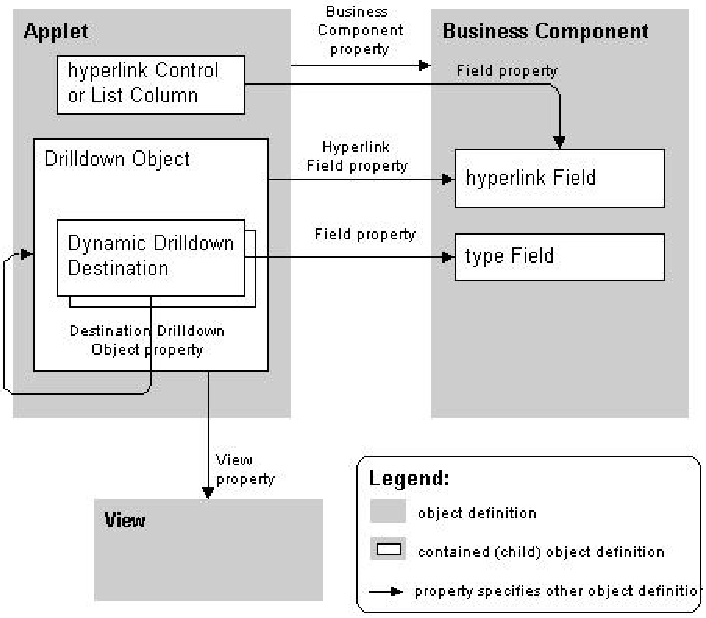


Static Drilldown Configuration

Dynamic drilldown enables hyperlink navigation to multiple views from the same hyperlink field, depending on the value of a field in the applet’s current record.

This is useful in the situation where special processing is desired for various types of contacts, opportunities, accounts, and so on. The business component may have a field that indicates a classification, such as the Lead Quality for an opportunity or the primary Industry for an account. The drilldown behavior can be to check this field in the current record, and navigate to different views for different values found there.

Dynamic drilldown behavior for a hyperlink field (and the corresponding list column or control) is configured with one or more Dynamic Drilldown Destination child object definitions of the Drilldown Object.

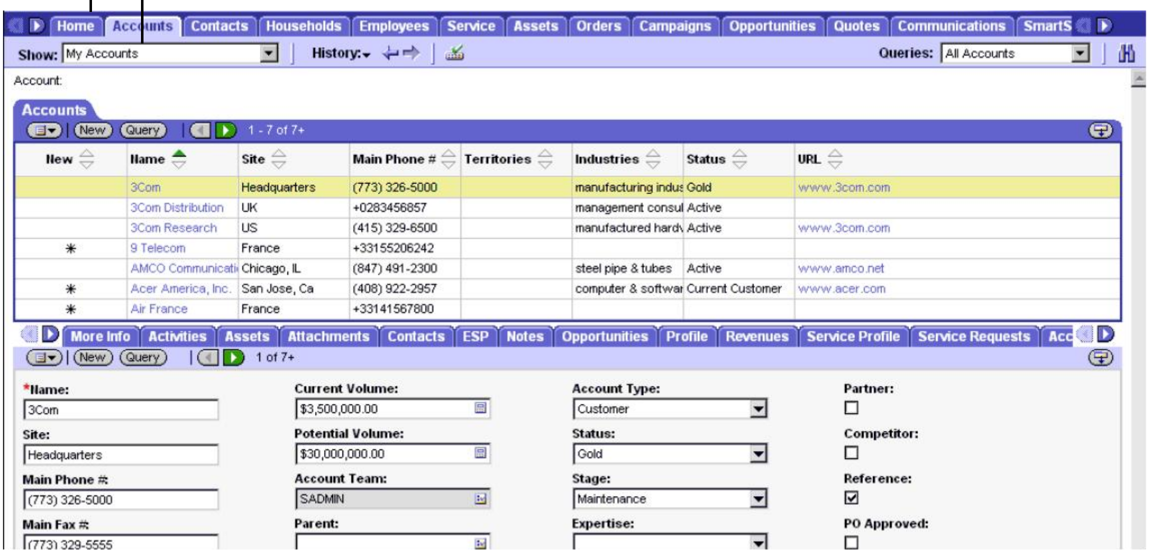

Applet
hyperlink Control or List Column
Drilldown Object
Dynamic Drilldown Destination
Destination Drilldown Object property
Business Component property
Hyperlink Field property
Field property
View property
Business Component
Field property
hyperlink Field
type Field
View
Legend:
object definition
contained (child) object definition
property specifies other object definitio

Dynamic Drilldown Configuration

14. Describe Screens in detail.

A *screen* is a logical collection of views. It is not a visual construct in itself; rather, it associates views so that other visual constructs, such as the Site Map and tab bar, can reflect the list of views contained in the currently active screen.

