Siebel Object Types Reference

Version 8.0

December 2006

ORACLE

Contents

Index

1 What's New in This Release

What's New in Siebel Object Types Reference, Version 8.0

Table 1 lists changes in this version of the documentation to support release 8.0 of the software.

Table 1. What's New in Siebel Object Types Reference, Version 8.0

Topic	Description
■ "Control" on page 139	Updated to include support for Applet Message.
■ "Applet Message" on page 35 ■ "Applet Message Locale" on page 36 ■ "Applet Message Variable" on page 36 ■ "Applet Message Variable Locale" on page 37	New object types added.
■ "Class User Prop" on page 128 ■ "Class User Prop Value" on page 129 ■ "Field Locale" on page 186 ■ "Task" on page 295	New object types added.
■ "Task Branch" on page 297 ■ "Task Branch Criteria" on page 298 ■ "Task Branch Criteria Value" on page 299 ■ "Task Chapter" on page 300	New object types added.
■ "Task Chapter Locale" on page 300 ■ "Task Event" on page 301 ■ "Task Event IO Argument" on page 302 ■ "Task Group" on page 303	New object types added.
■ "Task Group Item" on page 303 ■ "Task Group Locale" on page 304 ■ "Task Locale" on page 305 ■ "Task Metric" on page 305	New object types added.

Table 1. What's New in Siebel Object Types Reference, Version 8.0

Topic	Description
"Task Property" on page 306 "Task Step" on page 307 "Task Step Context" on page 311 "Task Step IO Argument" on page 312	New object types added.
"Task Step Locale" on page 313 "View Task Group" on page 332 "View Task Group Locale" on page 333 "WF Process Metric" on page 344	New object types added.

2 Using the Object Types Reference

The following sections explain how to get the best use of Object Types information:

- Conventions used in the object type descriptions. Read “Conventions” on page 15.
- Properties that appear in many object types are described in “Properties with Consistent Meanings” on page 16.
- Oracle’s Siebel Tools provides many object types that developers can modify. There are also internal object types that you should not modify. These are listed in “Nonconfigurable Object Types” on page 18.
- Several object types and properties are no longer visible in Siebel applications. These are listed in “Hidden Object Types and Properties” on page 21.

Conventions

The following conventions apply to object usage and the contents of this documentation:

- You must spell object names as they are listed in the documentation.
- A constant is case-sensitive and is usually spelled with all uppercase characters. Examples are TRUE and FALSE for Boolean fields.
- In property description tables, the Property column contains one of the following values:
  - (**R) Required.** You must make an entry for this property, either to complete data entry for the record or for the object definition to work correctly.
  - **(O) Optional.** You are not required to enter a value.
  - **(S) System.** The system supplies a value, which you cannot alter.

(H) marks object types and properties that are not used by the Web Client. These are hidden from view and are not listed in the Object Explorer. If an object type is hidden its properties are also hidden. To reveal hidden object types and hidden properties, change the ClientConfigurationMode argument in the configuration file for Siebel Tools (tools.cfg).

To access hidden object properties

1. Exit Siebel Tools.
2. Open the `tools.cfg` file and change the `ClientConfigurationMode` argument to `All`.
3. Save the file and restart Siebel Tools.

   The hidden object types and properties are now visible in Siebel Tools.

The following information details how to access object properties in Siebel Tools.

When first logging into Tools, the Object Explorer contains a listing of the most commonly used object types. The following instructions explain how to display more of the available object types.

To access object types

1. Start Siebel Tools.
2. Choose View > Options.

   The Development Tools Options dialog box opens.
3. Click the Object Explorer tab.

   The check boxes indicate which objects are visible. A box with a gray background indicates that at least one object within the hierarchy is not checked. A box with a white background indicates that all objects within its hierarchy are checked.
4. To make an object visible, check its box.
5. To make your changes active, click OK.

   The Object Explorer now contains the object types you wish to use.

Properties with Consistent Meanings

The following properties have consistent meanings across object types.

Property	Description
Application Code	Reserved for future use.
Changed **(S)**	If TRUE, indicates that changes have been made to the contents of the corresponding object definition or its children since a date and time set in Tools > Options. FALSE indicates that the object definition and its children are unchanged since that time.
Comments **(O)**	Provides a place for the developer to add descriptive information.
GParent, GGParent **(S)**	Name of the grandparent or great-grandparent object definition. This is a system-supplied value. This can be seen for many types of objects in the Object List Editor if you select the Flat tab of the Object Explorer.
Inactive **(O)**	Value is TRUE or FALSE. If TRUE, prevents this object definition from being compiled when you compile your Siebel repository file and being accessed by any other object definition.
Language Code **(R)**	The abbreviation of the language being used for the object type. For example, ENU stands for U.S. English.
Module	Reserved for future use.

Property	Description
Needs High Interactivity **(O)**	A TRUE or FALSE value indicating whether high interactivity is being set for an object type. It is recommended that this property not be changed from the default value. Any changes could affect the productivity of the application. Generally applies to Browser Script objects.
Object Language Locked	If an object is locked, this indicates the operating system of the computer the developer is using.
Object Locked	A TRUE or FALSE value indicating if a specific object is locked by a developer.
Object Locked by Name	The user name of the developer who has locked a project either by checking out of the server or locking directly.
Object Locked Date	The date the object was locked either by checking out of the server or by locking the object directly in the UI.
Parent Name, Parent *object_defn_name* **(S)**	Name of the current object definition's parent. This is a system-supplied value. Note that the Parent Business Component property in the Link object type is different from the typical Parent *object_defn_name* property. It is a user-entered, required property. This can be seen for many types of objects in the Object List Editor if you select the Flat tab of the Object Explorer.
Project **(R)**	Identifies the project that contains the object definition.
Redo **(O)**	Single-level redo of a field change or a record change. Used by the Locale Management Utility for tracking locale- specific changes.
Repository Name **(S)**	Identifies the repository that contains the object definition.
String Override **(O)**	A language-independent override of the String Reference. If an override exists, then it is used in place of the String Reference. This appears as <property_name>- String Override. Example: Display Name - String Override.
String Reference **(R)**	A pointer to a symbolic string. This appears as <property_name>- String Reference, for example: Display Name - String Reference. Example: For the Account List Applet the value of Title - String Reference is SBL_ACCOUNTS-1004224727-0H5. In the Symbolic String Locale object, there is a record for this string reference. For the ENU language, the string value is "Accounts".

Property	Description
Translate **(S)**	Used by the Locale Management Utility for tracking localization changes. For Oracle use only.
Upgrade Ancestor **(S)**	The property is used for cases when the developer copies a business component (or applet or report) from an existing version. The property is set on the copied object to the name of the original that it was copied from. For example, if a copy is made of the Account business component called *ESG Account* you would set the Upgrade Ancestor property of the ESG Account business component to *Account*. This instructs the Application Upgrader to upgrade the Sub Account business component in the same way it would upgrade the Account business component.

Nonconfigurable Object Types

The object types listed in this topic are used to manage the repository, or are for use by Oracle only and are not available for customer developers to configure.

- "Object Types Used by the Siebel Enterprise Integration Manager" on page 18
- "Objects Related to Information Access" on page 19
- "Server Component Objects" on page 19
- "Miscellaneous Objects" on page 20

CAUTION: Do not modify any of these object types. Any modification can adversely affect the performance and operation of your implementation.

Object Types Used by the Siebel Enterprise Integration Manager

The following object types are used by the Siebel Enterprise Integration Manager:

- "Attribute" on page 85
- "Attribute Mapping" on page 85
- "EIM Explicit Primary Mapping" on page 168
- "EIM Interface Table" on page 168
- "EIM Interface Table Column" on page 169
- "EIM Table Mapping" on page 172
- "Foreign Key Mapping" on page 191
- "Foreign Key Mapping Column" on page 192
- "Interface Table User Key Usage" on page 211

- “User Key” on page 322
- “User Key Attribute” on page 323
- “User Key Attribute Join” on page 323
- “User Key Attribute Tree Node” on page 324
- “User Key Column” on page 324

See Also
“Nonconfigurable Object Types” on page 18

Objects Related to Information Access

The following objects are for information access:

CAUTION: Do not modify any of these object types. Any modification can adversely affect the performance and operation of your implementation.

- “Dock Object” on page 156
- “Dock Object Visibility Rule” on page 159
- “Dock Object Table” on page 158
- “Dock Object Related DObj” on page 157

See Also
“Nonconfigurable Object Types” on page 18

Server Component Objects

The following objects are related to server components:

- “Server Component Event Subtype” on page 276
- “Server Component Event Type” on page 277
- “Server Component Parameter” on page 277
- “Server Component State Value” on page 279
- “Server Component Statistic” on page 279
- “Server Component Subsystem” on page 280
- “Server Component Type” on page 280

See Also
“Nonconfigurable Object Types” on page 18

Miscellaneous Objects

The following objects do not belong in any other categories:

- “Class” on page 125
- “Class Method” on page 127
- “Class Method Menu Item” on page 127
- “Class Method Menu Item Locale” on page 128
- “DLL” on page 155
- “Repository” on page 255
- String (H)
- String Map (H)
- “Type” on page 321

See Also

“Nonconfigurable Object Types” on page 18

Hidden Object Types and Properties

The object types and properties listed in this topic are not used in Web Client mode. These object types and properties are hidden in Siebel Tools.

NOTE: To make these object properties visible, in the Tools configuration file, set `ClientConfigurationMode = All`. These object properties are denoted with (H) in Siebel Object Types on page 27. For more information on accessing object properties, read Conventions on page 15.

Object Type	Property
Applet	■ Background Bitmap ■ Background Bitmap Style ■ Browser Class ■ Height ■ HTML Popup Dimension ■ Popup Dimension ■ Text Style ■ Width
Applet Script (H)	■ All properties
Applet Web Template	■ Expression
Application	■ Text Style
Application Method Menu Item (H)	■ All properties
Application Method Menu Item Locale (H)	■ All properties
Application Toolbar (H)	■ All properties
Assignment Attribute (H)	■ All properties
Assignment Attribute Column (H)	■ All properties
Assignment Criteria (H)	■ All properties
Assignment Criteria Attribute (H)	■ All properties
Assignment Criteria Attribute Locale (H)	■ All properties
Assignment Object (H)	■ All properties
Assignment Object Extension (H)	■ All properties
Assignment User Prop (H)	■ All properties
Bitmap	■ Data ■ Transparent Color
Bitmap Category	■ Predefined
Business Component	■ Browser Class

Object Type	Property
Business Service	■ Browser Class
Chart	■ Compare
Chart Element	■ Font Effect ■ Font Style
Class	■ Java Thin Client ■ Thin Client ■ UNIX Support
Command	■ Bitmap

Object Type	Property
Control	■ ActiveX Bind Property
	■ ActiveX Properties
	■ Automatic Horizontal Scroll
	■ Automatic Vertical Scroll
	■ Background Color
	■ Bitmap
	■ Default Button
	■ Foreground Color
	■ Group
	■ Height
	■ Height - Language Override
	■ HTML Default Control
	■ Left
	■ Left - Language Override
	■ Left Text
	■ Multi Line
	■ Owner Draw
	■ Popup Edit
	■ Sequence
	■ Sequence - Language Override
	■ Tab Stop
	■ Text Style
	■ Tooltip Text
	■ Top
	■ Top - Language Override
	■ Type
	■ Vertical Scroll
	■ Want Return
	■ Width
	■ Width - Language Override

Object Type	Property
Control Locale	■ Height ■ Left ■ Sequence ■ Tooltip Text ■ Top ■ Width
Dynamic Candidate (H)	■ All properties
Dynamic Candidate Component (H)	■ All properties
Dynamic Candidate Component Col (H)	■ All properties
Dynamic Candidate Locale (H)	■ All properties
External Search Engine (H)	■ All properties
List	■ Alpha Tab Search Field ■ Alpha Tab Search Field - Language Override ■ Variable Row Height
List Column	■ Bitmap Column Heading ■ Check Bitmap Identifier ■ Popup Edit ■ Sequence ■ Sequence - Language Override ■ Type ■ Width ■ Width - Language Override
List Column Locale	■ Sequence ■ Width
List Locale	■ Alpha Tab Search Field
Page Tab	■ Bitmap Category
Screen	■ Bitmap Category ■ Unrestricted Viewbar
Screen View	■ Six Sectors ■ Eight Sectors
Search Engine Field (H)	■ All properties
Search Engine Field Locale (H)	■ All properties

Object Type	Property
Search Engine Pick View (H)	■ All properties
Search Engine Table (H)	■ All properties
Search Engine Table Locale (H)	■ All properties
String	■ All Properties
String Map	■ All Properties
Text Style (H)	■ All properties
Toolbar	■ Client Restriction
Tree Node	■ Bitmap Category ■ Bitmap Index ■ Bitmap Index Field ■ Selected Bitmap Index ■ Selected Bitmap Index Field
View	■ Background Bitmap ■ Bitmap Category ■ Container Web Page ■ Drop Sectors ■ No Borders ■ Sector0 Applet ■ Sector1 Applet ■ Sector2 Applet ■ Sector3 Applet ■ Sector4 Applet ■ Sector5 Applet ■ Sector6 Applet ■ Sector7 Applet ■ Status Text ■ Text Style ■ Vertical Line Position
View Locale	■ Status Text

NOTE: Your Siebel application implementation may not have all features described in this documentation, depending on which software modules you have purchased.

3 Siebel Object Types

Double-click the Siebel Object Types book in the Contents tab to see the list of tags.

Accelerator

Siebel Object Types > Command > Accelerator

Defines an accelerator key sequence for each browser platform for a menu item that uses this command.

Properties

Property	Description / Usage	Valid Values / Examples
Browser Platform **(O)**	User Agent. User agents are typically versions of browsers, such as Internet Explorer 5.0, Internet Explorer 5.5, and Netscape 6.0.	All — Applies to all versions of a particular browser. Basic — Defines the capabilities of a major version of a particular browser. Extended — Defines the extended capabilities of the various minor versions of the same browser.
Display Name **(O)**	The series of keyboard strokes making up a shortcut and how it appears on a menu.	Example: CTRL+N.
Key Sequence **(O)**	The series of keyboard strokes making up a shortcut.	Read “Accelerator Creation” on page 28.
Key Sequence - Language Override **(O)**	Specifies the language code used for the current key sequence. This may be overridden with an Accelerator Locale child object.	
Name **(R)**	The name of the accelerator.	Example: New record.

See Also

“Accelerator Locale” on page 28
“Command” on page 135

Accelerator Creation

Siebel Object Types > Command > Accelerator

Specify an accelerator using the following format:

```
Accelerator: | modifiers "+" name | modifier + name "," accelerator

Modifiers: | modifier | modifier "+" modifiers

Modifier: Ctrl | Alt | Shift

Name: 0-9 | A-Z | F{0-24} | key_code | constant
```

For key_code, use a three-symbol decimal code such as “128” or a four-symbol hex code such as “0x10”.

```
Constant: INSERT | DEL | LEFT | RIGHT | UP | DOWN | HOME |
     END | PAGEUP | PGUP | PAGEDOWN | PGDN | SPACE | TAB | ENTER
```

The syntax is not case-sensitive:

Examples:

- Ctrl+F10, Alt+A
- Ctrl+ INSERT
- Ctrl+Alt+0x221

Your accelerator is used if there is a menu or toolbar item defined with the command field containing the command with your accelerator.

Accelerator Locale

Siebel Object Types > Command > Accelerator > Accelerator Locale

Represents language-specific overrides used with the Accelerator object type.

Properties

Property	Description / Usage	Valid Values / Examples
Display Name **(O)**	The series of keyboard strokes making up a shortcut for a specific language and how it appears on a menu.	Example: CTRL+N.
Key Sequence **(O)**	The series of keyboard strokes making up a shortcut for a specific language.	
Name **(O)**	The abbreviation of the language being used by the application.	Example: ENU.

See Also
“Accelerator” on page 27
“Command” on page 135

Applet

Siebel Object Types > Applet

Defines a list or form applet (or a specialized applet type such as chart applet, Web applet, tree applet, or file attachment applet) to be included in views. The applet is responsible for displaying the data to the user and contains the control or list column layout information (through child object definitions).

You can add user properties to applets.

Properties

Property	Description/Usage	Valid Values/Examples
Associate Applet **(O)**	The name of an applet used to relate data from two separate entities (a many-to-many relationship). Rows from entity B (association applet) can be associated (many-to-many relationship) with the selected row from entity A (base applet).	An applet of type “association list.”
Auto Query Mode **(O)**	Allows for automatically showing the applet in query mode when New Query is entered as the value.	This property can have the following values: New Query — When creating the applet, clears the business component’s search specification, clears the business object’s predefined query, and sets the business component to Query mode. Automatically shows the applet in Query mode. Refine Query — Retains existing search specifications and predefined queries. Automatically shows the applet in Query mode.
Background Bitmap **(O)(H)**	The bitmap object definition to display in the background of the applet.	Any Applet Background bitmap definition.

Property	Description/Usage	Valid Values/Examples
Background Bitmap Style **(O)(H)**	Setting indicating how the background bitmap is arranged in the applet.	Center — Display once in the middle. Stretch — Display once stretched to cover the applet area. Tile — Display repeatedly, edge to edge, to cover the applet area.
Browser Class **(O)(H)**	Not used.	Not applicable.
Business Component **(O)**	The name of the business component whose data is to be displayed. The business component must be defined in the business object specified for any view that uses this applet.	
Class **(O)**	The name of a C++ class used to manage the applet. Specialized applet classes are a subclass of one of the standard classes.	Standard classes: ■ CSSFrame (form applets). ■ CSSFrameChart (chart applets). ■ CSSFrameList (list applets). ■ CSSFrameTree (tree applets). ■ Examples of specialized classes: ■ CSSFrameListAttachment. ■ CSSFrameListActivity.
Height **(R)(H)**	The vertical size of the applet in sector units.	1, 2, 3, or 4.
Help Identifier **(O)**	Not used.	Not applicable.
HTML Number of Rows **(O)**	Denotes the number of rows to be displayed at a time when rendering a list applet. Applies only to list applets, which show rows of data. Used by the Siebel Web Engine.	Default is 7 rows. Read "HTML Number of Rows" on page 33 for details.

Property	Description/Usage	Valid Values/Examples
HTML Popup Dimension **(O)(H)**	Dimension, in pixels, of the pop-up window. Used by the Siebel Web Engine.	An example is 640x480 (specified with a small x and without blank spaces).
Insert Applet **(O)**	The name of a pop-up applet to be displayed for creation of a new record. Used with dynamic insert.	Any valid pop-up applet name.
Insert Position **(O)**	The position in which new records are created.	BEFORE — Before the current record. AFTER — After the current record. FIRST — The first record. LAST — The last record.
Mail Address Field **(O)**	A default email recipient in the Message Properties dialog box. The list of recipients is set up in user properties for the applet.	Example: Email Address (field in Business Component).
Mail Template **(O)**	A text file that is used to fill in the default email message. Can contain placeholders in the form "%field%" to provide for value substitutions of fields.	The list of available mail templates is determined from the contents of the MSGTEMPL folder of the installation directory (C:\Siebel for most users).
Name **(R)**	The name of the applet. All references to the applet are made through its name. The name can contain spaces and must be unique among applets. The name cannot contain parentheses.	Account Form Applet.
No Delete **(O)**	A TRUE or FALSE value that places additional security on top of the business component in a particular usage.	TRUE — No row deletion allowed. FALSE — Row deletion allowed.
No Insert **(O)**	A TRUE or FALSE value that places additional security on top of the business component in a particular usage.	TRUE — No row insert allowed. FALSE — Row insert allowed.

Property	Description/Usage	Valid Values/Examples
No Merge **(O)**	A TRUE or FALSE value that places additional security on top of the business component in a particular usage.	TRUE — No row merge allowed. FALSE — Row merge allowed.
No Update **(O)**	A TRUE or FALSE value that places additional security on top of the business component in a particular usage.	TRUE — No field updates allowed. FALSE — Field updates allowed.
Popup Dimension **(O)(H)**	The size of the pop-up applet. This text property is valid only for association, pick, MVG, and detail applets.	*W X H* where: *W* — Width in pixels. *H* — Height in pixels. Example: 512 X 220. Blank spaces and the capital X are part of the syntax.
Scripted **(S)**	A TRUE or FALSE value indicating whether the applet has a Siebel VB or Siebel eScript script attached to it.	TRUE — Siebel script attached. FALSE — No Siebel script attached.
Search Specification **(O)**	A conditional expression used to restrict the records retrieved from the applet's business component. For more information, read "Search Specification" on page 33.	Example: [Service Flag] = 'Y'.
Text Style **(O)(H)**	Defines the named text style to be used by default by all controls in the applet that do not specify their own text style. If not specified, the text style of the view is used.	
Title **(O)**	The text used for the title of the applet.	Accounts.
Type **(O)**	The type of the applet.	Valid values are Association List, Detail, MVG, Pick List, and Standard.

Property	Description/Usage	Valid Values/Examples
Upgrade Behavior **(R)**	This property is set by Oracle and can not be modified. Indicates if the object should be ignored if the repository merge is run with the Incorporate Custom Layout option.	Admin or Null Admin indicates that the object should be ignored. Null indicates that the object is to be processed by the Incorporate Custom Layout option.
Width **(R)(H)**	The horizontal size of the applet in sector units.	Valid values are 1 and 2.

HTML Number of Rows

If this property is not specified for the applet, the application's CFG file is checked. In the CFG file, the default number of rows for list applets belonging to that application can be specified using the NumberOfListRows property. If this property is also not specified, the property defaults to ten.

However, in some areas of the application, the default specified for this property has been set to seven. So for those areas, you might see a default of seven rows instead of ten.

Search Specification

An applet search specification cannot be used to override the search specification of the underlying business component, if the business component has one. Rather than overriding the business component's search specification, the applet's search specification is appended to that of the business component.

Siebel applications do not requery, and therefore keep the existing search specification, if you navigate to an applet without an applet search specification. There are two exceptions. The first is for child applets. An empty applet search specification is applied when it is on a child applet. The second is when the visibility mode changes for the parent applet. In this case, the applet level search specification should always be applied regardless of whether it is empty.

If two applets are defined with the same business component and both have a search specification, then the search specification for the applet in lowest sector is used.

Do not specify different search specifications where several applets in one view are based on the same business component. All of these applets should have the same or an empty search specification.

The applet search specification was not applicable to MVG and association applets in versions prior to 5.0 but is applicable in subsequent versions.

The search specification can change the query context for users performing queries.

See Also
"Applet Method Menu Item" on page 37
"Applet Locale" on page 35
"Applet Script (H)" on page 39
"Applet Toggle" on page 40
"Chart" on page 112
"Control" on page 139
"List" on page 220
"Text Style (H)" on page 313
"Tree" on page 318
"View" on page 325
"Hidden Object Types and Properties" on page 21

Applet Browser Script

Siebel Object Types > Applet > Applet Browser Script

Defines the browser script used by an Applet object type.

Define Applet Browser scripts by right-clicking on the applet and selecting Edit Browser Scripts.

For more information, read *Siebel Object Interfaces Reference*.

Properties

Property	Description / Usage	Valid Values / Examples
Browser Group **(O)**	Not currently used.	
Name **(S)**	Unique name for the script record. Do not modify this property. This is generated by Siebel Tools by combining the User Agent and Procedure properties.	Example: Applet_ChangeRecord_All.
Procedure **(O)**	Refers to the name of the applet that is used in upgrading the current applet during the next application upgrade (repository merge). In the majority of cases, the "ancestor" applet as specified by this property is the one from which the new applet is being replicated.	
Script **(O)**	Text for the function.	

See Also
"Applet" on page 29

Applet Locale

Siebel Object Types > Applet > Applet Locale

Represents language-specific overrides used with the Applet object type.

Properties

Property	Description / Usage	Valid Values / Examples
Name **(R)**	The abbreviation of the language being used by the application.	Example: ENU.
Title **(O)**	The text used for the title of the applet for a specific language.	

See Also

"Applet" on page 29

Applet Message

Siebel Object Types > Applet > Applet Message

The applet message allows static text to be combined with dynamic data (such as values from business component fields). When bound to an applet control, the message is displayed as a read-only, translated string with the dynamic data substituted. For more information about configuring an applet message, see the *Siebel Business Process Framework: Task UI Guide*.

Properties

Property	Description / Usage	Valid Values / Examples
Name **(R)**	The name for the Applet Message. This name is used in binding the applet message to a control, using the Field picklist.	
Text Message **(R)**	Text with a reference to a symbolic string that contains the text and placeholders for the dynamic data to be substituted. The placeholders are of the format "%N", where N is numerical.	Example: Here is the tracking number for your case: %1
Text Message - String Override	Override of symbolic string for Text Message.	

See Also
“Applet” on page 29
“Applet Message Locale” on page 36
“Applet Message Variable” on page 36
“Control” on page 139

Applet Message Locale

Siebel Object Types > Applet > Applet Message > Applet Message Locale

Represents language-specific overrides used with the Applet Message object type.

Properties

Property	Description/Usage	Valid Values/Examples
Language Code **(R)**	The abbreviation of the language being used by the application.	Example: ENU
Text Message **(R)**	The string used for the Applet Message.	

See Also
“Applet” on page 29
“Applet Message” on page 35
“Applet Message Variable” on page 36

Applet Message Variable

Siebel Object Types > Applet > Applet Message > Applet Message Variable

List of business component fields and their corresponding numerical key. The numerical key is used in determining which placeholder to replace in the parent Applet Message.

For example if the applet message is "Here is the tracking number for your case: %1", then the Applet Message Variable is defined for business component field "Case Id" with a value of "1".

At runtime, if the value of [Case Id] is "12-ABCX", the value of the Applet Message is: "Here is the tracking number for your case: 12-ABCX".

Properties

Property	Description / Usage	Valid Values / Examples
Field **(R)**	A field from the business component that the applet is bound to.	
Field - Language Override **(O)**	A field from the business component that the applet is bound to.	
Value **(R)**	For the field value to be substituted at runtime, this must correspond to one of the placeholders in the text of the parent Applet Message.	The value of this property is a Numeric value.

See Also

"Applet" on page 29
"Applet Message" on page 35
"Applet Message Variable Locale" on page 37

Applet Message Variable Locale

Siebel Object Types > Applet > Applet Message > Applet Message Variable > Applet Message Variable Locale

Represents language-specific overrides used with the Applet Message Variable object type.

Properties

Property	Description / Usage	Valid Values / Examples
Field **(O)**	The name of an alternative field to use when the value of the Name property is the active locale.	
Name **(R)**	The abbreviation of the language being used by the application.	Example: ENU.

See Also

"Applet" on page 29
"Applet Message" on page 35
"Applet Message Variable" on page 36

Applet Method Menu Item

Siebel Object Types > Applet > Applet Method Menu Item

Defines the applet-specific menu that is available when the applet is activated.

NOTE: You must write script in the PreInvokeMethod event handler to test for the method string and take some action. If the test code is not present, an error occurs. For more information, read *Siebel Object Interfaces Reference*.

Properties

Property	Description/Usage	Valid Values/ Examples
Command **(O)**	Specifies which invoke method is called when an applet-level menu item associated with the command is executed.	
Menu Text **(R)**	The text displayed in the menu item.	
Name **(O)**	Same as position property.	Integer value.
Position **(R)**	The sequence of the menu item in the single-level list of menu items.	Integer value.
Suppress Menu Item **(O)**	Default is FALSE. If TRUE, causes the class-level menu item of the specified name to be removed from the applet-level menu in the applet where this property is specified.	TRUE — Menu item is suppressed. FALSE — Menu item is not suppressed.

Menu Editor

Menus should be edited by right-clicking on the applet and selecting Edit Web Menus.

See Also

"Applet" on page 29
"Applet Script (H)" on page 39

Applet Method Menu Item Locale

Siebel Object Types > Applet > Applet Method Menu Item > Applet Method Menu Item Locale

Represents language-specific overrides used with the Applet Method Menu Item object type.

Properties

Property	Description/Usage	Valid Values/ Examples
Menu Text **(R)**	The text displayed in the menu item for a specific language.	
Name **(O)**	The abbreviation of the language being used by the application.	Example: ENU.

See Also

"Applet" on page 29
"Applet Method Menu Item" on page 37

Applet Script (H)

Siebel Object Types > Applet > Applet Script (H)

Defines a script (written in Siebel VB or eScript) implemented in an applet.

In version 7.0, this object type is replaced by the Applet Browser Script and the Applet Server Script objects.

Properties

Property	Description/Usage	Valid Values/ Examples
Name **(R)**	The name of the script.	
Program Language **(O)**	A value maintained by the Siebel application indicating whether the programming language of the script is Siebel VB or Siebel eScript.	SBL — Siebel VB. JS — Siebel eScript.
Script **(R)**	The body of the script.	Upper limit is 16 K.
Sequence **(R)**	The order of the script in compilation relative to the other scripts for the object.	Integer value.

See Also

"Applet" on page 29
"Application Server Script" on page 51
"BusComp Server Script" on page 89
"Hidden Object Types and Properties" on page 21

Applet Server Script

Siebel Object Types > Applet > Applet Server Script

Script associated with a server applet. For more information, read *Siebel Object Interfaces Reference*.

Applet server scripts are edited by right-clicking on the applet and selecting Edit Server Scripts.

As of version 7.0, the Applet Server Script object type replaces the Applet Script and the Applet Web Script objects.

Properties

Property	Description/Usage	Valid Values/ Examples
Name **(R)**	The name of the script.	
Program Language **(O)**	A value maintained by the Siebel application indicating whether the programming language of the script is Siebel VB or Siebel eScript.	SBL — Siebel VB. JS — Siebel eScript.
Script **(R)**	The body of the script.	Upper limit is 16 K.
Sequence **(R)**	The order of the script in compilation relative to the other scripts for the object.	Integer value.

See Also

"Applet" on page 29

Applet Toggle

Siebel Object Types > Applet > Applet Toggle

Identifies one or more alternate applets to use in displaying the data of a business component.

Properties

Property	Description/Usage	Valid Values/Examples
Applet **(R)**	The name of the applet that can be displayed (static toggles).	Example: Account Entry Form.
Auto Toggle Field **(O)**	The test field in the business component of the applet (dynamic toggles). The current value in this field is checked against the value in the Auto Toggle Value property.	Example: Find Category.

Property	Description/Usage	Valid Values/Examples
Auto Toggle Value **(O)**	The test value to match in the test field (dynamic toggles). When this value is matched in the test field, the corresponding applet is activated.	Example: Smart Answer.
Name **(R)**	The name of the applet to be displayed.	Example: Smart Answer Find Entry.
Sequence **(O)**	The order of this applet relative to other applets in the toggle cycle (dynamic toggles).	Integer value.

See Also

"Applet" on page 29

Applet User Prop

Siebel Object Types > Applet > Applet User Prop

Communicates a value to C++ code that implements specialized applet behavior. The values of the applet user properties can be changed at configuration time. These values persist in the repository and in the Siebel repository file.

Properties

Property	Description/Usage	Valid Values/Examples
Name **(R)**	The name of the user property.	
Value **(R)**	The value of the user property.	

Changing Undocumented User Properties

Developers can change user properties that have been documented. For example, user properties for applets of class CSSFrameBrowser. Only internal Oracle developers who create or change specialized C++ code should create or delete user properties that are not documented. For more information on supported, configurable Applet User Prop, see the *Siebel Developer's Reference*.

See Also

"Applet" on page 29
"Business Component User Prop" on page 101

For information about user properties, read *Siebel Developer's Reference.*

Applet Web Template

Web templates to which an applet is associated. This object has child objects—Applet Web Template Items—that define the mappings between controls or list columns in the applet, and placeholder tags (identified by IDs) in the template file.

Identifies external HTML (or other markup language) files that define the layout and Siebel Web Engine interactions for an applet.

Properties

Property	Description/Usage	Valid Values/Examples
Expression **(O)(H)**	Not used.	
Name **(R)**	The name of this Applet Web Template association.	
Sequence **(O)**	(Edit templates only) Used when users have multiple Edit pages—in this case each one is identified by this sequence number. Used by the Siebel Web Engine.	Integer value. The starting/default value for Web template sequence is 0. If more than one template exists with the same type, then use sequence 0, 1, 2...
Type **(R)**	Mode for which this Applet Web Template is to be used. Used by the Siebel Web Engine.	Base — Renders the applet in standard read-only mode. Edit — Used when the user invokes the EditRecord method. Modifies existing data in a record. Edit List — Can be used instead of the BASE mode for list applets. Renders a list applet in a mode where the selected row or column is editable. New — Used when the user invokes the NewRecord method. Enters data into new records. Query — Used when the user invokes the NewQuery method. Shows the fields to be searched.
Web Template **(R)**	The Web template to be used to show the applet in the corresponding mode. Used by the Siebel Web Engine.	

See Also
"Applet" on page 29
"Applet Server Script" on page 40
"View Web Template" on page 334

Applet Web Template Item

Siebel Object Types > Applet > Applet Web Template > Applet Web Template Item

Provides a mapping of controls or list columns to placeholder IDs. In Siebel Tools you create an item, enter the ID, and select the control and type for the mapping. This can be done manually using the Object List Editor or visually using the Web Applet Editor.

Defines a control, list item, or special Web control in the Web implementation of an applet.

Properties

Property	Description/Usage	Valid Values/Examples
Column Span **(O)**	Do not modify manually. Valid for grid-based form applets that indicate the number of cells (each cell is 8x8 pixels) for the item's width.	Integers that are likely to be less than 120.
Column Span - Language Override **(O)**	Do not modify manually. For items on a grid-based applet web template, this provides the ability to override the Column Span property (the width of the item) when a particular language locale is active.	Integers that are likely to be less than 120.
Control **(R)**	The name of the control or list column. Used by the Siebel Web Engine.	Example: Button Reorder.
Expression **(O)**	Specifies the condition under which this item is mapped.	Example: Not Siebel Sales.
Grid Property **(R)**	Defines the type of the item that exists on a grid-based form applet. This is set automatically when editing in the grid layout editor.	FormattedLabel and FormattedHtml.
Item Identifier - Language Override **(O)**	The language used for the item identifier that is different from the language being used by the application.	

Property	Description/Usage	Valid Values/Examples
Item Identifier **(R)**	ID of the swe:control tag. Used by the Siebel Web Engine to associate a control with a particular position within a web template. This property is populated by dragging and dropping controls within the Web Applet Layout Editor.	
Mode **(O)**	Used for the More/Less feature. Feature provides the ability for an applet to display initially with a limited set of fields and expand to display a larger number when the more/less button is clicked. Provides the ability to specify for each control whether the control appears always or only in More mode. A combo box appears in the toolbar allowing a developer the ability to switch between modes when editing a layout.	Valid values are More and Less. Less — Each control created in this mode is tagged with Less in this property field. More — Each control created in this mode is tagged with More in this property field.
Name **(R)**	The name of the item. Same as the control property.	
Namespace - Language Override **(O)**	Not used in this release.	
Namespace **(O)**	Not used in this release.	
Row Span **(O)**	Do not modify manually. Valid for grid-based form applets that indicate the number of cells (each cell is 8x8 pixels) for the item's height.	Integers that are likely to be less than 100.
Row Span - Language Override **(O)**	Do not modify manually. For items on a grid-based applet web template, this provides the ability to override the Row Span property (the height of the item) when a particular language locale is active.	Integers that are likely to be less than 100.
Type **(R)**	The type of the item. Used by the Siebel Web Engine.	Valid values are Control, List Item, and Web Control.

See Also
"Applet" on page 29
"Web Template" on page 340
"View Web Template Item" on page 335

Applet Web Template Item Locale

Siebel Object Types > Applet > Applet Web Template > Applet Web Template Item > Applet Web Template Item Locale

Represents the language-specific overrides used with the Applet Web Template Item object type.

Properties

Property	Description/Usage	Valid Values/ Examples
Column Span **(O)**	Provides the ability to change the Column Span property when a particular language or locale is active. This is linked directly to the Column Span Language Override property for an Applet Web Template Item.	
Item Identifier **(O)**	The ID assigned to the control, which is mapped to the control identified in the SWT file for a specific language.	
Name **(R)**	The abbreviation of the language being used by the application.	Example: ENU.
Namespace **(O)**	Not used in this release.	
Row Span **(O)**	Provides the ability to change the Row Span property when a particular language or locale is active. This is linked directly to the Row Span Language Override property for an Applet Web Template Item.	

See Also
"Applet Method Menu Item" on page 37

Application

Siebel Object Types > Application

Defines which screens and views accessible through menus and page tabs for an application. An application object definition is also used to define application-level scripts and find object definitions. An application is created by adding a new application object definition and adding the appropriate entries to each of its child types. These object types include Application Find, Application Menu Item, Application Script, Page Tab, and Screen Menu Item. An Application object definition is also used to define application-level scripts and Application Find object definitions.

Properties

Property	Description/Usage	Valid Values/ Examples
Acknowledgement Web Page **(O+)**	The name of the Web page to show after the user logs in. Used only if the Acknowledgement Web View property is not set. The Acknowledgement Web Page has precedence over the Acknowledgement Web View. (+) Either Acknowledgement Web Page or Acknowledgement Web View is required. Used by the Siebel Web Engine.	
Acknowledgement Web View **(O+)**	The name of the view to show after the user logs in. The Acknowledgement Web Page has precedence over the Acknowledgement Web View. (+) Either Acknowledgement Web Page or Acknowledgement Web View is required. Used by the Siebel Web Engine.	
Company Logo Bitmap **(O)(H)**	The name of a bitmap object definition in the Company Logos bitmap category.	
Container Web Page **(O)**	The page template within which all views, pages, or applets is displayed. This page contains the common user interface components such as viewbars, screenbars, logos, and so on. Used by the Siebel Web Engine.	By default, all views, pages, and applets use this page. For views, it can be overridden by the Container Page property of the View object.

Property	Description / Usage	Valid Values / Examples
Default Find **(O)**	Setting this property allows you to set a Find object as the default Find. The Find object appears in the Look In drop-down list in the Search Center, given the Default Find value of the Application Find object is set to True.	
Display Name **(O)**	Name displayed on menus and page tabs for an application.	
Error Web Page **(O)**	Error page to show when an error occurs in the application. Used by the Siebel Web Engine.	Must be of the Web Page object type.
Login Web Page **(O)**	The page where the user fills in a user name and password to log on. Used by the Siebel Web Engine.	Must be of the Web Page object type.
Logoff Acknowledgement Web Page **(O)**	The page to show after the user logs off. Used by the Siebel Web Engine.	Must be of the Web Page object type.
Menu **(O)**	The menu object definition that is used by the application.	
Name **(R)**	A unique application name.	
Scripted **(S)**	A TRUE or FALSE value indicating whether the application has a Siebel VB or Siebel eScript script attached to it.	TRUE — Siebel script attached. FALSE — No Siebel script attached.
Sort Web Page **(O)**	The page where the user can choose which fields to sort on in list applets. Used by the Siebel Web Engine.	Must be of the Web Page object type.
Text Style **(O)(H)**	The named text style to be used by default by all controls in the application, if the control, applet, and view do not explicitly specify a text style.	Default: The value from the CFG file.

NOTE: Siebel Tools, as an application, should not be configured like other Siebel applications, such as Siebel Sales and Siebel Service. Even though the views and applets for Siebel Tools exist in the repository, you must not modify this information.

See Also

"Application Find" on page 49
"Application Method Menu Item (H)" on page 50
"Application Server Script" on page 51
"Hidden Object Types and Properties" on page 21
"Page Tab" on page 242
"Screen Menu Item" on page 258

Application Browser Script

Siebel Object Types > Application > Application Browser Script

Defines the browser script used by an Application object type. For more information, see the *Siebel Object Interfaces Reference*.

Application browser scripts are edited by right-clicking on the application and selecting Edit Browser Scripts.

Properties

Property	Description/Usage	Valid Values/ Examples
Browser Group **(O)**	Browser type for which this script has been written.	
Name **(R)**	Unique name for the script record. Do not modify this property. This is generated by Siebel Tools by combining the User Agent and Procedure properties.	
Procedure **(O)**	Name of function or procedure.	
Script **(O)**	This is the text for the function.	

See Also

"Application" on page 45

Application Event Services

Siebel Object Types > Application > Application Event Services

Defines which application event services used by the application.

Properties

Property	Description / Usage	Valid Values / Examples
Event Name **(R)**	The name of the application event service.	Start is the only legal value.
Method Name **(R)**	The name of the method to invoke on the business service.	Start is the only legal value.
Name **(R)**	The description of the event.	
Sequence **(R)**	The order in which the services are processed.	Integer value.
Service Name **(R)**	The name of the business service on which to invoke the method.	

See Also

"Application" on page 45
"Business Service" on page 103

Application Find

Siebel Object Types > Application > Application Find

Defines which find object definitions are used by the application.

Properties

Property	Description / Usage	Valid Values / Examples
Default Find **(O)**	Valid values are TRUE or FALSE. Setting this value to TRUE allows you to set the chosen Find object as the default Find object for your application. The default Find object will appear in the Look In drop-down list in the Search Center.	
Find **(R)**	The name of the find object definition to include in the application.	Examples: Contact, Account.

Property	Description/Usage	Valid Values/ Examples
Name **(R)**	Same as the Find property.	
Sequence **(O)**	The order relative to other finds in which this find object definition is listed in the Find dialog box.	Integer value.

See Also

"Application" on page 45
"Find" on page 187

Application Locale

Siebel Object Types > Application > Application Locale

Represents language-specific overrides used with the Application object type.

Properties

Property	Description/Usage	Valid Values/ Examples
Display Name **(O)**	Name displayed on menus and page tabs for an application for a specific language.	
Name **(R)**	The abbreviation of the language being used by the application.	Example: ENU.

See Also

"Application" on page 45

Application Method Menu Item (H)

Siebel Object Types > Application > Application Method Menu Item (H)

Defines the menu for the application that is available when the application is active. You edit objects of this type using the Menu Designer.