A screen does not have a direct relationship with a business object in the same way that a view does. No property in the Screen object type specifies a business object. However, a screen normally contains only views relating to the same business object; this is good design practice. In this sense, it can be loosely said that a screen corresponds to one business object.



New	Name	Site	Main Phone #	Territories	Industries	Status	URL
	3Com	Headquarters	(773) 326-5000		manufacturing indus	Gold	www.3com.com
	3Com Distribution	UK	+0283456857		management consul	Active	
	3Com Research	US	(415) 329-6500		manufactured hardv	Active	www.3com.com
*	9 Telecom	France	+33155206242				
	AMCO Communicati	Chicago, IL	(847) 491-2300		steel pipe & tubes	Active	www.amco.net
*	Acer America, Inc.	San Jose, Ca	(408) 922-2957		computer & softwar	Current Customer	www.acer.com
*	Air France	France	+33141567800				

*Name: 3Com	Current Volume: $3,500,000.00	Account Type: Customer	Partner: □
Site: Headquarters	Potential Volume: $30,000,000.00	Status: Gold	Competitor: □
Main Phone #: (773) 326-5000	Account Team: SADMIN	Stage: Maintenance	Reference: ☑
Main Fax #: (773) 329-5555	Parent:	Expertise:	PO Approved: □

Applications of a Screen:

Siebel applications are primarily a collection of screens that users can invoke from the desktop by double-clicking an icon or by pointing a browser to a server running the application. Each combination of screens that is appropriate to a specific class of users can be provided as an application. Siebel Sales, Siebel Service, and Siebel eMarketing are examples of applications. Custom applications can be configured as well, uniquely combining user interface object definitions to meet particular requirements of the organization.

In addition to collecting a group of screens and their views, an application object definition includes the following:

Find object definitions that configure the Find dialog box.

Scripts written in Siebel VB or Siebel eScript and browser JavaScript that can be implemented as event procedures on startup, prior to closing, and so on. These are implemented through Application Script child object definitions, and created and maintained in the Siebel VB or Siebel eScript Editor.

Custom menu options for Siebel-provided methods. These are implemented through the application method menu item object definitions, and created in the Applet Method Menu Item Wizard.

Configuring Application:

When configuring applications for deployment in High Interactivity mode, consider the following:

- Browser scripting is fully supported in High Interactivity mode.
- For fields to interpret and display custom HTML, such as a URL entered by the user, the field's Type property must be set to URL. If it is not set to URL, the HTML is presented and interpreted as plain text. For example, if a user typed a URL in a field of type TEXT, the URL would not be recognized as a link to a Web page.
- You cannot modify the appearance of the rich text editor.
- You cannot modify the background and text color of list applets.
- You cannot place method-invoking controls, such as the delete function, on every row in a list. Instead place a button that calls the method on the applet itself. The function will act on the selected record.

There are cases when an application's configuration file is set to run in High Interactivity mode and all the applets in a view are configured to support High Interactivity, but the view appears in Standard Interactivity mode. Reasons this might occur are:

- One of the applets is in the Query mode. Because High Interactivity implicitly supports query operations from the user interface, it does not support the explicit use of the Query mode.
- One of the applets is in the New mode and uses a New template that is different from the Edit template used in its default mode. This can be avoided by inactivating New templates associated with the applets used in High Interactivity applications. The framework will then default to using the Edit template itself to create new records.
- One of the list applets has multi-row edits or multi-row select enabled.
- One of the list applets is a hierarchical list applet.
- The view uses a template that shows applets in a catalog-style layout. None of the employee applications should be using this layout.
- A combo box picklist uses Long Lists or has an associated pick applet. For example, if you perform an action from a High Interactivity applet that causes a pick applet to be displayed, the pick applet will not be in High Interactivity mode.