Properties

Property	Description/Usage	Valid Values/Examples
Menu Text **(R)**	The text displayed in the menu item.	
Method **(R)**	The method invoked by the item.	

Property	Description/Usage	Valid Values/Examples
Name **(R)**	The name of the application method menu item.	
Position **(R)**	The position in the menu tree.	Example: 3.1 is the first item in the submenu of the third item of the top-level menu.

See Also

"Application" on page 45
"Application Find" on page 49
"Application Server Script" on page 51
"Hidden Object Types and Properties" on page 21

Application Method Menu Item Locale (H)

Siebel Object Types > Application > Application Method Menu Item (H) > Application Method Menu Item Locale (H)

Represents language-specific overrides used with the Application Method Menu Item object type.

Properties

Property	Description/Usage	Valid Values/ Examples
Menu Text **(R)**	The text displayed in the menu item for a specific language.	
Name **(O)**	The abbreviation of the language being used by the application.	Example: ENU.

See Also

"Application" on page 45
"Application Method Menu Item (H)" on page 50
"Hidden Object Types and Properties" on page 21

Application Server Script

Siebel Object Types > Application > Application Server Script

Defines the scripts (Siebel VB or Siebel eScript) implemented for this application object. For more information, read *Siebel Object Interfaces Reference*.

Application Server Scripts are edited by right-clicking on the Application and selecting Edit Server Scripts.

Properties

Property	Description / Usage	Valid Values / Examples
Name **(R)**	The name of the script.	
Program Language **(S)**	A value indicating the programming language of the script.	SBL — Siebel VB. JS — Siebel eScript.
Script **(R)**	The body of the script.	Upper limit is 16K.
Sequence **(R)**	The order of the script in the compilation relative to other scripts for the same object.	Integer value.

See Also

"Application" on page 45
"Application Method Menu Item (H)" on page 50

Application Toolbar (H)

Siebel Object Types > Application > Application Toolbar (H)

Associates a toolbar with an application. Different applications can use different toolbars. When running a specific application, the main window shows only the associated toolbars.

Properties

Property	Description / Usage	Valid Values / Examples
Name **(R)**	Name of the application toolbar object definition.	Must be unique within this application.
Sequence **(O)**	The order of this toolbar relative to other toolbars appearing at the top of the main window.	Integer value.
Toolbar **(R)**	The name of the toolbar object definition to be used in the application.	

See Also
"Application" on page 45
"Hidden Object Types and Properties" on page 21
"Toolbar" on page 314
"Toolbar Item" on page 315

Application User Prop (H)

Siebel Object Types > Application > Application User Prop (H)

The Application User Prop is for Oracle use only.

Assignment Attribute (H)

Siebel Object Types > Assignment Attribute

Defines an attribute that can be referenced in assignment criteria records. An Assignment Attribute object definition specifies a logical attribute that can be chosen from a picklist for defining comparisons; it does not directly specify a particular database column or combination of columns. Column mapping is accomplished through the child assignment attribute column object definitions, one for each assignment object that uses the parent assignment attribute.

An assignment attribute also specifies the picklist to be used in the Values applet in the Assignment Criteria view when the administrator is populating an attribute in a value record.

Properties

Property	Description/Usage	Valid Values/Examples
Bounded **(O)**	A TRUE or FALSE value.	TRUE — Picklist is bounded; that is, the administrator can select a value only from the picklist. FALSE — The administrator can enter a value that does not appear in the picklist.
Data Type **(R)**	The data type for the assignment attribute.	Valid values are Varchar and Number.
Name **(R)**	The name of the assignment attribute.	Must be unique within the repository.
Order By LOV Type **(O)**	Not used.	Not applicable.
Pick Field **(O)**	The name of the field to select from the picklist.	Valid values are Created, Created By, Id, Updated, and Updated By.

Property	Description/Usage	Valid Values/Examples
Pick List **(O)**	The name of the picklist that supplies the selection values for populating any list column in the Values applet (in the Assignment Criteria view) that is based on this assignment attribute.	
Translate Pick Field **(O)**	Determines if the Language Independent Code is stored in the database or the display value for Assignment Criteria.	Valid values are Created, Created By, Id, Updated, and Updated By.
Use Range **(O)**	A TRUE or FALSE value.	TRUE — A pair of list columns, indicating a range, is to appear for assignment criteria attributes based on this assignment attribute. FALSE — A single list column is to appear for assignment criteria attributes based on this assignment attribute. Example: The Revenue assignment attribute, which has a TRUE value for Use Range, displays two list columns, Revenue Low and Revenue High. The administrator would enter 50000 in the former and 100000 in the latter in a value record to indicate the range $50,000 through $100,000.

Check In Before Use

The projects used for creating new assignment attributes must be checked back into the server before use. It is not sufficient to compile the object definitions and distribute the resulting SRF file.

See Also

"Assignment Attribute Column (H)" on page 55

Assignment Attribute Column (H)

Siebel Object Types > Assignment Attribute (H) > Assignment Attribute Column

Assigns an assignment attribute to an assignment object and a workflow policy component column. These mappings set up value matching within the assignment object and workflow policy object for criteria that use the parent assignment attribute.

The parent assignment attribute is an abstract logical attribute to test for a value match and is specified in an assignment criterion by the administrator. Its child assignment attribute columns specify the actual mappings to assignment objects and workflow policy component columns. Each assignment attribute column can specify a different assignment object to search or a different attribute within the same assignment object.

Properties

Property	Description / Usage	Valid Values / Examples
Assignment Object **(R)**	An assignment object that is to be checked for a value match with the parent assignment attribute.	
Name **(R)**	The name of the assignment attribute column. The typical format for the name is: *assignment_object: workflow_component_column.*	Example: Service Request: Account Area Code.
Sequence **(R)**	A unique sequence number for the assignment attribute column within the assignment attribute. If an assignment attribute has multiple assignment attribute columns, Assignment Manager searches for attribute values in the order of the values in this property.	Integer value.
Workflow Component **(R)**	The name of the workflow policy component, within the specified workflow policy object, with which this assignment attribute column is associated.	Logical mapping of a database table; for example, Account.

Property	Description/Usage	Valid Values/ Examples
Workflow Component Column **(R)**	The alias of the workflow policy component column, within the specified workflow policy component, with which this assignment attribute column is associated.	Column names. For example, account_id.
Workflow Object **(R)**	The name of the workflow policy object with which the assignment attribute column is associated.	Defaults when an assignment object is picked.

Check In Before Use

The projects used for creating new assignment attribute columns must be checked back into the server before using the assignment attribute column. It is also advisable to compile the object definitions and distribute the resulting SRF file.

See Also

"Assignment Attribute (H)" on page 53
"Assignment Object (H)" on page 60
"Workflow Policy Component" on page 355
"Workflow Policy Component Col" on page 357

Assignment Criteria (H)

Siebel Object Types > Assignment Criteria

Defines an attribute that can be used in assignment criteria records. Assignment criteria appear in the picklist that is displayed when you click in the Criteria list column when editing an assignment criterion record in Siebel applications.

The Assignment Criteria object definitions have four properties. Two are for specifying the name (internally and as displayed), and two are used to configure skill tables. Most of the behavior of assignment criteria is configured in the assignment criteria attribute children.

Properties

Property	Description / Usage	Valid Values / Examples
Display Flag **(O)**	A TRUE or FALSE value.	TRUE — The Criteria object is available on the Assignment Criteria view in Siebel applications. FALSE — The Criteria object is not available on the Assignment Criteria view in Siebel applications.
Display Name **(O)**	The name that appears for this assignment criteria object in the picklist and the Criteria list column in the Assignment Criteria applet in the Assignment Criteria view.	If not specified, the value in the Name property is used.
Employee Skill **(O)**	A TRUE or FALSE value.	TRUE — The Assignment Criteria object is available in the Skill applets in Siebel applications. FALSE — The Assignment Criteria object is not available in the Skill applets in Siebel applications.
Name **(R)**	The name of the assignment criteria.	
Team	A TRUE or FALSE value. Used in conjunction with dynamic candidates feature.	TRUE — The assignment criterion is based on a column on the team table. FALSE — The assignment criterion is not based on a column on the team table.
Use Expertise **(O)**	A TRUE or FALSE property that applies to assignment criteria that are skills. It specifies whether expertise codes are to be stored for the skill. If so, Assignment Manager uses the expertise code to match an assignment object to people.	TRUE — Expertise codes are to be stored for the skill. FALSE — Expertise codes are not to be stored for the skill.

Check In Before Use

The projects used for creating new assignment criteria must be checked back into the server before using the assignment criteria. It is not sufficient to compile the object definitions and distribute the resulting SRF file.

See Also

"Assignment Criteria Attribute (H)" on page 58

Assignment Criteria Attribute (H)

Assignment criteria attributes make it possible for an Assignment Criteria object to consist of multiple attributes.

Properties

Property	Description/Usage	Valid Values/Examples
Assignment Attribute **(R)**	The name of the assignment attribute that this assignment criteria attribute is based on.	Selected from a picklist.
Display Name **(R)**	The list column label that appears for this assignment criteria attribute in the Values list applet.	If omitted, Name is used in the list column label.
Display Sequence **(R)**	The order in which the list column for this assignment criteria attribute appears in the list applet relative to other assignment criteria attributes in the assignment item. A lower number places the list column farther to the left.	
Display Sequence - Language Override	The list columns in the list applet can be ordered differently in different languages.	
Name **(R)**	Name of the assignment criteria attribute, for identification.	Must be unique within the parent assignment criteria.
Pick Applet **(O)**	The name of a pick applet to display the picklist for selection of a value in the assignment attribute's list column, if a picklist is defined.	
Store Column **(R)**	The column in the assignment factor items table (S_ASGN_RULE_ITEM) in which to store the value for the assignment criteria attribute. There are four columns of each data type (Number and Varchar) provided for storage of assignment criteria attribute values in each assignment criteria. The data type may be determined from the Data Type property in the referenced assignment attribute object definition.	A unique value, between 1 and 4, for each assignment criteria attribute of the same data type.

Check In Before Use

The projects used for creating new assignment criteria attributes must be checked back into the server before using the assignment criteria attributes. It is not sufficient to compile the object definitions and distribute the resulting SRF file.

See Also

"Assignment Attribute (H)" on page 53
"Assignment Criteria (H)" on page 56

Assignment Criteria Attribute Locale (H)

Siebel Object Types > Assignment Criteria (H) > Assignment Criteria Attribute (H) > Assignment Criteria Attribute Locale

Represents language-specific overrides used with the Assignment Criteria Attribute object type.

Properties

Property	Descriptions / Usage	Valid Values / Examples
Display Name **(R)**	The list column label that appears for this assignment criteria attribute in the Values list applet for a specific language.	
Display Sequence	The order in which the list column for this assignment criteria attribute appears in the list applet relative to those of other assignment criteria attributes in the assignment item. A lower number places the list column farther to the left.	
Name **(R)**	The abbreviation of the language being used by the application.	Example: ENU.

See Also

"Assignment Criteria Attribute (H)" on page 58

Assignment Criteria Locale (H)

Siebel Object Types > Assignment Criteria (H) > Assignment Criteria Locale

Represents language-specific overrides used with the Assignment Criteria object type.

Properties

Property	Description/Usage	Valid Values/ Examples
Display Name **(R)**	The name that appears for this assignment criteria in the picklist and the Criteria list column in the Assignment Criteria applet in the Assignment Criteria view for a specific language.	
Name **(R)**	The abbreviation of the language being used by the application.	Example: ENU.

See Also

"Assignment Criteria (H)" on page 56

Assignment Object (H)

Siebel Object Types > Workflow Policy Object (H) > Assignment Object

Specifies a business entity to which a candidate can be assigned, what gets updated to accomplish that assignment, and other assignment behavior arguments. Assignment Object object definitions are assigned to assignment rules in the Object field of assignment rule records.

The parent workflow policy object provides a set of available column mappings through its child workflow policy components and grandchild workflow policy component columns. These can be specified in assignment attribute column object definitions as columns to test for value matches and to monitor in dynamic assignment for value changes.

NOTE: The Property column in the following table shows property names as they appear in the Properties window in Siebel Tools, and the List Applet column shows property names as they appear in the Assignment Objects list applet in Siebel Tools. In this table, the list applet name for a property is shown only if different from the property name.

Properties

Property	List Applet	Description/Usage	Valid Values/Examples
Add Team Members **(O)**	Replace Team Members	A TRUE or FALSE value.	TRUE — Assignment Manager deletes existing team members from the teams and existing assignment rules in the assignment rule intersection table that no longer qualify. FALSE — Assignment Manager never deletes existing team members from the assignment object, even if they no longer qualify.
Assignment Date Column **(O)**		The datetime stamp when a record was last assigned.	ASGN_DT

Property	List Applet	Description/Usage	Valid Values/Examples
Assignment Mode **(O)**		Determines if Assignment Manager performs any filtering based on organizations or people.	Valid values are: ■ Independent. Assigns people and organizations that qualify, regardless of whether they are related or not. ■ Person-Oriented. Assigns people that qualify, then assigns only qualified organizations that the assigned people belong to. ■ Organization-Oriented. Assigns organizations that qualify, then assigns qualified people from those organizations. ■ Organization and Person-Oriented. Identifies all qualified people and organizations, then assigns only those that have both a qualified person and related qualified organization.
Assignment Scoring Mode **(R)**		Defines which mode is being used to determine an assignment, person-based or organization-based.	Person-based (default) — assignment rule score is set to the score of the highest-scoring candidate (existing behavior). Organization-based — assignment rule score is set to the score of the highest-scoring organization for the rule. **NOTE:** If no value is specified for this property, it defaults to Person-based.

Property	List Applet	Description/Usage	Valid Values/Examples
Auto Reassign **(R)**		A TRUE or FALSE value that turns on or off dynamic assignment for rules using this assignment object. The Generate Triggers server component looks for this flag when it generates triggers for the active assignment policy. If the property is set to FALSE, no trigger is created for the update; the assignment occurs only once when the record is inserted.	TRUE — Assignment Manager automatically reassigns assignment objects when assignment attribute values are changed and when policies are defined. This is the default value. FALSE — Assignment Manager does not automatically reassign assignment objects. **NOTE:** If the value of this property is changed after running a Generate Triggers server component job, you must generate triggers again.
Calendar Activity Additional Fields **(O)**		A string that provides extra information to be used when creating an activity through availability-based assignment. This string follows a format of parent field, field 1, value1, field 2, value2. When this field is specified, Assignment Manager sets the additional fields to the specified field values. For more information about availability-based assignment, see *Siebel Assignment Manager Administration Guide*.	As an example, for the Service Request object, you can specify the following fields: Activity SR Id, Type, ToDo. When a new activity is created, these field values are used for the new activity.

Property	List Applet	Description/Usage	Valid Values/Examples
Calendar Create Activity **(O)**		A TRUE or FALSE value that determines if an activity is created in an employee's calendar if assignment by availability is used. **NOTE:** If using the Activity assignment object, activities may be created according to the following logic: If one employee passes from rules that have the Check Employee Calendar field turned on, and no employee passes from other rules, the same activity is reused. If the assignment object in question is not Activity, new activities are created for each employee who passes. The "Parent Activity" for all activities is set to the original activity Id.	TRUE — Assignment Manager can create activities to block out time in an employee's calendar. FALSE — Activity is not created in employee's calendar. **NOTE:** If no value is specified, it defaults to FALSE.
Calendar Duration Column **(O)**		The column in the assignment object primary table that references the duration of the object. This is the time for which an employee's schedule is blocked. Required if using availability-based assignment.	

Property	List Applet	Description/Usage	Valid Values/Examples
Calendar Early Start Time Column **(O)**		The column in the assignment object primary table that references the early start time of the object. This column stores the earliest date/time at which an activity can start. The time range from Calendar Early Start Time to Calendar Start Time represents the time range during which the activity can start. **NOTE:** If no value is specified, the activity can start at any time before the value in the Calendar Start Time Column. Required if using availability-based assignment.	
Calendar Start Time Column **(O)**		The column in the assignment object primary table that references the start time of the object. This column stores the latest time at which an activity can start. The time range from Calendar Early Start Time to Calendar Start Time represents the time range during which the activity can start. Required if using availability-based assignment.	

Property	List Applet	Description/Usage	Valid Values/Examples
Default Employee **(O)**		The login ID of the default employee. Assignment Manager assigns this employee when a conflict occurs (that is, no assignment rules match or or when two exclusive rules pass with the same score) and it cannot select the correct set of employees. If not specified, Assignment Manager sets the owner Id field in the object row to No Match Row Id.	
Default Group **(O)**		Not used.	Not applicable.
Default Org **(O)**	Default Organization	The name of the default organization. Assignment Manager assigns this organization when a conflict occurs (that is, no assignment rules match or when two exclusive rules pass with the same score) and it cannot select the correct set of organizations. If not specified, Assignment Manager sets the owner Id field in the object row to No Match Row Id.	

Property	List Applet	Description/Usage	Valid Values/Examples
Default Position **(O)**		The name of the default position. Assignment Manager assigns this position when a conflict occurs (that is, no assignment rules match or when two exclusive rules pass with the same score) and it cannot select the correct set of positions. If not specified, Assignment Manager sets the owner Id field in the object row to No Match Row Id.	
Employee Column **(O)**		The name of the column in the employee intersection table that points to rows in the employee table (S_EMPLOYEE). Required if Employee Table property is non-NULL.	
Employee Denorm Column **(O)**	Employee Team Denorm Column	Name of the column in the employee intersection table that stores whether the employee is assigned by Assignment Manager in denormalization program.	
Employee Key Column **(O)**		The name of the column that stores the key values of the employees assigned to the team at runtime. Required if the UseKeyValue server component parameter is set to True.	
Employee Manual Column **(O)**	Employee Team Manual Column	Name of the column in the employee intersection table that stores whether the employee is manually assigned.	

Property	List Applet	Description/Usage	Valid Values/Examples
Employee Primary Column **(O)**		Name of the column in the primary table that stores the primary employee (on a team) or a single-owner employee. **TIP:** If you specify a value in the Employee Primary Column and do not specify a value in the Employee Table, the assignment object is enabled for employees in the single-owner mode. Required if the rule is of the single-owner type (that is, Assignment Manager can assign only one candidate), or if Set Primary Employee is TRUE.	
Employee Primary Column List **(O)**	Employee Team Copy Columns	Comma-separated list of column names in the employee intersection table that should be copied when the primary employee changes. Assignment Manager copies values from these columns to form the old primary employee to the new primary employee.	
Employee Primary Denorm Column **(O)**		Name of the column in the primary table that stores whether the primary employee is assigned by Assignment Manager in denormalization mode. Required if the rule is of the single-owner type (that is, Assignment Manager can assign only one candidate) or if Set Primary Employee is TRUE.	

Property	List Applet	Description/Usage	Valid Values/Examples
Employee Primary Manual Column **(O)**		Name of the column in the primary table that stores whether the primary employee is manually assigned. Required if the rule is of the single-owner type (that is, Assignment Manager can assign only one candidate) or if Set Primary Employee is TRUE.	
Employee Primary System Column **(O)**		The name of the column in the primary table that stores whether the primary employee is assigned by Assignment Manager in denormalization mode. Required if the rule is of the single-owner type (that is, Assignment Manager can assign only one candidate) or if Set Primary Employee is TRUE.	
Employee System Column **(O)**	Employee Team System Column	Name of the column in the employee intersection table that stores whether the employee is assigned by Assignment Manager in non-Denormalization mode. Required if Employee Table property is non-NULL.	

Property	List Applet	Description/Usage	Valid Values/Examples
Employee Table **(O)**		The name of the employee intersection table. If the assignment object can be assigned to multiple employees, this intersection table stores the set of employee IDs that are assigned to the assignment object IDs. Required if you want team-based assignment enabled. Otherwise, Assignment Manager assumes the assignment is not team-enabled.	
Employee Team Score Column **(O)**		The name of the column in the team table that stores scores for each member of an employee team.	Any column that stores a number.
Exclude Column **(O)**	Lock Assignment Column	The name of the lock assignment column in the primary table, which is a column in the assignment object's primary table that can be used to prevent Assignment Manager from processing certain rows. **NOTE:** Most of the assignment object base tables have an ASGN_USR_EXCLD_FLG column, and for some objects, this column is exposed in the Siebel application as a field named Lock Assignment. However, for assignment objects that do not have this column exposed, you can query the database for the ASGN_USR_EXCLD_FLG column in the appropriate table.	**NOTE:** Typically the value of this column is set through business component configuration; it is not set by Assignment Manager. For example, when if you create a new activity, the Lock Assignment checkbox for that activity is automatically checked. Before you can assign this activity, you must uncheck this field.

Property	List Applet	Description / Usage	Valid Values / Examples
Group Column **(R)**	Rule Column	The name of a row ID column in the rule intersection table that points to rows in the assignment rule table (S_ASGN_GRP). This column serves as a foreign key to the assignment rule table, uniquely identifying a rule for each association between an assignment entity and a rule. Required if Rule Table property is non-NULL.	
Group Denorm Column **(O)**	Rule Team Denorm Column	The name of the column in the rule intersection table that stores whether the rule is assigned by Assignment Manager in denormalization mode.	
Group Manual Column **(O)**	Rule Team Manual Column	The name of the column in the rule intersection table that stores whether the rule is manually assigned.	
Group Primary Column **(O)**	Rule Primary Column	The name of the column in the primary table that stores the primary or single-owner rule. Required if Set Primary Rule is TRUE.	
Group Primary Column List **(O)**	Rule Team Copy Columns	A comma-separated list of column names in the rule intersection table that should be copied when the primary rule changes. Assignment Manager copies values from these columns from the old primary rule to the new primary rule.	

Property	List Applet	Description/Usage	Valid Values/Examples
Group Primary Denorm Column **(O)**	Rule Primary Denorm Column	The name of the column in the primary table that stores whether the primary rule is assigned by Assignment Manager in denormalization mode. Required if the rule is of the single-owner type (that is, Assignment Manager can assign only one candidate) or if Set Primary Group is TRUE.	
Group Primary Manual Column **(O)**	Rule Primary Manual Column	The name of the column in the primary table that stores whether the primary rule is manually assigned. Required if the rule is of the single-owner type (that is, Assignment Manager can assign only one candidate) or if Set Primary Group is TRUE.	
Group Primary System Column **(O)**	Rule Primary System Column	The name of the column in the primary table that stores whether the primary rule is assigned by Assignment Manager in denormalization mode. Required if the rule is of the single-owner type (that is, Assignment Manager can assign only one candidate) or if Set Primary Group is TRUE.	
Group System Column **(O)**	Rule Team System Column	The name of the column in the rule intersection table that stores whether the rule is assigned by Assignment Manager in denormalization mode	
Group Table **(O)**	Rule Table	The name of the rule intersection table.	

Property	List Applet	Description/Usage	Valid Values/Examples
Ignore Assignment Attributes **(R)**	Ignore Extra Attributes	A runtime setting for Assignment Manager (TRUE or FALSE value).	TRUE — Assignment Manager ignores assignment criteria that are not relevant to the assignment object being assigned. Example: If you have an assignment rule that specifies Service Request Priority as a criterion, Assignment Manager ignores this criterion when assigning opportunities and accounts because Service Request Priority is an assignment attribute that is relevant only to service requests. FALSE — All assignment criteria are evaluated.
Keep Creator **(R)**		A TRUE or FALSE value.	TRUE — Assignment Manager does not delete the creator of the assignment object from the team. FALSE — Assignment Manager can delete the creator.
Keep Man Asgn Primary Employee **(O)**	Keep Manual Primary Employee	A TRUE or FALSE value.	TRUE — Causes Assignment Manager to not set the primary employee if a Siebel user uses the Siebel client to explicitly set the primary employee in the team. Instead, Assignment Manager checks the primary person manual flag in the assignment object's primary table. FALSE — Assignment Manager can set the primary employee.

Property	List Applet	Description/Usage	Valid Values/Examples
Keep Man Asgn Primary Group **(O)**	Keep Manual Primary Rule	(Applies to teams only) A TRUE or FALSE value.	TRUE — Assignment Manager does not set the primary assignment rule if a Siebel user uses the Siebel client to explicitly set the primary assignment rule in the team. Assignment Manager checks the primary assignment rule manual flag in the assignment object's primary table. FALSE — Assignment Manager sets the primary assignment rule even if it is already manually assigned.
Keep Man Asgn Primary Org **(O)**		A TRUE or FALSE value.	TRUE — If the assignment object has the Org Primary Manual Column defined, Assignment Manager checks the value of that column. If the column value is Y, Assignment Manager does not reassign the primary organization. FALSE — Assignment Manager does not check the value.
Keep Man Asgn Primary Position **(O)**	Keep Manual Primary Position	A TRUE or FALSE value.	TRUE — Causes Assignment Manager to not set the primary position if a Siebel user uses the Siebel client to explicitly set the primary position in the team. Assignment Manager checks the primary person manual flag in the assignment object's primary table. FALSE — Assignment Manager can set the primary position.

Property	List Applet	Description/Usage	Valid Values/Examples
Keep User Assigned **(R)**	Keep Manual Assigned	A TRUE or FALSE value that applies to teams only. **NOTE:** Assignment Manager checks the value of the corresponding Team Manual Column property to determine if this assignee was manually assigned or not.	TRUE — Assignment Manager keeps manually assigned people (such as positions added to an account sales team) and assignment rules (such as territories added to a territory list for an opportunity). FALSE — Manually assigned employees, positions, and rules can be replaced by an automatic assignment operation.
Max Number Per Role **(O)**		Not used.	Not applicable.
Name **(R)**		The name of the assignment object. Uniquely identifies it within the repository and provides the name that appears in the selection picklist and multi-value group applet in assignment rules in Assignment Manager administration views.	
Org Column **(R)**	Organization Column	The name of the column in the organization intersection table that points to rows in the organization table. Required if Org Table property is non-NULL.	
Org Denorm Column **(O)**		The name of the column in the organization intersection table that specifies whether the organization is assigned by Assignment Manger in denormalization mode.	

Property	List Applet	Description / Usage	Valid Values / Examples
Org Key Column		The name of the column that stores the key values of the organizations assigned to the team at runtime. Required if the UseKeyValue server component parameter is set to True.	
Org Manual Column **(O)**		The name of the column in the organization intersection table that specifies whether the organization is manually assigned.	
Org Primary Column **(O)**		The name of the column in the primary table that stores the primary organization (on a team) or single-owner organization. **TIP:** If you specify a value in the Org Primary Column and do not specify a value in the Org Table, the assignment object is enabled for organizations in the single-owner mode. Required if the rule is of the single-owner type (that is, Assignment Manager can assign only one candidate) or if Set Primary Org is TRUE.	
Org Primary Column List **(O)**		A comma-separated list of column names in the organization intersection table that should be copied when the primary organization changes. Assignment Manager copies values from these columns from the old primary organization to the new primary organization.	

Property	List Applet	Description/Usage	Valid Values/Examples
Org Primary Denorm Column **(O)**		The name of the column in the primary table that specifies whether the primary organization is assigned by Assignment Manger in denormalization mode. Required if the rule is of the single-owner type (that is, Assignment Manager can assign only one candidate) or if Set Primary Org is TRUE.	
Org Primary Manual Column **(O)**		The name of the column in the primary table that specifies whether the primary organization is manually assigned. Required if the rule is of the single-owner type (that is, Assignment Manager can assign only one candidate) or if Set Primary Org is TRUE.	
Org Primary System Column **(O)**		The name of the column in the primary table that specifies whether the primary organization is assigned by Assignment Manager in denormalization mode. Required if the rule is of the single-owner type (that is, Assignment Manager can assign only one candidate) or if Set Primary Org is TRUE.	
Org System Column **(O)**		The name of the column in the organization intersection table that specifies whether the organization is manually assigned.	

Property	List Applet	Description/Usage	Valid Values/Examples
Org Table **(O)**	Organization Table	The name of the organization intersection table. If the assignment object can be assigned to multiple organizations, this intersection table stores the set of organization IDs that are assigned to assignment object IDs. Required if you want team-based assignment enabled. Otherwise, Assignment Manager assumes the assignment is not team-enabled.	
Org Team Score Column **(O)**		The name of the column in the team table that stores scores for each member of an organization team.	Any column that stores a number.
Position Column **(R)**		The name of the column in the position intersection table that points to rows in the position table (S_POSTN). Required if Position Table property is non-NULL.	
Position Denorm Column **(O)**	Position Team Denorm Column	The name of the column in the position intersection table that stores whether the position is assigned by the denorm program.	
Position Key Column		The name of the column that stores the key values of the employees assigned to the team at runtime. Required if the UseKeyValue server component parameter is set to True.	

Property	List Applet	Description / Usage	Valid Values / Examples
Position Manual Column **(O)**	Position Team Manual Column	The name of the column in the position intersection table that stores whether the position is manually assigned.	
Position Primary Column **(O)**		The name of the column in the primary table that stores the primary position (for teams) or single-owner position. Required if single-position, or if Set Primary Position is TRUE.	
Position Primary Column List **(O)**	Position Team Copy Columns	A comma-separated list of column names in the position intersection table that should be copied when the primary position changes. Assignment Manager copies values from these columns from the old primary position to the new primary position.	
Position Primary Denorm Column **(O)**		The name of the column in the primary table that stores whether the primary position is assigned by the denorm program.	
Position Primary Manual Column **(O)**		The name of the column in the primary table that stores whether the primary position is manually assigned.	
Position Primary System Column **(O)**		The name of the column in the primary table that stores whether the primary position is assigned by the Siebel application.	

Property	List Applet	Description/Usage	Valid Values/Examples
Position System Column **(O)**	Position Team System Column	The name of the column in the position intersection table that stores whether the position is assigned by the Siebel application.	
Position Table **(O)**		The name of the position intersection table. If the assignment object can be assigned to multiple positions, this intersection table stores the set of position IDs that are assigned to the assignment object IDs. Required if you want team-based assignment enabled. Otherwise, Assignment Manager assumes the assignment is not team-enabled.	
Position Team Score Column **(O)**		The name of the column in the team table that stores scores for each member of a position team.	Any column that stores a number.
Primary Table **(R)**		The name of the primary table. This is the base table of the business component that the assignment object is used to assign.	Example: S_ORG_EXT is the primary table for the Account assignment object.
Set Primary Employee **(O)**		A TRUE or FALSE value. The primary employee is generally the highest-scoring employee in the highest-scoring assignment rule for the assignment object being assigned.	TRUE — Assignment Manager selects and sets the primary employee. FALSE — Assignment Manager does not select and set the primary employee.
Set Primary Group **(O)**	Set Primary Rule	A TRUE or FALSE value. The primary assignment rule is generally the highest-scoring assignment rule for the assignment object being assigned.	TRUE — Assignment Manager selects and sets the primary position. FALSE — Assignment Manager does not select or set the primary position.

Property	List Applet	Description/Usage	Valid Values/Examples
Set Primary Org **(O)**		A TRUE or FALSE value. The primary organization is generally the highest-scoring organization in the highest-scoring assignment rule for the object being assigned.	TRUE — Assignment Manager selects and sets the primary organization. FALSE — Assignment Manager does not select and set the primary organization.
Set Primary Position **(O)**		A TRUE or FALSE value. The primary position is generally the highest-scoring position in the highest-scoring assignment rule for the object being assigned.	TRUE — Assignment Manager selects and sets the primary position. FALSE — Assignment Manager does not select and set the primary position.
Skill Item Table **(O)**		The name of the skill item table that stores skills for the assignment object. This table must be a child table of the table specified by the Skill Table property where the Skill Id column is a foreign key column to the skill table.	
Skill Table **(O)**		The name of the table that stores skills for an assignment object.	
Team		A TRUE or FALSE value. Used in conjunction with dynamic candidates feature.	TRUE — The assignment criterion is based on a column on the team table. FALSE — The assignment criterion is not based on a column on the team table.

Check In Before Use

The projects used for creating new assignment objects must be checked back into the server before the assignment objects are available for use. You typically also compile the object definitions and distribute the resulting SRF file, but this is not necessary to make the assignment objects available.

See Also

"Workflow Policy Object (H)" on page 358

Assignment Object Extension (H)

Siebel Object Types > Workflow Policy Object (H) > Assignment Object Extension (H)

Stores the results tables when running in Reporting Mode. The values in the Assignment Object Extension object properties determine what columns are used to determine:

- What candidates were added or dropped
- What tables to use for reporting
- What column to use as the key column
- What columns to use as the position, employee, or organization column

Properties

Property	Description/Usage	Valid Values/ Examples
Employee Reporting Column **(O)**	The name of the column in the employee reporting intersection table that points to rows in the employee table. Required if Employee Table property is non-NULL.	
Employee Reporting Deletion Column	Marks the employees that are removed from the team. Applicable in both snapshot and delta reporting modes.	
Employee Reporting Insertion Column	Marks the employees that are added to the team. Applicable in both snapshot and delta reporting modes.	
Employee Reporting Key Column **(O)**	Stores the key values of the employees assigned to the team at runtime. Required if the UseKeyValue server component parameter is set to True.	
Employee Reporting Table **(O)**	The name of the employee reporting intersection table. If the assignment object can be assigned to multiple employees, this intersection table stores the set of employee IDs that are assigned to the assignment object IDs.	
Name **(R)**	The name of the assignment object extension. This name should always match the name of the assignment object.	
Org Reporting Column **(O)**	The name of the column in the organization reporting intersection table that points to rows in the organization table. Required if Organization Table property is non-NULL.	
Org Reporting Deletion Column	Marks the organizations that are removed from the team. Applicable in both snapshot and delta reporting modes.	

Property	Description/Usage	Valid Values/ Examples
Org Reporting Insertion Column	Marks the organizations that are added to the team. Applicable in both snapshot and delta reporting modes.	
Org Reporting Key Column **(O)**	Stores the key values of the organizations assigned to the team at runtime. **NOTE:** The UseKeyValue server component parameter must also be set to True.	
Org Reporting Table **(O)**	The name of the organization reporting intersection table. If the assignment object can be assigned to multiple organizations, this intersection table stores the set of organization IDs that are assigned to the assignment object IDs.	
Position Reporting Column **(O)**	The name of the column in the position reporting intersection table that points to rows in the position table. Required if Position Table property is non-NULL.	
Position Reporting Deletion Column	Marks the positions that are removed from the team. Applicable in both snapshot and delta reporting modes.	
Position Reporting Insertion Column	Marks the positions that are added to the team. Applicable in both snapshot and delta reporting modes.	
Position Reporting Key Column **(O)**	Stores the key values of the positions assigned to the team at runtime. **NOTE:** The UseKeyValue server component parameter must also be set to True.	
Position Reporting Table **(O)**	The name of the position reporting intersection table. If the assignment object can be assigned to multiple positions, this intersection table stores the set of position IDs that are assigned to the assignment object IDs.	

See Also

"Assignment Object (H)" on page 60
"Workflow Policy Object (H)" on page 358

Assignment User Prop (H)

Siebel Object Types > Workflow Policy Object (H) > Assignment Object (H) > Assignment User Prop (H)

Communicates a property value to C++ code that implements specialized behavior. The values of these properties can be changed at configuration time. These values persist in the repository and the Siebel repository file.

There are two types of assignment user properties:

- PositionTeamDenormN where N is any number

  Use these properties to copy additional columns when replacing a position by another position from the same rule. For example, you use these properties when you need to replace an existing sales representative on an account team with another representative. For more information, see the topic on configuring assignment objects to copy additional columns to the team table in *Siebel Assignment Manager Administration Guide*.

- User properties for availability-based assignment

  For more information about availability-based assignment, see *Siebel Assignment Manager Administration Guide*.

NOTE: A Column user property takes precedence over a corresponding Value user property. For example, if you define a Breakable Flag Column user property and a Breakable Flag Value user property, the former user property takes precedence. The order of precedence for Assignment Manager is:

Column user property > Value user property > Calendar Activity Additional Fields property

Properties

Property	Description/Usage	Valid Values/ Examples
Name	The name of the user property.	
Value **(R)**	The value of the user property.	

Changing Undocumented User Properties

Customer developers can change only those user properties that have been documented. Only internal Oracle developers who create or change specialized C++ code should create or delete user properties that are not documented.

See Also

"Assignment Object (H)" on page 60
"Workflow Policy Object (H)" on page 358

For information about user properties, read *Siebel Developer's Reference.*

Attribute

Siebel Object Types > Type > Attribute

An Attribute column holds data and does not point to data in other tables. This object type is used by the Siebel Enterprise Integration Manager (EIM). For more information, read *Siebel Developer's Reference*.

CAUTION: Do not modify this object type. Any modification can adversely affect performance and operation.

Properties

Property	Description/Usage
Category **(S)**	Groups object properties when they are shown on the property sheet. This grouping makes it easier when the user views property sheets.
Column Name **(S)**	Name of the attribute column.
Language Override Attribute **(S)**	Provides the ability to change the order of attribute columns when a particular language/locale is active.
Name **(S)**	Name of attribute column on the user interface.
No Compile **(S)**	A TRUE or FALSE value. If TRUE, the object definition is not compiled when Compile is selected from the Tools menu in Siebel Tools.
Translatable **(S)**	A TRUE or FALSE value. If TRUE, attribute can be translated. FALSE is the default.

Attribute Mapping

Siebel Object Types > EIM Interface Table > EIM Table Mapping > Attribute Mapping

Identifies an attribute (data) column to update in the destination (base) table specified in the parent Siebel Enterprise Integration Manager (EIM) table mapping.

CAUTION: Do not modify this object type. Any modification can adversely affect performance and operation.

Properties

Property	Description/Usage
Base Table Attribute Column Name **(S)**	Name of attribute column on target table.
Interface Table Data Column **(S)**	Name of corresponding EIM Interface Table Column generated.
Name **(S)**	Name of attribute column on target table.

See Also
"EIM Table Mapping" on page 172

Bitmap

Siebel Object Types > Bitmap Category > Bitmap

Bitmap objects are implemented in Siebel 7.x as references to GIF and JPG image files used by button controls and other objects in the repository.

Properties

Property	Description/Usage	Valid Values/Examples
Alt Text **(O)**	Alternative text used in place of name property for a bitmap.	
Data **(S)(H)**	The actual bitmap data.	
File Name **(O)**	The file name of the bitmap. The default directory for this file is \public*language*\images (example for *language* is enu for U.S. English).	
Height **(S)**	The height (in pixels) of the bitmap.	
MIME Type **(O)**	MIME type of the file's content.	
Name **(R)**	The name of the bitmap object definition.	
Transparent Color **(O)(H)**	A setting that provides 1-bit transparency support for logos, toolbars, buttons, and similar bitmap object definitions.	The color you select or enter in the Transparent Color property as an RGB triple indicates the color for which Clear is substituted. You can enter it as a triplet of integers, or when you click the Transparent Color button during a bitmap import, a color selection dialog box opens. This is useful especially for toolbar buttons and applet command buttons, where the user may have a custom background color setting. When the button appears on the user's background, you may want it to take on the background color from the application.
Width **(S)**	The width (in pixels) of the bitmap.	

See Also
"Applet" on page 29
"Bitmap Category" on page 87
"Command" on page 135
"Hidden Object Types and Properties" on page 21
"Toolbar" on page 314

Bitmap Category

Siebel Object Types > Bitmap Category

Provides the means to group bitmaps by function or some other relationship. The Bitmap object type is a child of Bitmap Category; therefore, every bitmap image in the repository must be found in exactly one bitmap category. Certain bitmap categories are predefined and must be used for their established purpose. For example, an applet's Background Bitmap property always checks the Applet Backgrounds bitmap category for the specified bitmap file.

Properties

Property	Description/Usage	Valid Values/Examples
Name **(R)**	The descriptive name of the bitmap category.	
Predefined **(O)(H)**	A TRUE or FALSE value indicating one of two kinds of bitmap category.	TRUE — Predefined category. These typically contain both user-provided and Siebel application-provided bitmap files. Bitmap files can be added or removed from predefined categories, but the bitmap category object definition itself cannot be deleted or renamed. FALSE — User-defined category. These typically have a specially named (by the Siebel application) bitmap in them, but the bitmap category object definition can be created, deleted, or modified by the user. These bitmaps are typically view bar logos, which have the fixed name Logo.bmp. The name of the bitmap category is user-provided and is referenced elsewhere (in the Bitmap Category property of the view, in this case).

See Also
"Applet" on page 29
"Bitmap" on page 86
"Command" on page 135
"Hidden Object Types and Properties" on page 21
"Screen" on page 256
"Toolbar" on page 314
"View" on page 325

Bitmap Locale

Siebel Object Types > Bitmap > Bitmap Locale

Represents language-specific overrides used with the Bitmap object type.

Properties

Property	Description/Usage	Valid Values/ Examples
Alt Text **(O)**	Alternative text used in place of the parent name property for a bitmap for a specific language.	
Name **(R)**	The abbreviation of the language being used by the application.	Example: ENU.

See Also
"Bitmap" on page 86

BusComp Browser Script

Siebel Object Types > Business Component > BusComp Browser Script

The browser script used by a Business Component object type. BusComp Browser scripts can be used for data validation. BusComp browser scripts are edited by right-clicking on the Business Component and selecting Edit Browser Scripts.

Properties

Property	Description / Usage	Valid Values / Examples
Browser Group **(O)**	Not currently used.	
Name **(R)**	Unique name for the script record. Do not modify this property. This is generated by Siebel Tools by combining the Browser Group and Procedure properties.	
Procedure **(O)**	Name of function or procedure.	
Script **(R)**	This is the text for the function.	

See Also

"Business Component" on page 91

BusComp Server Script

Siebel Object Types > Business Component > BusComp Server Script

Defines the scripts (Siebel VB or Siebel eScript) implemented for this business component object. For more information, read *Siebel Object Interfaces Reference*.

BusComp server scripts are edited by right-clicking on the Business Component and selecting Edit Server Scripts.

In version 7.0, the BusComp Server Script object type replaces the Business Component Script.

Properties

Property	Description / Usage	Valid Values / Examples
Name **(R)**	Unique name for the script record. Do not modify this property. This is generated by Siebel Tools by combining the Browser Group and Procedure properties.	
Program Language **(S)**	A value indicating the programming language of the script.	SBL — Siebel VB. JS — Siebel eScript.
Script **(O)**	Text for the function.	
Sequence **(O)**	Specifies the order of the script in compilation relative to other scripts for the same object.	Integer value.

See Also
"Business Component" on page 91

BusComp View Mode

Siebel Object Types > Business Component > BusComp View Mode

Defines allowable access control mechanisms that can be applied to the business component in any view.

Properties

Property	Description/Usage	Valid Values/Examples
Name **(R)**	Name of the view mode.	Valid values are Catalog, Group, Organization, Personal, and Sales Rep.
Owner Type **(R)**	Used to identify the owner entity of the records in the current view mode. For example, in the My Opportunities case, Position is an owner of an Opportunity record.	Valid values are Catalog, Category, Group, Organization, Person, and Position.
Private Field **(O)**	Primarily used for business components that enable predefined queries, such as Bug Query List, and Query List.	For more information, read "Private Field" on page 91.
Visibility Field **(O)**	Identifies the field used to establish single-owner based visibility. The designated field is a record ID pointer field to records in the owner business component.	

Property	Description / Usage	Valid Values / Examples
Visibility MVField **(O)**	Specifies the name of the multi-value field used for maintenance of the list of owners in the team visibility case. This property is used in conjunction with Visibility MVLink, which identifies the corresponding multi-value link. Typically the multi-value field points to the Name field in the Owner business component.	Valid values are Sales Rep, Organization, and Organization Id.
Visibility MVLink **(O)**	A property that specifies the name of the multi-value link used for maintenance of the list of owner records in team visibility case. This property is used in conjunction with Visibility MVField, which identifies the corresponding multi-value field.	Valid values are Catalog Category and Organization.

Private Field

This property enables Siebel applications to include predefined queries that are partitioned by the user but still allow public queries, defined by an administrator, that show up in the same list. The value in this property specifies the name of a field that indicates private or public access to the record, generally named Private. If a field is specified as the Private Field, that field contains a TRUE or FALSE value in each record, indicating whether or not the record is private. If the record is marked as private and current visibility is in effect, then only the owner of the record can display it. If it is not marked as private and current visibility is in effect, then all users are able to display it.

See Also

"Business Component" on page 91

Business Component

Siebel Object Types > Business Component

Defines a logical entity that associates columns from one or more tables into a single structure. Business components provide a layer of wrapping over tables, and the applets 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 system administrator role.

A business component not only encapsulates data, but is used by the developer to define the behavior of the entity.

A virtual business component represents external data as a business component. For a list of supported and unsupported attributes on field objects in virtual business components, read "Virtual Business Components" on page 183.

You can add user properties to business components.

Properties

Property	Description/Usage	Valid Values/Examples
Browser Class **(O)(H)**	Not used.	Not applicable.
Cache Data **(O)**	A TRUE or FALSE value. For more information, read "Cache Data" on page 100.	TRUE — Data is read once for each query; that is, if the same query on a given business component is executed twice, the data is read only once. If the business component is queried again with different search specifications, the data is read again. FALSE — Business component data is read each time the data is requested.
Class **(R)**	The name of the class that dictates the behavior of the business component.	Example: CSSBcVAbsResult.
Data Source **(O)**	Allows a business component to use a different configuration file data source name (such as Server or Local) than the one the application is currently using. This is necessary for business components whose data comes from a database other than the normal SQL database. If a business component specifies a data source, then all the tables it joins to (implicitly and explicitly) must be in the same data source. This is appropriate where there is a pocket of self-contained functionality, such as server administration.	If this property is empty, then the data source that was used to start the product is used.

Property	Description / Usage	Valid Values / Examples
Dirty Reads **(O)**	A TRUE or FALSE value. This property is ignored except when running against a Sybase database. When TRUE, users read data that can be changed without their knowledge. This feature is desirable with applications because the data is available for reading and anybody can update it. If dirty reads were not allowed, then nobody could update the data until everybody was finished displaying the data. However, if dirty reads are allowed, the data can be updated immediately as long as the same record has not been modified by another user since it was last queried. For example, if user 1 updates a record and user 2 is not aware of the change and tries to update the same record, user 2 gets a message stating that the data has been updated and he or she needs to query again.	TRUE (default) — The business component supports database dirty reads. FALSE — The business component does not support database dirty reads.
Distinct **(O)**	A TRUE or FALSE value. Setting this property to TRUE suppresses system fields (such as Created, Updated, Id, and so). Therefore, you may want to clone this business component if you plan to use it in other areas within the application.	TRUE — System fields are suppressed. FALSE — System fields are not suppressed.
Enclosure Id Field **(O)**	The ID of a recipient of literature, usually the contact ID.	

Property	Description/Usage	Valid Values/Examples
Extension Type **(O)**	Used in conjunction with the extension tables. If no value is specified, this property defaults to LINK if running Sybase and to JOIN otherwise.	LINK — A separate SQL statement is executed to retrieve the extended data; this property is used with a Sybase database server due to Sybase's 2 KB record length limit. JOIN — The extension table is joined and the extended data is retrieved with the rest of the data. If no value is specified, this property defaults to LINK if running Sybase and to JOIN otherwise.
Force Active **(O)**	A TRUE or FALSE value.	TRUE — All fields are automatically retrieved from the database server. This substantially degrades the performance of the business component and is not recommended. FALSE — (Default) Only those fields registered by the user interface or by a specialized business component are retrieved. This improves performance by retrieving only those data items required by the user interface or specialized
GenReassignAct **(O)**	A TRUE or FALSE value.	TRUE — If the business component uses an MVG for visibility (sales team) and this value is TRUE, then an activity is generated when the primary team member changes. FALSE — No activity is generated.

Property	Description / Usage	Valid Values / Examples
Hierarchy Parent Field **(O)**	Enables the business component to be a specialized hierarchy business component. The field specified there stores the parent information in the hierarchies. Shows nesting in Hierarchical list applets.	
Insert Update All Columns **(S)**	Used internally to make all SQL INSERT and UPDATE statements include all columns. During internal batch processing the program can reuse parsed statements much more effectively. Do not modify this value.	
Log Changes **(O)**	A TRUE or FALSE value.	TRUE — Logging is enabled. FALSE — Logging is disabled.
Maximum Cursor Size **(O)**	If this property if set, it overrides MaxCursorSize in the configuration file. The maximum number of records to retrieve from the database server (DB2 UDB only). If -1, then records are retrieved until the EOF is reached; if a number is greater than 0, then, at most, that number of records is retrieved and reading is discontinued. Also works for MVG business components. For more information, read "MVG Business Component" on page 101.	Valid values are -1 or a number greater than 0.
Multi Recipient Select **(S)**	A TRUE or FALSE value. Set by the Service Request business component.	TRUE — Multiple noncontact record selection for correspondence generation is allowed. FALSE — Multiple noncontact record selections for correspondence generation is not allowed.

Property	Description / Usage	Valid Values / Examples
Name **(R)**	A name for the business component.	The name must be unique among business components. It can contain spaces but it cannot contain special characters.
No Delete **(O)**	A TRUE or FALSE value that puts limitations on what operations can be performed on the business component. This property is ignored if the current view has the property Admin Mode Flag set to TRUE.	TRUE — New records cannot be deleted through the business component. FALSE — New records can be deleted through the business component.
No Insert **(O)**	A TRUE or FALSE value that puts limitations on what operations can be performed on the business component. This property is ignored if the current view has the property Admin Mode Flag set to TRUE.	TRUE — New records cannot be added through the business component. FALSE — New records can be added through the business component.
No Merge **(O)**	A TRUE or FALSE value that puts limitations on what operations can be performed on the business component. This property is ignored if the current view has the property Admin Mode Flag set to TRUE. If this property is set to FALSE, then the No Delete property must also be set to FALSE, because merging requires that one of the records be deleted.	TRUE — The user cannot merge two existing records into a single record. FALSE — The user can merge two existing records into a single record.
No Update **(O)**	A TRUE or FALSE value that puts limitations on what operations can be performed on the business component. This property is ignored if the current view has the property Admin Mode Flag set to TRUE.	TRUE — Records cannot be updated through the business component. FALSE — Records can be updated through the business component.

Property	Description/Usage	Valid Values/Examples
Owner Delete **(O)**	A TRUE or FALSE value that establishes the deletion rights of the user to the business component. The owner of a record in a business component set up for team visibility mode (non-blank Visibility MVLink and Visibility Emp MVField properties) is the primary position on the sales team. The owner when set up for position-based personal visibility is the position in the field pointed to by the Position Visibility Field property. When a view is designated for Admin mode, the Owner Delete protection is disabled, and anyone having access to the view can delete any record.	TRUE — Only the owner (under personal or team visibility) can delete the record. FALSE — Anyone can delete the record.
Placeholder **(O)**	Not used.	Not applicable.
Popup Visibility Auto All **(O)**	A TRUE or FALSE value. If TRUE then if user has access to any "All" views, then pop-up visibility is All; otherwise pop-up visibility is determined by Popup Visibility Type property.	
Popup Visibility Type **(O)**	The type of pop-up visibility. If null, then the default value of Sales Rep is used.	Values are All, Personal, Sales Rep, Manager, Organization, Sub-Organization, Group, and Catalog.

Property	Description / Usage	Valid Values / Examples
PreFetch Size **(O)**	If this property if set, it overrides MaxCursorSize in the configuration file. Used in conjunction with the Maximum Cursor Size property (DB2 UDB only). PreFetch Size controls how many records are requested up front while the business component is being accessed. When you set this number to be the same as the Maximum Cursor Size during the execution call, the PreFetch Size number of records is read. Because that number is the same as Maximum Cursor Size as well, the database cursor is closed, freeing up the cursor to be used for the next query. Also works for MVG business components. For more information, read "MVG Business Component" on page 101.	
Recipient Id Field **(O)**	The ID for a recipient of correspondence, usually the contact ID. This is the field of the other business object that contains the ID, not the ID itself.	
Reverse Fill Threshold **(O)**	The maximum number of records to read when executing the "first record" or "last record" request before the SQL statement is automatically re-executed with a reversed sort sequence.	The default is 100.
Scripted **(S)**	A TRUE or FALSE value indicating whether the business component has a Siebel VB or Siebel eScript script attached to it.	TRUE — Siebel script attached. FALSE — No Siebel script attached.
Search Specification **(O)**	A conditional expression used to restrict the records retrieved.	The property text is limited to 255 characters.

Property	Description/Usage	Valid Values/Examples
Sort Specification **(O)**	A sort expression used to order the records returned. Use of a specialized business component class (that is, other than CSSBusComp) may disable a sort specification. Because predefined queries are shared across all visibility types, most likely the current sort specification may be slow in the All and Manager view modes. For this reason, Siebel applications force the sort in the All Visibility Mode to be on the primary key and on a column in the denormalized reporting relationship table for Manager mode. Users can still sort records after the initial query. For better performance, they should do so after filtering for a smaller record set. You cannot define a sort specification	The property text is limited to 255 characters. For more information, read "Sort Specification" on page 101
Status Field **(O)**	The name of a field whose value determines the New status for a record. If the specified field evaluates to TRUE, it indicates that the record is new. If FALSE, it indicates that the record has been viewed. This property should be used in conjunction with the visibility properties on the business component, such as Visibility MVLink and Position Visibility Field.	
Synonym Field **(O)**	A field (single- or multi-value) that contains synonyms for the business component. When this property is specified, bounded picklists on the business component search the synonyms for matches as well as the appropriate name field.	

Property	Description / Usage	Valid Values / Examples
Table **(R)**	The name of the primary SQL table where records are retrieved. This property is not used for virtual business components.	The table's name is case-sensitive and must be specified the same as it is in the database.
XA Attribute Value BusComp **(O)**	Specifies if the business component has extended dynamic attributes enabled. This property indicates which child business component stores the attribute values for this business component.	
XA Class Id Field **(O)**	Specifies if the business component has extended dynamic attributes enabled. This property indicates which field in this business component stores the class info.	

Restrictions on Inserts, Updates, and Deletes

Although other tables can be specified within a business component using a join, all INSERT, UPDATE, and DELETE operations are performed on the table defined by the business component's Table property.

Persistence

The persistence of a particular instance of a business component ends when the instance of the business object that is referencing it ends.

Cache Data

If you cache the contents of a business component and the contents are changed, the business component shows "stale" data. This occurs when the same user changes part of the contents through another business component (for example, a multi-value link business component) or "user two" changes the data while "user one" is still connected.

For this reason, you should use caching only with semistatic data like lists of values and products. Do not use caching with volatile data like accounts and opportunities because of the potential for the client-side cache to be out of sync with what is actually in the database.

Do not set both the Cache Data property of the Business Component object and the Use Primary Join property of the Multi Value Link object type to TRUE. This is because if the primary record on the MVG business component is changed, the update is not reflected in the (cached) parent.

MVG Business Component

Maximum Cursor Size and PreFetch Size properties also work for MVG business components. These properties are available for DB2 UDB only. The number of rows returned for the first operation is affected by the following:

- If the MVG business component is using aggregation (for example, the sum or count functions), every record is read.
- The MVG business component tries to make the primary record the current record. Therefore, the MVG business component loads records until the primary record is found. So, if the primary record is number 4000, then 4,000 records are loaded.

  To prevent this behavior of reading up to the primary record, you could use a detail view (for example, Contact-Address detail view, with the Contact in the top applet, and a list of addresses in the bottom applet) instead of the MVG.

Sort Specification

At times, it might appear that the All and Manager view modes are being ignored in the Siebel application. This is not always the case. The All Mode Sort business component user property also controls the behavior for the All and Manager view modes. For more information about this user property, read *Siebel Developer's Reference*.

See Also

"BusComp Server Script" on page 89
"Field" on page 177
"Join" on page 212
"Multi Value Field" on page 234

For information about user properties, read *Siebel Developer's Reference*. For information about virtual business components, read *Siebel Developer's Reference* or *Overview: Siebel Enterprise Application Integration*.

Business Component User Prop

Siebel Object Types > Business Component > Business Component User Prop

Communicates a property value to C++ code that implements specialized business component behavior. The values of these properties can be changed at configuration time. These values persist in the repository and the Siebel repository file.

Properties

Property	Description / Usage	Valid Values / Examples
Name **(R)**	The name of the user property.	
Value **(R)**	The value of the user property.	

Changing Undocumented User Properties

Developers may change user properties that have been documented; for example, user properties for business components of class CSSBCFile. Only Oracle developers who create or change specialized C++ code should create or delete user properties that are not documented.

See Also

"Applet User Prop" on page 41
"Business Component" on page 91
"Control User Prop" on page 154
"Field User Prop" on page 186
"List Column User Prop" on page 230

For information about user properties, read *Siebel Developer's Reference.*

Business Object

Siebel Object Types > Business Object

Groups one or more business components into a logical unit of information. For example, an Opportunity business object may consist of an Opportunity, Contact, and Product business component, with the Opportunity business component dictating the information of the others (parent-child relationships).

Properties

Property	Description/Usage	Valid Values/Examples
Name **(R)**	The name of the business object. The name must be unique among business objects. All references to the business object are made through its name.	The name can contain spaces but cannot contain special characters.
Primary Business Component **(O)**	The parent business component within the business object.	Valid values are Constraint Intersection, Scheduler Constraint Set, Scheduler Rule Manager, and Service Region.
Query List Business Component **(O)**	The name of the business component that is used to store predefined queries for the business object.	

See Also

"Business Component" on page 91
"Business Object Component" on page 103

Business Object Component

Siebel Object Types > Business Object > Business Object Component

Identifies the business component to include as part of a business object's logical unit of work. A business component can be included in one or more business objects.

Properties

Property	Description/Usage	Valid Values/ Examples
BusComp **(R)**	The name of the business component to include in the business object.	Example: Contact.
Link **(O)**	The name of a link to use between two business components within the business object. If this property is not specified, no link is established. See also "Link" on page 215.	Example: Account/ Contact.
Name **(S)**	The name of the business object component, unique within the business object.	Example: Contact.

See Also

"Business Component" on page 91
"Business Object" on page 102
"Link" on page 215

Business Service

Siebel Object Types > Business Service

Defines reusable business logic that can be executed within the Object Manager. There are two types of business services: built-in services that are defined in Siebel Tools, and run-time services that are defined in the Siebel client application by administrators.

Built-in business services can be based on the CSSService Class (standard business service) or specialized classes (specialized business service). Specialized business services are for Oracle use only.

You can configure business services with both properties and scripts (written in Siebel VB or Siebel eScript). Business services can be used for generic code libraries that can be called from any other scripts.

Business services are used as building blocks in the Siebel Application Integration framework. Commands associated with a menu item or toolbar button can also refer to a business service.

Specialized business services are for Oracle use only. Customers should not use specialized business services unless their behavior is documented.

Properties

Property	Description/Usage	Valid Values/Examples
Browser Class **(O)(H)**	Not used.	Not applicable.
Cache **(O)**	A TRUE or FALSE value.	TRUE — The business service instance remains in existence until the user's session is finished. FALSE — The business service instance is deleted after it has finished executing.
Class **(R)**	An underlying Siebel class, typically a data transport class. Customer-created business services must have the class CSSService. Specialized business services can be based on a different class.	Example: CSSService.
Display Name **(R)**	The name as displayed in the picklist.	
External Use **(O)**	A TRUE or FALSE value. Customers should always mark their scripts as External Use = TRUE.	TRUE — Business services can be used in scripts and be tested in the stimulator.
Hidden **(O)**	A TRUE or FALSE value.	TRUE — The object is hidden. FALSE — The object is visible.
Name **(R)**	The name of the business service.	
Server Enabled **(S)**	A TRUE or FALSE value.	TRUE — The business service is server enabled. FALSE — The business service is not server enabled.

See Also

"Business Service Browser Script" on page 105
"Business Service Locale" on page 105
"Business Service Method" on page 106
"Business Service Server Script" on page 110
"Business Service Subsystem" on page 111
"Business Service User Prop" on page 111

Business Service Browser Script

Siebel Object Types > Business Service > Business Service Browser Script

The browser script used by the Business Service object type. For more information, read *Siebel Object Interfaces Reference*.

Properties

Property	Description/Usage	Valid Values/ Examples
Browser Group **(O)**	Not used.	
Name **(S)**	Unique name for the script record. Do not modify this property. This is generated by Siebel Tools by combining the Browser Group and Procedure properties.	
Procedure **(O)**	Name of function or procedure.	
Script **(R)**	The text for the function.	

See Also

"Business Service" on page 103
"Business Service Locale" on page 105
"Business Service Method" on page 106
"Business Service Server Script" on page 110
"Business Service Subsystem" on page 111
"Business Service User Prop" on page 111

Business Service Locale

Siebel Object Types > Business Service > Business Service Locale

Represents language-specific overrides used with the Business Service object type.

Properties

Property	Description/Usage	Valid Values/ Examples
Display Name **(R)**	The name as displayed in the picklist for a specific language.	
Name **(R)**	The abbreviation of the language being used by the application.	Example: ENU.

See Also
"Business Service" on page 103
"Business Service Browser Script" on page 105
"Business Service Method" on page 106
"Business Service Server Script" on page 110
"Business Service Subsystem" on page 111
"Business Service User Prop" on page 111

Business Service Method

Siebel Object Types > Business Service > Business Service Method

Method associated with the Business Service. The information recorded about the Business Service Method objects serves as documentation for the various methods supported by the parent business service.

Properties

Property	Description/Usage	Valid Values/Examples
Display Name **(O)**	The name to display in various picklists.	
Hidden **(O)**	A TRUE or FALSE value.	TRUE — Makes the business service method invisible to picklists in the Workflow Process Designer. FALSE (default) — Does not make the business service method invisible.
Name **(R)**	The name of the business service method.	

See Also
"Business Service" on page 103
"Business Service Browser Script" on page 105
"Business Service Locale" on page 105
"Business Service Method Arg" on page 106
"Business Service Method Locale" on page 109
"Business Service Server Script" on page 110
"Business Service Subsystem" on page 111
"Business Service User Prop" on page 111

Business Service Method Arg

Siebel Object Types > Business Service > Business Service Method > Business Service Method Arg

The arguments to a business service method.

Properties

Property	Description/Usage	Valid Values/Examples
Data Type **(R)**	The data type of the business service method arg.	Valid values are Date, Hierarchy, Number, Integration Object, and String.
Display Name **(O)**	The name to display in the user interface.	
Hidden **(O)**	A TRUE or FALSE value.	TRUE — Makes the business service method arg invisible. FALSE (default) — Does not make the business method arg invisible.
Integration Object **(O)**	Name of the integration object that is used for service method argument.	The value can be chosen from picklist of the available integration objects. This fields active only if the Data Type property is set to Integration Object.
Name **(R)**	The name of the business service method arg.	
Optional **(O)**	A TRUE or FALSE value.	TRUE — Makes the business service method arg optional. FALSE — Makes the business service method arg required.
Pick Applet **(O)**	Name of the pick applet associated with the business service method arg.	
Pick Field **(O)**	Name of the pick field associated with the business service method arg.	
Picklist **(O)**	Name of the picklist associated with the business service method arg.	
Preferred Sequence **(O)**	In the case that a relative order of service method arguments matter to a business service, this property can be used to specify the order.	Integer value.

Property	Description / Usage	Valid Values / Examples
Storage Type **(R)**	The storage type.	Valid values are Hierarchy, Property, Type, and Value.
Type **(R)**	The type of business service method arg.	Input — The argument is part of the Input property set for the InvokeMethod call. Input / Output — The argument is part of the Input and Output property sets. Output — The argument is part of the Output property set.

See Also

"Business Service" on page 103
"Business Service Method" on page 106
"Business Service Method Arg Locale" on page 108
"Business Service Method Arg User Prop (H)" on page 109
"Business Service Method Locale" on page 109

Business Service Method Arg Locale

Siebel Object Types > Business Service > Business Service Method > Business Service Method Arg > Business Service Method Arg Locale

Represents language-specific overrides used with the Business Service Method Arg object type.

Properties

Property	Description / Usage	Valid Value / Examples
Display Name **(O)**	The name to display in the user interface for a specific language.	
Name **(R)**	The abbreviation of the language being used by the application.	Example: ENU.

See Also

"Business Service" on page 103
"Business Service Method" on page 106
"Business Service Method Arg" on page 106
"Business Service Method Arg User Prop (H)" on page 109

Business Service Method Arg User Prop (H)

Siebel Object Types > Business Service > Business Service Method > Business Service Method Arg > Business Service Method Arg User Prop

Communicates a property value to C++ code that implements special business services method argument behavior. The value of these properties can be changed at configuration time. These values persist in the repository and in the Siebel repository file.

Properties

Property	Description / Usage	Valid Value / Examples
Name **(R)**	The name of the user property.	
Value **(R)**	The value of the user property.	

Usage

These user properties are typically created when importing web service definitions (WSDL documents) and are used to enhance the metadata for creating SOAP documents at run-time.

See Also

"Business Service" on page 103
"Business Service Method" on page 106
"Business Service Method Arg" on page 106
"Business Service Method Arg Locale" on page 108
"Business Service Method Locale" on page 109

Business Service Method Locale

Siebel Object Types > Business Service > Business Service Method > Business Service Method Locale

Communicates a property value to C++ code that implements special business services method argument behavior. The value of these properties can be changed at configuration time. These values persist in the repository and in the Siebel repository file.

Represents language-specific overrides used with the Business Service Method object type.

Properties

Property	Description / Usage	Valid Values / Examples
Display Name **(O)**	The name to display in various picklists for a specific language.	
Name **(R)**	The abbreviation of the language being used by the application.	Example: ENU.

See Also

"Business Service" on page 103
"Business Service Method" on page 106
"Business Service Method Arg" on page 106
"Business Service Method Locale" on page 109

Business Service Server Script

Siebel Object Types > Business Service > Business Service Server Script

Script (Siebel VB or Siebel eScript) associated with a business service. For more information, read *Siebel Object Interfaces Reference*.

Properties

Property	Description / Usage	Valid Values / Examples
Name **(R)**	The name of the business service script.	
Program Language **(S)**	Indicates whether the programming language of the script is Siebel VB or Siebel eScript.	SBL — Siebel VB. JS — Siebel eScript.
Script **(R)**	The body of the script.	
Sequence **(R)**	The order of the script in compilation relative to the other scripts for the object.	Integer value.

Usage

Business service scripts are created and edited using the script editor.

See Also

"Business Service" on page 103
"Business Service Browser Script" on page 105
"Business Service Locale" on page 105
"Business Service Method" on page 106
"Business Service Subsystem" on page 111
"Business Service User Prop" on page 111

Business Service Subsystem

Siebel Object Types > Business Service > Business Service Subsystem

Subsystem associated with a business service.

Properties

Property	Description/Usage	Valid Values/ Examples
Name **(R)**	The name of the (Oracle internal) business service subsystem.	
Subsystem **(R)**	The name of the (external) business service subsystem.	

See Also

"Business Service" on page 103
"Business Service Browser Script" on page 105
"Business Service Locale" on page 105
"Business Service Method" on page 106
"Business Service Server Script" on page 110
"Business Service User Prop" on page 111

Business Service User Prop

Siebel Object Types > Business Service > Business Service User Prop

Provides initial values for the properties of the business service. This can be used to communicate a property value to C++ code or scripts that implement the business service behavior. The values of these properties can be changed at configuration time or run time. For supported properties, read *Siebel Developer's Reference*. Run-time access is by the GetProperty and SetProperty functions.

Properties

Property	Description/Usage	Valid Values/ Examples
Name **(R)**	The name of the business service user prop.	
Value **(R)**	The value of user property.	

Changing Undocumented User Properties

Customer developers can change only those user properties that have been documented. Only Oracle developers who create or change specialized C++ code should create or delete user properties that are not documented.

See Also

"Business Service" on page 103
"Business Service Browser Script" on page 105
"Business Service Locale" on page 105
"Business Service Method" on page 106
"Business Service Server Script" on page 110
"Business Service Subsystem" on page 111

For information about user properties, read *Siebel Developer's Reference.*

Chart

Siebel Object Types > Applet > Chart

Displays graphical data within the boundaries of its applet.

Properties

Property	Description/Usage	Valid Values/Examples
Bar Colors **(O)**	A list of colors for bars or pie slices in an itemized list of red, green, and blue triplets. The first triplet specifies the color for the first series, the second triplet for the second series, and so on. If the list contains fewer triplets than series members, the list of colors repeats.	Up to 15 triplets can be specified. Each number in a triplet must be between 0 and 255; for example, 128 0 0, 0 128 0, 0 0 128. The default value is 0 255 0,0 0 255,255 0 0,255 0 255,255 255 0,0 255 255,255 0 128,255 128 0,0 255 128,128 0 255,255 255 128,128 255 255,128 128 64,128 64 128,64 128 128.
Category Captions **(O)**	Caption for the x-axis.	

Property	Description/Usage	Valid Values/Examples
Category Field **(R)**	When the business components records are scanned, the different values found in this field get mapped into different categories. These values are displayed on the chart's x-axis labels. For more information, read "Category Field" on page 117.	
Compare **(O)(H)**	Obsolete. Do not use.	Not applicable.
Data Function **(O)**	An operation to perform on the data. If COUNT is the only data function you want to apply to the chart (because the Picklist Functions property is empty), you can achieve better performance by not specifying a Data Point Field property value. For more information, read "Data Function" on page 117.	COUNT — Counts the number of records. SUM — Sums the specified field values across all records in the category. AVERAGE — Calculates the average value for each record of the specified field. PLOT — Plots all cell values. When a cell in the logical table does not have a value, the chart's value is set to NULL instead of 0.
Data Point Captions **(O)**	Caption for the y-axis.	
Data Point Field **(O)**	A field to be charted whose data gets plotted on the y-axis. If this property is not specified, then specify COUNT as the Data Function value. For more information, read "Data Point Field" on page 117.	
Editable **(O)**	Obsolete. Do not use.	Not applicable.
Invert **(O)**	A TRUE or FALSE value that indicates if the x-axis and z-axis, and their labels and titles, can be flipped. Selecting a By picklist item changes this property (except in period charts, where it changes the period on the x-axis).	TRUE — Flips the x-axis, z-axis, labels, and titles. This may be useful for getting a different view of the data. FALSE — Does not flip the x-axis, z-axis, labels, and titles.

Property	Description/Usage	Valid Values/Examples
Multi Data Point **(O)**	A TRUE or FALSE value that plots multiple line-graph curves against the same y-axis based on different source field/function combinations. The name for each curve appears in the legend. To accomplish this, you need to set the following property values in the chart object definition: Data Point Field, Data Function, Picklist Functions, and Multi Data Point.	TRUE — Plot multiple curves. FALSE (default) — Multiple curves not plotted.
Name **(R)**	The name of the chart.	
Period **(O)**	The default period for this chart, either a calendar period or a user-defined period. If this property is specified, the field type of the Category business component must be Date or DateTime. A user-defined period can be any period defined in the Period business component, such as fiscal month, quarter, and year. For user-defined periods to work in a chart, the Period business component must be added to the current business object.	Valid values are Day, Week, Month, Quarter, and Year.
Picklist Function Captions **(O)**	Captions for the list of functions used by the y-axis.	

Property	Description/Usage	Valid Values/Examples
Picklist Functions **(O)**	The data functions to list in the applet's Show picklist. To suppress a function in the picklist, do not specify a string in the position of that function. For more information, read "Picklist Functions" on page 118.	By default, none is specified and the picklist does not appear in the applet. This property accepts a comma-separated list of y-axis titles, which are also the entries that appear in the picklist. The order of titles in the comma-separated list determines the association of each with a data point field and data function in the corresponding positions in the Data Point Field and Data Function properties. Example: Number of Opportunities, Opportunity Revenue, Average Revenue.
Picklist Period Captions **(O)**	Captions for the list of date periods used by the x-axis.	
Picklist Periods **(O)**	A comma-separated list of possible periods that appears in a picklist in the applet. Do not put a space after a comma in the list of values. For user-defined periods (fiscal periods) to work in a chart, the Period business component should be added to the current business object.	If not specified, a period picklist does not appear. Examples: Month(Calendar), Quarter (Calendar), Year(Calendar),Day,Week,Month, Quarter,Year.
Picklist Type Captions **(O)**	Captions for the types of charts.	
Picklist Types **(O)**	A comma-separated list of valid chart types other than 2dScatter, 3dScatter, or Combo (these three chart types are treated specially and can be specified only in the Type property). This property specifies which chart types are listed in the applet's picklist. Do not put a space after a comma in the list of values. For a list of chart types, read the description of the Type property.	If not specified, a picklist does not appear. Example: 2dBar,3dBar.

Property	Description/Usage	Valid Values/Examples
Selection Based **(O)**	A TRUE or FALSE property that specifies whether to chart only the currently active record in the business component.	TRUE — Chart only the currently active record in the business component. FALSE — Chart all records in the business component.
Series Captions **(O)**	Caption for the z-axis.	
Series Field **(O)**	Specifies a field for z-axis labels. When scanning the business component's records, Siebel applications map the different values found in this field into different series. These values are displayed on the chart's z-axis labels. When no series is specified, all of the records are mapped into a single series. For more information, "Series Field" on page 118.	
Type **(O)**	The type of chart to be displayed. For combo charts, a Data Point Field must be specified. Its Sum (Data Function) displays on the y-axis and its Count displays on the second y-axis. For 2dScatter and 3dScatter charts, the Coordinates property for the axis label of the x-axis must be specified (see the Chart Element object type).	Valid values include: ■ 2dBar, 3dBar, 2dStackedBar, 3dStackedBar, 2dLine, 3dLine, Combo, 2dHorizBar, 3dHorizBar, 2dCube, 3dCube ■ 3dClusteredBar, 2dPie, 3dPie, 2dDoughnut ■ 2dScatter, 3dScatter, 2dSpline, 3dSpline
Use Category MVGroup Data **(O)**	A TRUE or FALSE value.	TRUE — If the category field is a multivalue group (MVG), Siebel applications must obtain the charting data from the field. **NOTE:** You cannot use the Data Function, Data Point Captions, and Data Point Field properties when setting this property to TRUE. FALSE (default) — Use the value listed in the business component.

Comma-Separated Lists

When specifying a comma-separated list of values, do not include a space after the comma.

Category Field

If the Category Field property contains the name of a field from the business component, the x-axis labels are derived from the contents of that field. The By picklist in this case offers the choice between using the x-axis or the z-axis to display the field values.

If the Category Field property contains a comma-separated list of field names, the user is provided with this list of fields at run time in the By picklist. The user's selection determines the field that populates the x-axis. The first value in the comma-separated list is the default. Avoid blank spaces before or after field names. The properties of the chart element that apply to the axis label for the x-axis (Coordinates, Display Format, Divisions, List Of Values, Sort Specification, and Text) should not be used when specifying a list of x-axis source fields, as they can only be relevant for one x-axis field.

If a new field is created in the business component on which the chart is based, and this new field is mapped on to a Calculated field in another business component through a multi-value link and this same new field is used as a category field for the chart, then the drilldown on the chart breaks. The drilldown breaks because the Calculated field cannot preserve the context.

Data Function

The Show picklist can be configured to display a selection list of field/function combinations, the selection from which determines what values are plotted along the y-axis. To configure this, a comma-separated list is entered in the Data Point Field, Data Function, and Picklist Functions properties. In the Data Function property, the comma-separated list consists of SUM, COUNT, AVERAGE, and PLOT entries. If the Data Function property is empty or shorter than the Picklist Functions list, the list Sum,Count,Average,Plot is used.

Multiple line-graph curves can be plotted against the same y-axis, based on different source field/function combinations. The name for each curve appears in the legend. To accomplish this, set the following property values in the chart object definition: Data Point Field, Data Function, Picklist Functions, Category Field, and Multi Data Point (Series Field must be empty). In the Data Function property, provide a comma-separated list consisting of the following function names: SUM, COUNT, AVERAGE, and PLOT.

Two line-graph curves can appear in the same chart, plotted against different y-axes (one to the left of the graph, the other to the right). To accomplish this, in the Data Function property specify two functions, separated by a comma. The first is for the left y-axis, the second is for the right y-axis. You also must specify two fields in the Data Point Field property.

Data Point Field

The Show (ChartPickfunction) picklist can be configured to display a selection list of field/function combinations, the selection from which determines what values are plotted along the y-axis. To configure this, a comma-separated list is entered in the Data Point Field, Data Function, and Picklist Functions properties. In the Data Point Field property, the comma-separated list consists of the names of source fields, one for each entry that is to appear in the Show picklist. The first entry in the list is the default. If only one field is specified, all entries in the Show picklist use the same field.

Multiple line-graph curves can be plotted against the same y-axis, based on different source field/function combinations. The name for each curve appears in the legend. To accomplish this, set the following property values in the chart object definition: Data Point Field, Data Function, Picklist Functions, Category Field, and Multi Data Point (Series Field must be empty). In the Data Point Field property, provide a comma-separated list of source fields, one for each curve to appear in the graph.

Two line-graph curves can appear in the same chart, plotted against different y-axes (one to the left of the graph, the other to the right). To accomplish this, in the Data Point Field property specify two fields, separated by a comma. The first is for the left y-axis, the second is for the right y-axis.

Picklist Functions

The Show (ChartPickfunction) picklist can be configured to display a selection of Field/function combinations, the selection from which determines what values are plotted along the y-axis. To configure this, a comma-separated list is entered in the Data Point Field, Data Function, and Picklist Functions properties. In the PicklistFunctions property, the comma-separated list consists of y-axis titles, which are also the text that appears in the picklist. The list of titles needs to have the same number of entries as the Data Point Field list.

Multiple line graph curves can be plotted against the same y-axis, based on different source field/function combinations. The name for each curve appears in the legend. To accomplish this, set the following property values in the Chart object definition: Data Point Field, Data Function, Picklist Functions, Series Field, and Multi Data Point. In the Picklist Functions property, provide a comma-separated list of the y-axis titles, which are what identify the individual curves in the legend.

Series Field

If the Series Field property contains the name of one field from the business component, the z-axis labels are derived from the contents of that field.

If the Series Field property contains a comma-separated list of field names, the user is provided with this list of fields at run time in the second By picklist. The user's selection determines the field that populates the z-axis. The first value in the comma-separated list is the default. The properties of the chart element that apply to the axis label for the z-axis (Coordinates, Display Format, Divisions, List Of Values, Sort Specification, and Text) should not be used when specifying a list of z-axis source fields, as they can only be relevant for one z-axis field.

See Also

"Applet" on page 29
"Applet Method Menu Item" on page 37
"Applet Script (H)" on page 39
"Applet Toggle" on page 40
"Control" on page 139
"List" on page 220
"View" on page 325

Chart Element

Specifies one of the static characteristics of a chart, including title, legend, labels, chart body, and axes.

Properties

The following table shows which properties apply to which chart elements.

Chart Element	Applicable Properties
Axis Label	Axis Id, Color, Coordinates, Display Format, Divisions, Font Effect, Font Name, Font Size, Font Style, List Of Values, Log Base, Show Category Legend, Sort Specification, Vertical
Axis Line Grid	Axis Id, Color, Line Style, Line Width **NOTE:** The AxisLineGrid element for XAxis is not configurable.
Axis Title	Axis Id, Color, Font Effects, Font Name, Font Size, Font Style, Text
Graphic	Color, Coordinates, Graphic Type, Line Style, Line Width, Text
Legend	Color, Font Effect, Font Name, Font Size, Font Style **NOTE:** Font Name and Font Size properties are not configurable.
Plot	Color **NOTE:** Plot element is not supported in Release 7.x.
Title	Color, Font Effect, Font Name, Font Size, Font Style, Text

The following table lists the Chart Element properties.

Property	Description/Usage	Valid Values/Examples
Axis Id **(O)**	The axis to which subsequent axis elements apply. Applies to the AxisLabel, AxisLineGrid, and AxisTitle elements.	Valid values are XAxis, YAxis, Y2Axis, and ZAxis.
Color **(O)**	For elements containing text, sets the text color. For the Plot element, the chart element, and the HorizArea graphic, sets the fill color. For the Axis Line Grid element and the Line, Rectangle, Ellipse, and HorizLine graphics, sets the line color. Applies to all elements.	Valid values are a space-separated red, green, blue triplet with each value in the range 0–255.

Property	Description/Usage	Valid Values/Examples
Coordinates **(O)**	Specifies coordinates for AxisLabel and Graphic elements. Rectangle, Ellipse, and Line apply only to scatter charts, whereas HorizLine and HorizArea apply to all charts. For histogram charts, coordinates are required for the axes that are being “bucketized” (the Divisions property is also required for histogram charts). Applies to the Graphic and AxisLabel elements. Required for Graphics, for x-axis labels in scatter charts, and for x-axis and z-axis labels in histogram charts.	For AxisLabel, specifies the minimum and maximum values for the axis (Axis Id); two numbers are required (min max). For Rectangle and Ellipse, four numbers are required (left bottom top right). For Line, four numbers are required (x1 y1 x2 y2). For HorizArea, two numbers are required (minYaxisValue maxYaxisValue). For HorizLine, one number is required (YaxisValue). Examples: AxisLabel: 0 1000000 (range between 0 and 1 million) HorizLine:40000 (horizontal line drawn at y = 40000) HorizArea:20 50 (horizontal area drawn between y = 20 and y = 50) Rectangle, Ellipse, or Line: 60 1000 80 3000 (coordinates of graphic)
Display Format **(O)**	Formats labels on the specified axis (Axis Id). When this property is specified, the axis labels are passed through the formatting string before they are displayed in the chart. Otherwise, the default formatting for the business component Category, Series, or Data Point Field is used for the x-, z-, or y-axis. Applies to the AxisLabel element.	Valid values include any formatting string that Display Format accepts; for example, 0.##%.

Property	Description/Usage	Valid Values/Examples
Divisions **(O)**	Sets the number of divisions in the specified axis (Axis Id) for axis labels (AxisLabel element). For example, if the Divisions value is 4, five labels are drawn on the axis. This property always applies to the y-axis. For scatter charts, it also applies to the x-axis. For other charts, this property applies to the x-axis or z-axis (where the field values are numbers), in which case the data is "bucketed" into a histogram chart. (The Coordinates property must also be set for histogram charts.) The axis of interest must show numeric data. Otherwise, labels display and there is no bucketing. Applies to the AxisLabel element. Required only for x-axis and z-axis in histogram charts.	Valid values include numbers greater than 0. If no value is specified, the number of divisions is calculated automatically.
Font Effect **(O)(H)**	Identifies the effect to use in drawing the element's text. Applies to the AxisLabel, AxisTitle, Legend, and Title elements.	Valid values are Regular, Underline, StrikeThrough, and StrikeThroughUnderline.
Font Name **(O)**	Specifies the name of the font to use for the element's text. Applies to the Axis Label, Axis Title, and Title elements.	
Font Size **(O)**	Specifies the font size to use for the element's text. This font size applies when the chart is minimized and the screen resolution is 1024 by 768 or higher. When the chart is maximized, Siebel applications use a larger font. When the screen resolution is lower, Siebel applications use a smaller font. Applies to the Axis Label, Axis Title, and Title elements.	
Font Style **(O)(H)**	Specifies the font style to use for the element's text. Applies to the AxisLabel, AxisTitle, Legend, and Title elements.	Valid values include Regular, Bold, Italic, Outline, BoldItalic, BoldOutline, ItalicOutline, and BoldItalicOutline.

Property	Description/Usage	Valid Values/Examples
Graphic Type **(O)**	Specifies the type of Graphic element to draw. The geometry of the given graphic is specified in the Coordinates property. The Graphic Type property must be set when the chart element has the type Graphic. Applies to the Graphic element.	Valid values include Ellipse, HorizArea, HorizLine, Line, and Rectangle. Ellipse, Line, and Rectangle apply only to scatter charts (2dScatter or 3dScatter). HorizArea and HorizLine apply to all charts.
Line Style **(O)**	Specifies the line style for a Graphic or AxisLineGrid element. This property applies only when the Line Width property is set to 0 (or left empty). Applies to the AxisLineGrid and Graphic elements.	Valid values are Solid, Dash, Dot, DashDot, and DashDotDot.
Line Width **(O)**	Specifies the line thickness for an AxisLineGrid or Graphic element. Applies to the AxisLineGrid and Graphic elements.	Any positive number is a valid value. The default is 0, a thin line.

List Of Values **(O)**	Describes the list of possible values for the specified x-axis, y-axis, or z-axis (Axis Id). Setting this property, which applies to the AxisLabel element, enables the user to specify values that are not represented (such as 0 or an empty value).
	
	If this property is applied to the y-axis, then the string values are shown on the y-axis instead of numeric values.
	
	Applies to the AxisLabel element.
		By default, only the corresponding field values found in the current business component are used for a given axis.
		
		A valid value for this property is in the following format:
		
		Business Component,Field Name,Search Spec,Sort Spec.
		
		This comma-separated list includes a business component that is part of the current business object, a field from which to obtain the values, a specification of the search (Search Spec) to be performed on the business component, and a specification of the sort (Sort Spec) to be applied to the business component. For example:
		
		`List Of Values,Value, Type = LEAD_QUALITY,Order By`
		
		and
		
		`Sales Cycle Def,Sales Cycle Stage,Sales Stage Order.`

Property	Description/Usage	Valid Values/Examples
Log Base **(O)**	Sets an axis to be logarithmic instead of linear. Applies to the AxisLabel element.	Possible values include any positive number. For example, if the value of LogBase is 10, the axis shows 1, 10, 100, 1000, 10000, and so on; if the value is 2, the axis shows 1, 2, 4, 8, 16, 32, 64, and so on.
Name **(R)**	The name that is unique within the chart.	
Show Category Legend **(O)**	Specifies whether to show a legend window for the given axis (Axis Id). This property applies to the AxisLabel element, for only the x-axis and z-axis. If the x-axis labels are long or if there are many labels, the labels might overlap. Applies to the AxisLabel element.	Value is TRUE or FALSE. If TRUE, always shows the entire x-axis group label along the x-axis. By default, whenever there is more than one group on the x-axis, an x-axis legend window is shown, and only the first three letters of each x-axis group are shown along the x-axis.
Sort Specification **(O)**	Specifies the sorting operation to be performed on the given axis (Axis Id). A field name appended to one of these values specifies sorting based on a column other than the Category or Series. This property applies to the AxisLabel element, for the x-, y-, or z-axes. It does nothing when the List Of Values property is set. Behavior of the x-axis and z-axis is as described in the previous paragraph. Behavior of the y-axis is to sort the x-axis labels based on y-axis numeric values. If there is a Sort Specification value for the y-axis, there cannot be one in the x-axis.	Valid values are Ascending, Descending, and None. Example: `Ascending,Sales Stage Order` specifies a sort based on Sales Stage Order instead of Sales Stage.
Text **(O)**	Specifies the text to display for the element. Applies to the AxisTitle, Legend, and Title elements. It also applies to the HorizLine Graphic element.	

Property	Description/Usage	Valid Values/Examples
Type **(O)**	Identifies the type of chart element. Applies to all elements.	Valid values are AxisLabel, AxisLineGrid, AxisTitle, Graphic, Legend, Plot, Title, Font, Color, Size, and Fill Color.
Vertical **(O)**	Specifies whether labels are displayed vertically when shown on the x-axis. Applies to the AxisLabel element. Not all fonts can be displayed vertically; experiment with your selected font to verify if it can be used. If the x-axis has fewer than five labels, the labels are displayed horizontally regardless of the value of this property. Long labels display vertically, and this causes the chart to be moved upward. Space is especially limited in low-resolution mode. The vertical property does not apply to a Period chart. Period charts are formatted and displayed in different manner.	Value is TRUE or FALSE. If TRUE, displays the labels vertically.

See Also

"Applet" on page 29
"Chart" on page 112
"Hidden Object Types and Properties" on page 21
"View" on page 325

Chart Element Locale

Siebel Object Types > Applet > Chart > Chart Element > Chart Element Locale

Represents language-specific overrides used with the Chart Element object type.

Properties

Property	Description/Usage	Valid Values/ Examples
Name **(R)**	The abbreviation of the language being used by the application.	Example: ENU.
Text **(O)**	The name of the chart element in the user interface for a specific language.	

See Also
"Chart" on page 112
"Chart Element" on page 119

Chart Locale

Siebel Object Types > Applet > Chart > Chart Locale

Represents language-specific overrides used with the Chart object type.

Properties

Property	Description/Usage	Valid Values/ Examples
Category Captions **(O)**	Caption for the x-axis for a specific language.	
Data Point Captions **(O)**	Caption for the y-axis for a specific language.	
Name **(R)**	The abbreviation of the language being used by the application	Example: ENU.
Picklist Function Captions **(O)**	Captions for the list of functions used by the y-axis for a specific language.	
Picklist Period Captions **(O)**	Captions for the list of date periods used by the x-axis for a specific language.	
Picklist Type Captions **(O)**	Captions for the types of charts for a specific language.	
Series Captions **(O)**	Caption for the z-axis for a specific language.	

See Also
"Chart" on page 112

Class

Siebel Object Types > Class

Class is a property of certain object types, such as applet or business component. The object type contains the listing of classes used in the repository.

CAUTION: Do not modify this object type unless adding a new ActiveX control or Java Applet to the web user interface. Any modification can adversely affect performance and operation.

Properties

Property	Description/Usage
DLL **(S)**	Specifies which DLL the class resides in.
Handheld Client **(S)**	A TRUE or FALSE value. If TRUE, the class can be used for the handheld client.
High Interactivity Enabled (**S**)	Specifies a view as a high interactivity view. When every applet in a view has its underlying classes marked High Interactivity Enabled, then the view is considered to be a high interactive view. For more information, read "High Interactivity Enabled" on page 126
Java Thin Client **(S)(H)**	A TRUE or FALSE value. If TRUE, the class can be used for the Java thin client.
Name **(S)**	The name of the C++ class.
Object Type **(S)**	The name of the object type associated with the class.
Super Class **(S)**	The name of the parent (super) class.
Thin Client **(S)(H)**	A TRUE or FALSE value. If TRUE, the class can be used for the thin client.
UNIX Support **(S)(H)**	Signifies if the class is supported on the UNIX platform.

High Interactivity Enabled

When enabled, the layout of a view is loaded from the server the first time a user navigates to it. The layout loads again (and the view frame containing the applet refreshed) only if some action requires the applet to be displayed using a different template.

In a high interactive view, a page refresh is expected only when the layout changes.

The following table represents the values for the High Interactivity Enabled property.

Value	Works in High Interactivity	Works in Standard Interactivity	Cacheable
1	No	Yes	No
2	Yes	No	Yes
3	Yes	No	No
4	Yes	Yes	Yes
5	Yes	Yes	No

See Also

"Hidden Object Types and Properties" on page 21

Class Method

Siebel Object Types > Class > Class Method

CAUTION: Do not modify this object type. Any modification can adversely affect performance and operation.

Properties

Property	Description / Usage
Method **(S)**	Method that is invoked by the class.
Name **(S)**	Name of the method.

See Also

"Class" on page 125

Class Method Menu Item

Siebel Object Types > Class > Class Method Menu Item

Adds (or suppresses) a menu item on applet-level menus for all Siebel Web Engine applets of the specified applet class and its subclasses.

CAUTION: Do not modify this object type. Any modification can adversely affect performance and operation.

Properties

Property	Description / Usage
Command **(S)**	Name of the Command object type that provides the method and target for the menu item.
Menu Text **(S)**	The text displayed in the menu item.
Name **(S)**	Integer value that contains the same information as the position property.
Position **(S)**	The sequence of the menu item in the single-level list of menu items. Integer value.
Suppress Menu Item **(S)**	A TRUE or FALSE value. Default is FALSE. If TRUE, causes the applet-level menu items of the specified name to be removed from the applet-level menu in all applets derived from this class and its subclasses.

See Also
“Class” on page 125
“Class Method” on page 127
“Class Method Menu Item Locale” on page 128

Class Method Menu Item Locale

Siebel Object Types > Class > Class Method Menu Item > Class Method Menu Item Locale

Represents language-specific overrides used with the Class Method Menu Item object type.

Properties

Property	Description/Usage	Valid Values/ Examples
Menu Text **(S)**	The text displayed in the menu item.	
Name **(S)**	The abbreviation of the language being used by the application.	Example: ENU.

See Also
“Class Method Menu Item” on page 127

Class User Prop

Siebel Object Types > Class > Class User Prop

A user property that can be set on Siebel repository objects that are based on this class.

Properties

Property	Description/Usage	Valid Values/Examples
Comments **(O)**	General comments about the user property that are not part of the functional or technical description of it.	
Description **(O)**	A description of the behavior of the user property. This description appears in the pick list where the user property is invoked so the user can quickly understand the usage and implications of the user property.	For example, in the All Mode Sort user property: "Standard Siebel application behavior is to override the sort specification on views with certain visibility types to force it to ORDER BY the standard user key. The All Mode Sort user property determines whether or not the Siebel application overrides the sort specification and, if so, determines the sort (if any) that is applied to the business component for the affected views."
Name **(R)**	The name of the user property. This is the name used by the class's C++ code when retrieving the value of the user property. It is also the name that appears in the user property pick list under various objects in Siebel Tools (for example, Business Component > Business Component User Prop.	Example: All Mode Sort.
Type **(R)**	The Siebel repository object type to which this user property applies.	Business Component, Business Service, Applet
Value Type **(O)**	The type of the value entered in this user property, if applicable. If the user property holds plain text this property may be left blank.	Field, Link, Business Service

See Also

"Class" on page 125
"Class User Prop Value" on page 129

For information about user properties, read *Siebel Developer's Reference*.

Class User Prop Value

Siebel Object Types > Class > Class User Prop > Class User Prop Value

Defines a value that appears in the picklist for its parent Class User Prop.

Properties

Property	Description/Usage	Valid Values/Examples
Comments **(O)**	General comments about the user property value that are not part of the functional or technical description of it.	
Description **(O)**	A description of the behavior of the user property value. This description appears in the picklist where the user property value is used so the user can quickly understand the usage and implications of this value as compared to other values in the picklist.	For example, in the All Mode Sort user property value of Normal: "Uses the business component-defined sort specifications. This setting also allows the user to run a PDQ (that incorporates a SORT)."
Name **(R)**	The name of the user property value. This is the name that appears in the user property value picklist under various repository objects in Siebel Tools (for example, Business Component > Business Component User Prop).	Examples: TRUE, FALSE, Normal.

See Also

"Class" on page 125
"Class User Prop" on page 128

For information about user properties, read *Siebel Developer's Reference*.

Column

Siebel Object Types > Table > Column

Corresponds to one column in a database table that is represented by the parent table object definition.

Properties

Property	Description/Usage	Valid Values/Examples
Alias **(O)**	Not used.	Not applicable.
ASCII Only **(O)**	A TRUE or FALSE value.	TRUE — ASCII text is used within the columns. The default is FALSE.
Cascade Clear **(O)**	For Oracle use only.	
Default **(O)**	For Oracle use only.	
Denormalization Path **(O)**	For Oracle use only.	

Property	Description/Usage	Valid Values/Examples
EIM Processing Column Flag **(O)**	For Oracle use only.	
FK Column 1:M Rel Name **(O)**	For Oracle use only.	
FK Column M:1 Rel Name **(O)**	For Oracle use only.	
Force Case **(O)**	Indicates the case of the letters in the column.	Valid values are FirstUpper, Lower, None, and Upper.
Foreign Key Table (Foreign Key Table Name) **(O)**	Specifies the table to which this column is a foreign key, used by EIM. Leave NULL in extension columns.	
LOV Bounded **(O)**	A TRUE or FALSE value. If LOV Bounded is TRUE, EIM checks the values as it imports against the values contained in a list defined in LOV Type. In that case, LOV data should be imported first into S_LST_OF_VAL, and LOV Type must be specified. This property is read-only for standard columns in Siebel applications but is editable for columns that have been added by way of the Database Extension Designer.	TRUE — Requires that a field based on this column be populated with a value from the list defined in LOV Type. FALSE — There is no restriction on values entered for a field based on this column.
LOV Type **(O)**	Specifies the list of values domain in which this column is validated. Used in conjunction with the LOV Bounded property. List of values domains are defined in List of Values Administration in the client. This property is read-only for standard columns in Siebel applications but is editable for columns that have been added by way of the Database Extension Designer.	
Name **(R)**	Specifies the name, which must be unique within the table.	

Property	Description / Usage	Valid Values / Examples
No Match Value **(O)**	A string value used to indicate unmatched primary Id fields. Should only be configured for the primary key column of the base table of a MVG External Business Component. If this is left blank, the default string “No Match Row Id” is used.	Typically this property should be left blank. It should be set to an alternative string value only for the base tables of External Business Components that generate their own RowId values that might conflict with the default “No Match Row Id” string.
Nullable **(O)**	A TRUE or FALSE value.	TRUE — NULL can be stored in this column by the database. FALSE — NULL cannot be stored in this column by the database.
Physical Type Name (Physical Type) **(R)**	Identifies the data type of the column in the database.	■ Character — Fixed-length text. ■ Long — Long text. ■ Varchar — Variable-length text. ■ Number — Any numeric data. ■ Date — Date values only, without time. ■ Date Time — Combined date and time values in the same column. ■ Time — Time values only, without the date.
Precision **(R)**	The total number of digits in a number column. For noninteger columns, the precision is 22. For integer columns, the precision is 10.	
Primary Child Col **(S)**	Used by EIM. For Oracle use only.	

Property	Description/Usage	Valid Values/Examples
Primary Child Column Name **(O)**	The name of the column that contains the primary ID value.	
Primary Child Join Column Name **(O)**	The name of the child table column on which to join the base record.	
Primary Child Table Name **(O)**	The name of the table from which the primary child is to be found.	
Primary Inter Table Name (Primary Inter Table) **(O)**	Used by EIM. For Oracle use only.	
Primary Join Column Name **(O)**	The name of the column on which to join to the base record.	
Primary Key **(O)**	A TRUE or FALSE value.	TRUE — Consider this column part of the primary key of the table.
Required **(O)**	A TRUE or FALSE value.	TRUE — The application requires this column. For example, even if the database allows NULLs, the application does not.
Scale **(O)**	The number of decimal digits in a number column, out of the total number of digits (precision). For noninteger columns, the scale is 7. For integer columns, the scale is 0.	
Sequence **(O)**	Represents the sequence with which the columns of a cluster index are physically stored. Although this property is set primarily with the MSSql platform in mind, it should be applicable to all other platforms as well.	An integer value.
Sequence Object **(O)**	Specifies the name of the RDBMS utility that generates a unique sequence number for the table column. For Oracle use only.	
Status	Indicates the state of the column.	Active, Inactive, EOL (end of life), and Not Used.

Property	Description/Usage	Valid Values/Examples
System Field Mapping **(O)**	Used to indicate alternative mapping of columns to system fields. Should only be configured for tables used by external business components.	"Conflict Id", "Created", "Created By", "Extension Parent Id", "Mod Id", "Non-system", "Updated", "Updated By", and "Id". Should be left blank for all Siebel business components.
Text Length **(O)**	The length in characters for Text and Varchar columns.	
Translation Table Name **(O)**	The name of the translation table.	Values are S_LST_OF_VAL and NULL. Enables multilingual list of value functionality when set to S_LST_OF_VAL.
Txn Log Code **(O)**	Not used.	Not applicable.
Type **(S)**	Describes what this column is used for.	Values include Data (Public), Data (Private), Extension, Denormalized, and System.
Use FKey **(S)**	A TRUE or FALSE value.	TRUE — Indicates that this foreign key is to be used by synchronization in evaluating visibility. Leave FALSE in extension columns.
User Key Sequence **(O)**	The sequence in the user key where this column fits.	
User Name **(R)**	The "user-friendly" name for the column.	
Valid Condition **(O)**	The expression for validating column values.	

Properties for Custom Extension Columns

When configuring a custom extension column, you should only set the following properties (at a maximum): Comments, Default, Foreign Key Table Name, Inactive, LOV Bounded, LOV Type, Name, Nullable, Physical Type, Precision, Scale, Text Length.

Inactive Property

If the Inactive property is set to TRUE, the column is not created with the database server installation.

See Also
"Table" on page 292

Command

Siebel Object Types > Command

Specifies the method invoked when a toolbar icon associated with this command is executed. It also defines which bitmap appears on the toolbar icon. Command object definitions are referenced by Toolbar Item or Menu Item object definitions.

Properties

Property	Description/Usage	Valid Values/Examples
Bitmap **(R)(H)**	Not used.	Not applicable.
Business Service **(O)**	Specifies the service (either browser or server, depending on the target property) that handles the invoke method.	
Category **(R)**	The category name (for the customize dialog).	
Category Display Name **(O)**	Not used.	Not applicable.
Display Name **(O)**	Name of the command. Used as display name for a menu if menu does not have one.	
Dynamic Tooltip **(O)**	A TRUE or FALSE value indicating whether the tooltip is dynamic or static. Reserved for future use.	TRUE — A call is made to the underlying object for such an application, applet, or service to retrieve the dynamic tooltip text.
Force Enable **(O)**	Indicates whether the menu item or toolbar item associated with this Command object is always enabled. Reserved for future use.	
HTML Bitmap **(O)**	Used to set the name of the bitmap to be rendered when the control is in an enabled state. For example, the method can be invoked.	
HTML Disabled Bitmap **(O)**	Used to set the name of the bitmap to be rendered when the control is in a disabled state.	

Property	Description / Usage	Valid Values / Examples
HTML Popup Dimension **(O)**	Dimensions, in pixels, of the pop-up window, when Show Popup is TRUE.	Example: 640x480. Specified with the “x” and without blank spaces.
Method **(R)**	Specifies the name of the method to invoke when the menu item or toolbar icon is selected. This may be a preexisting standard method that is shipped with the product, or a developer-defined method in Siebel VB or Siebel eScript.	
Method Argument **(O)**	Provides the means to pass an argument to the invoke method specified in the Method property. For example, a command item that opens a new window and navigates to a URL in that window can specify the GotoURL method in Method and the URL to navigate to in Method Argument.	
Name **(R)**	The name of the command object definition. This is the name of the toolbar icon as it appears in the Commands list on the Commands tab of the Customize dialog box in Siebel applications.	
Show Popup **(O)**	A TRUE or FALSE value.	TRUE — Specifies that a new browser window is opened before invoking the method. FALSE — Specifies that the method is invoked in the current browser window.
Status Text **(O)**	Not used.	Not applicable.

Property	Description/Usage	Valid Values/Examples
Target **(O)**	The name of the target. If your target is Active Applet, your item is visible in all clients. If your target is Service, your command appears only in clients implementing that particular service. Your service must handle CanInvokeMethod and InvokeMethod to enable or invoke your command.	Valid values are Active Applet and Service. Do not use Application as a value. It is for Oracle use only.
Tooltip Text **(O)**	The tooltip text that appears when the pointer lingers on a toolbar icon. Reserved for future use.	

See Also

"Bitmap" on page 86
"Bitmap Category" on page 87
"Hidden Object Types and Properties" on page 21
"Toolbar" on page 314
"Toolbar Item" on page 315

Command Locale

Siebel Object Types > Command > Command Locale

Represents language-specific overrides used with the Command object type.

Properties

Property	Description/Usage	Valid Values/ Examples
Category Display Name **(O)**	Not used.	Not applicable.
Display Name **(O)**	The name of the command in the user interface.	
Name **(R)**	The abbreviation of the language being used by the application.	Example: ENU.
Status Text **(O)**	Not used.	Not applicable.
Tooltip Text **(O)**	The tooltip text that appears when the pointer lingers on a toolbar icon. Reserved for future use.	

See Also
"Command" on page 135

Content Object

Siebel Object Types > Content Object

Specifies a class of content for Content Center. Content Objects define the structure (fields and attachments) of content transferred from a staging environment to a production environment when published in Content Center.

Properties

Property	Description/Usage	Valid Values/Examples
Business Object **(O)**	Name of the business object.	Any business object with a primary business component.
Integration Object **(O)**	Name of the integration object.	Any EAI object that is created out of the business object.
Name **(R)**	Name of the content object.	
Pick Applet **(O)**	Name of the pick applet that should be used for selecting the content data for a content item.	Any pick applet that has its business component matching the primary business component of the business object.
Pick List **(O)**	Name of the pick list for the pick applet.	Any pick list that has its business component matching the pick applet business component.
Sequence **(O)**	When publishing a content project comprising content items of different content objects, the content items of content objects with lower sequence number is published prior to those with higher sequence number.	Integer value.
Source Field **(O)**	The field from the pick list that would be picked and filled into the content item reference ID.	Always use the value Id.

See Also
"Business Object" on page 102

Content Object View

Siebel Object Types > Content Object > Content Object View

Specifies a Siebel View used to work with a class of content in the Content Center.

Properties

Property	Description/Usage	Valid Values/Examples
Name **(R)**	The name of the content object view. It is recommended that the name of the content object view be the same as the view name.	
Sequence **(O)**	Multiple views can be associated with a content object. For drill-down purposes, the application looks up the view that has the lowest sequence value. Any views for which the user does not have visibility are ignored.	Integer value.
View Name **(O)**	The name of the view to which you want to have content contributor/approver drill down to from a content item for editing or adding the associated content data.	Any view. Usually, it is the view that has its top applet showing the business component matching the primary business component of the business object of the content object.

See Also

"Content Object" on page 138

Control

Siebel Object Types > Applet > Control

Defines a visual element in an applet—such as a text box, check box, or button—for user interaction, user data entry, or data display.

You can add user properties to controls.

Properties

Property	Applies To	Description/Usage	Valid Values/ Examples
ActiveX Bind Property **(O)(H)**	ActiveX	The name of an internal ActiveX Control property in the ActiveX Bind Property. You must also specify the name of a field in the business component of the applet in the Field property, to bind the specified field to the ActiveX control's specified property. When the ActiveX control's property value changes, this value is saved to the field in the current business component record. When the field value changes, this value is saved to the ActiveX Control property.	
ActiveX Properties **(O)(H)**	ActiveX	A compressed version of the set of properties internal to the control. You do not edit this property setting directly. The values it stores are changed using the ActiveX category of properties in the Properties window or through the control's native property window.	
Automatic Horizontal Scroll **(O)(H)**	TextBox	A TRUE or FALSE value.	TRUE — Automatically scrolls text to the right by ten characters when the user types a character at the end of the line. When the user presses Enter, the control scrolls all text back to position 0.
Automatic Vertical Scroll **(O)(H)**	TextBox	A TRUE or FALSE value.	TRUE — Automatically scrolls text up one page when the user presses ENTER on the last line.

Property	Applies To	Description / Usage	Valid Values / Examples
Background Color **(O)(H)**	See the description	A space-separated RGB value. Applies to Textbox, Label, PushButton, ComboBox, and Group. For Group, only the area under the caption gets set to the color. In addition, the Background Color value can be set to Transparent for Label, Image, and PushButton. This causes the control to become transparent. If the attribute is set to a value other than Transparent, these controls appear opaque.	
Bitmap **(O)(H)**	Image	The bitmap (from the category Control Icons) to be used by the image control. This applies to static image controls that are not tied to a specific field in the current record.	
Caption **(O)**		Used as headers for list applet columns, field prompts, and link names. For more information, read "Caption" on page 152.	Valid choices are bitmaps in the bitmap category Button Icons.
Class **(O)**	Java and ActiveX	Specifies the name of the C++ class associated with the control.	
Content Fixup Name **(O)**		Determines if the links and images within the HTML page should be "fixed up" to be proxied through the Siebel server (Inside Application) or left alone to be requested by the client's browser directly (Outside Application).	
Default Button **(O)(H)**	PushButton	Not used.	Not applicable.
Detail Applet **(O)**	TextBox PushButton	The name of the pop-up applet that shows information from the related fields in the same business component.	

Property	Applies To	Description / Usage	Valid Values / Examples
Display Format **(O)**	TextBox	The format for data displayed by the control.	
Field **(O)**	TextBox CheckBox ImageControl	The business component field or applet message that the control is displaying data for.	
Field Retrieval Type **(O)**		Specifies that the field data is either rich content HTML data (Field Data), a standard reference to a web page (URL), or a specially constructed URL integrating data taken from the Siebel database and supporting the external content service (Symbolic URL). When Field Retrieval Type is NULL, the application adds a <SPAN> tag around the field value.	Valid values are Field Data, Symbolic URL, and URL. The HTML Attachment value is obsolete and is no longer a valid value for this property.
Field Type **(O)**		Indicates that this control is used to display data from either a business component field or an applet message. This value is automatically pre-defaulted to the correct type when the Field attribute is populated.	Valid values are Field and Message.
Foreground Color **(O)(H)**	See the description	Space-separated RGB value. Applies to Textbox, Label, PushButton, CheckBox, ComboBox, and Group.	
Group **(O)(H)**	All	A TRUE or FALSE value. Not used.	
Height - Language Override **(O)(H)**		Provides the ability to change the order of controls or list columns when a particular language/locale is active.	

Property	Applies To	Description / Usage	Valid Values / Examples
Height **(O)(H)**	All	The height of the control in pixels.	Valid values are numbers greater than or equal to 0. The default value is 10.
HTML Attributes **(O)**		Can be used to add HTML tag attributes to the HTML tags that the Standard Interactivity client creates to render the control.	Example: If you put in `size=30` for a text box, the following is generated: `<input type = text size=30 . . .>`
HTML Bitmap **(O)**		Used when the control invokes a method. References a Bitmap object within the HTML Control Icons bitmap category. If the HTML Disabled Bitmap property is not set, this image is used when the control is enabled as well as when it is disabled. If the HTML Disabled Bitmap property is set, then this image is used only when the control is enabled.	
HTML Default Control **(O)(H)**		Not used.	Not applicable.
HTML Disabled Bitmap **(O)**		Similar to the HTML Bitmap property. Used instead of the HTML Bitmap property when the control is in a disabled state.	
HTML Display Mode **(O)**		Used with controls that are field values. Controls how the field value is displayed. For Applet Messages, this should be set to FormatData. This allows the use of newlines and spaces in the Applet Message.	Valid values are EncodeData, DontEncodeData, and FormatData. For more information, read "HTML Display Mode" on page 152.

Property	Applies To	Description / Usage	Valid Values / Examples
HTML Height - Language Override **(O)**		Provides the ability to change the height, in pixels, of the control when a particular language/locale is active.	
HTML Height **(O)**	Java applet, ActiveX and input type controls.	Height of control in pixels. In general this property should not be changed manually. This property is automatically set when editing an applet layout in the Tools layout editor. When using a TextArea field that spans multiple fields, you may find the need to add 2 pixels for each row to account for the buffer between fields on a grid-based form.	Example: On the Service Request Detail Applet for the Description control, the TextArea field spans three rows of single-row fields. To keep the bottom of this field aligned with the other fields (such as Home Phone#), you must increase the HTML Height. The increase is an additional 2 pixels for every row of fields the TextArea field spans, not including the first row. In this case, Description spans three rows of fields, so the HTML Height property should be increased by (2 rows) x (2 pixels), or 4 pixels total. The original HTML Height of 72 pixels should be adjusted to 76 pixels.
HTML Icon Map **(O)**		Can be set to an Icon Map object. Used to map field values to icons as defined in the Icon Map.	
HTML Max Chars Displayed - Language Override **(O)**		Provides the ability to change the maximum number of characters displayed in a control when a particular language/ locale is active.	

Property	Applies To	Description / Usage	Valid Values / Examples
HTML Max Chars Displayed **(O)**		The maximum number of characters to be displayed in a control. Only applies when the parent applet is in base mode or the control is read only. Property does not limit the number of characters that can be entered into a text control in edit mode.	
HTML Only **(O)**		A TRUE or FALSE value to indicate whether the control is used only in the HTML thin client.	TRUE — The control is used only in the HTML thin client. FALSE — The control is also used in the dedicated client.
HTML Row Sensitive **(O)**		Causes the Web engine to position the applet on the current record before invoking the method defined in the Method Invoked property.	Examples of methods that require this to be set to TRUE are EditRecord, DeleteRecord, and Drilldown.
HTML Sequence - Language Override **(O)**		Provides the ability to change the HTML sequence property when a particular language/ locale is active.	
HTML Sequence **(O)**		The (order) position of this control relative to other controls in the applet in the HTML thin client. Indicates tab sequence.	
HTML Type **(O)**		Defines how the control is to be rendered. If this is not set, then the application uses the Type property. For Applet Messages, this should be set to Plain Text. This enables wrap-around behavior when the text is long.	
HTML Width - Language Override **(O)**		Provides the ability to change the HTML Width property when a particular language/locale is active.	

Property	Applies To	Description / Usage	Valid Values / Examples
HTML Width **(O)**	Java applet, ActiveX and input type controls.	The width of the control in pixels in the HTML thin client (maximum is 2048).	
Left - Language Override **(O)(H)**		Provides the ability to change the left coordinate of a control when a particular language/ locale is active.	
Left **(O)(H)**	All	The left coordinate of this control in pixels.	Valid values are numbers greater than or equal to 0.
Left Text **(O)(H)**	CheckBox	A TRUE or FALSE value.	TRUE — Puts the text on the left side of the control.
Method Invoked **(O)**	PushButton	The method invoked when the button control is clicked. This property is ignored if the control type is not PushButton. Methods available for a button control depend on the class of the applet as well as the class of the business component. The basic methods available for CSSFrameList and CSSFrame applet classes are NewRecord, DeleteRecord, CopyRecord, and UndoRecord. Use other methods with caution. For more information, read "MethodInvoked" on page 152 and *Siebel Developer's Reference*.	

Property	Applies To	Description / Usage	Valid Values / Examples
Multi Line **(O)(H)**	TextBox	A TRUE or FALSE value indicating a multiple-line TextBox control. If the Automatic Vertical Scroll style is TRUE, the control shows as many lines as possible and scrolls vertically when the user presses Enter. If Automatic Vertical Scroll is FALSE, the control shows as many lines as possible and beeps if Enter is pressed when no more lines can be displayed. If Automatic Horizontal Scroll style is TRUE, the multiple-line control automatically scrolls horizontally when the cursor goes past the right edge of the control. To start a new line, the user must press ENTER. If Automatic Horizontal Scroll is FALSE, the control automatically wraps words to the beginning of the next line when necessary; a new line is also started if ENTER is pressed. The position of the text is determined by the window size. If the window size changes, the text is redisplayed. Multiple-line TextBox controls can have scroll bars. A TextBox	The default is single line.
MVG Applet **(O)**	TextBox	The applet to use for the pop-up MVG if the field for the control is a multi-value field and the Runtime property is set to TRUE for the control.	
Name **(R)**	All	The name for the control.	

Property	Applies To	Description / Usage	Valid Values / Examples
Owner Draw **(O)(H)**	PushButton	A TRUE or FALSE value.	TRUE — Displays on the button a bitmap specified in the Caption property.
Pick Applet **(O)**	TextBox	The applet to use for the pop-up picklist if the field for the control has a picklist specified and the Runtime property is set to TRUE.	
Popup Edit **(O)(H)**	TextBox	A TRUE or FALSE value.	TRUE — Specifies that an ordinary text field requires a pop-up edit box.
Prompt **(O)**	TextBox	Not used.	Not applicable.
Prompt Text **(O)**		Not used.	Not applicable.
Read Only **(O)**	TextBox	A TRUE or FALSE value. If the underlying field has a property setting of Read Only = TRUE, the control is also read-only, even if the control's Read Only property is FALSE.	TRUE — Prevents the control from being edited.
Runtime **(O)**	TextBox	A TRUE or FALSE value.	TRUE — Makes a run-time check to see if a pop-up button is provided for a picklist, calculator, calendar, or MVG (any control with a down arrow or glyph). FALSE — The application assumes that there is no picklist or MVG or the field is Date or Time in DateTime.
Sequence - Language Override **(O)(H)**		Provides the ability to change the order of controls when a particular language/locale is active.	
Sequence **(O)(H)**	All	The (order) position of this control relative to other controls in the applet. Indicates tab sequence.	Valid values are numbers greater than or equal to 1.

Property	Applies To	Description / Usage	Valid Values / Examples
Show Popup **(O)**		A TRUE or FALSE value. This property should not be set to TRUE on a button if there is underlying script that uses the application object method, GotoView. If this property is set to TRUE and TheApplication.GotoView is used, this causes the view to be opened in a new browser window. The Siebel client UI does not support a Multiple Document Interface (MDI) architecture, so this combination is not supported.	TRUE — Specifies that a new browser window is opened before invoking the method. FALSE — Specifies that the method is invoked in the current browser window.
Sort **(O)**	ComboBox	A TRUE or FALSE value.	TRUE — Automatically sorts strings entered into the control.
Tab Stop **(O)(H)**	All	A TRUE or FALSE value.	TRUE — User can move to the next control specified by the Sequence property by using the TAB key.
Target View Frame **(O)**	HTML thin client	The HTML page loaded into the frame that is specified by this property. Assumes that the control has an invoke method and a user clicks on the control to invoke the method.	Valid values: Self — Frame where the applet control resides. Blank — Opens a new full feature window. Others — Name of any frame that is created by Siebel or by users.
Text Alignment - Language Override **(O)**		Provides the ability to change the alignment of the text in a control when a particular language/locale is active.	
Text Alignment **(O)**	TextBox Label	Indicates how to align the text in the control.	Valid values are Left, Center, or Right.

Property	Applies To	Description / Usage	Valid Values / Examples
Text Alignment-Label - Language Override **(O)**		Provides the ability to change the Text-Alignment Label property for an Applet Control when a particular language or locale is active. Note that this occurs automatically for locales that read right-to-left (such as Hebrew).	Center, Left, and Right.
Text Style **(O)(H)**	CheckBox, ComboBox, Group, Label, PushButton, and TextBox.	Specifies the text style to use when displaying this control. Text styles are defined in the Text Style object and specify the font information.	
Text-Alignment Label **(O)**		Used to align the text of a control's label on form applets.	Center, Left, and Right. Example: On a grid-based form applet, a field and its label are two separate Applet Web Template Items that are based in the same control. With labels placed to the left of a field, this requires the label text to be aligned to the Right while the field text is Left aligned.
Tooltip Text **(O)(H)**	Button	Provides tooltip text that appears when the pointer lingers on a toolbar button. Applies to bitmap buttons.	
Top - Language Override **(O)(H)**		Provides the ability to change the order of controls when a particular language/locale is active.	
Top **(O)(H)**	All	Sets the top coordinate of this control in pixels. The top coordinate of the applet is 0.	Valid values are numbers greater than or equal to 0.

Property	Applies To	Description / Usage	Valid Values / Examples
Type **(O)(H)**	All	Sets the control type.	Valid values are ChartControl, CheckBox, ComboBox, Group, Label, ListControl, PushButton, TextBox, and TreeControl.
Vertical Scroll **(O)(H)**	ComboBox TextBox	A TRUE or FALSE value.	TRUE — Control has a vertical scroll bar.
Visible - Language Override **(O)**		A TRUE or FALSE value. Provides the ability to change the visibility of controls when a particular language/locale is active.	TRUE — Control is visible when the language specified as the language override is used.
Visible **(O)**	All	A TRUE or FALSE value that specifies whether or not the control is visible.	TRUE — Control is visible.
Want Return **(O)(H)**	TextBox	A TRUE or FALSE value that specifies whether or not a carriage return is inserted when the user presses ENTER while entering text into a multiple-line TextBox control. This property has no effect on a single-line TextBox control.	TRUE — Inserts a carriage return. FALSE — Pressing ENTER has the same effect as clicking the applet's default pushbutton.
Width - Language Override **(O)(H)**		Provides the ability to change the order of controls when a particular language/locale is active.	
Width **(O)(H)**	All	The width of the control in pixels.	Valid values are numbers greater than or equal to 0. Default value is 40.

Applet Designer

In most cases, controls are created and modified using the Applet Designer.

Caption

For list applets, this value is used to create the sortable list header (rendered using the ListHeader property of the swe:control tag). This value can also be used as field prompts (rendered using the DisplayName property of the swe:control tag).

For controls that invoke methods, this value is used as the link name or button label (rendered using the FormattedHtml property of the swe:control tag).

HTML Display Mode

The valid values for this property are listed below:

- EncodeData — If the field value contains characters that are HTML reserved ones (<>, &, and so on), they are encoded before showing it so that they appear correctly. This is the default.
- DontEncodeData — Switches off EncodeData. Used only in special cases where the field value actually is a valid HTML text and wants to be shown as such. (This has nothing to do with any text that users enter in the template file.)
- FormatData — Used when description or comment fields are in read-only layout. For example, for a service request description field, the user may enter data that uses new lines and tabs to format the text. If FormatData is not set to TRUE, when this is displayed in HTML these formats are lost. The whole text appears in one line. Setting FormatData = TRUE causes the data to be formatted in HTML so that it looks the same as when entered.

MethodInvoked

The following represents which events MethodInvoked property of Control invokes in HTML clients:

In the High Interactivity (HI) mode, a button with a MethodName property triggers the following chain:

1. Fires an InvokeMethod on the browser-side applet class (running in the HTML browser).
2. You can attach a PreInvokeMethod handler to this to intercept the method. You do this by adding a “Browser Script” on the applet, to handle the PreInvokeMethod event.
3. If the InvokeMethod on the browser-side applet class is not handled, it performs an InvokeMethod on the server-side applet class. The client-side applet class always calls the server-side applet with this method, unless there is a specialized client-side applet that is handling the method on the client side itself.
4. You can implement a WebApplet_PreInvokeMethod script in either Siebel VB or Siebel eScript to handle the method here.
5. If the InvokeMethod on the server-side applet class is not handled, it forwards it to the server-side business component for that applet.
6. You can implement a BusComp_PreInvokeMethod script in either Siebel VB or Siebel eScript to handle the method here.
7. If the InvokeMethod on the server-side business component class is not handled, an error is returned.

In the Standard Interactivity mode, only Step 4 through Step 7 apply.

See Also
"Applet" on page 29
"Hidden Object Types and Properties" on page 21

For information about user properties, read *Siebel Developer's Reference.*

Control Locale

Siebel Object Types > Applet > Control > Control Locale

Represents language-specific overrides used with the Control object type.

Properties

Property	Description/Usage	Valid Values/Examples
Caption **(O)**	Used as headers for list applets columns, field prompts, and link names.	
Height **(O)(H)**	Specifies the height of the control in pixels.	
HTML Height **(O)**	Height of the control in pixels.	
HTML Max Chars Displayed **(O)**	Maximum number of characters to be displayed in a control.	
HTML Sequence **(O)**	The (order) position of this control relative to other controls in the applet in the HTML thin client. Indicates tab sequence.	
HTML Width **(O)**	The width of the control in pixels in the HTML thin client.	
Left **(O)(H)**	Specifies the left coordinate of this control in pixels.	Valid values are numbers greater then or equal to 0.
Name **(R)**	The abbreviation of the language being used by the application.	Example: ENU.
Prompt Text **(O)**	Not used.	Not applicable.
Sequence **(O)(H)**	Sets the (order) position of this control relative to other controls in the applet.	Valid values are numbers greater than or equal to 1.
Text Alignment **(O)**	Indicates how to align the text in the control.	Value is Left, Center, or Right.

Property	Description/Usage	Valid Values/Examples
Text-Alignment Label **(O)**	Provides the ability to change the Text-Alignment Label property when a particular language or locale is active. This is linked directly to the Text Alignment Label Language Override property for a control.	
Tooltip Text **(O)**	Provides tooltip text that appears when the pointer lingers on a toolbar button. Used with bitmap buttons.	
Top **(O)(H)**	Set the top coordinate of this control in pixels. The top coordinate of the applet is 0.	Valid values are numbers greater than or equal to 0.
Visible **(O)**	A TRUE or FALSE value. Specifies whether or not this control is visible.	TRUE — The control is visible.
Width **(O)(H)**	Sets the width of the control in pixels.	Valid values are numbers greater than or equal to 0.

See Also

"Control" on page 139
"Hidden Object Types and Properties" on page 21

Control User Prop

Siebel Object Types > Applet > Control > Control User Prop

Communicates a property value to specialized C++ code. The values of user properties can be changed at configuration time. These values persist in the repository and the Siebel repository file.

Properties

Property	Description/Usage	Valid Values/Examples
Name **(R)**	The name of the user property.	
Value **(R)**	The value of the user property.	

Changing Undocumented User Properties

Customer developers can change only those user properties that have been documented in this guide or in *Siebel Developer's Reference*. Only Oracle developers who create or change specialized C++ code should create or delete user properties that are not documented.

See Also

"Applet" on page 29
"Applet User Prop" on page 41
"Business Component User Prop" on page 101
"Control" on page 139
"Field User Prop" on page 186
"List Column User Prop" on page 230

For information about user properties, read *Siebel Developer's Reference.*

Data Source (H)

Siebel Object Types > Table > Data Source

The data source that the application Object Manager uses to connect to the external table. Define this object only for tables that do not reside within the Siebel database.

Properties

Property	Description/Usage	Valid Values/Examples
Name **(R)**	Name of the data source. The name of the data source should match the value entered in the application configuration file ([DataSources] section) or the Alias name of an "InfraDatasources" server component type.	

See Also

"Table" on page 292

DLL

Siebel Object Types > DLL

Contains the Siebel Tools program library files.

CAUTION: Do not modify this object type unless adding a new ActiveX control or Java Applet to the Web user interface. Any modification can adversely affect performance.

Properties

Property	Description / Usage
Attributes **(O)**	Any attributes that are associated with the applet tag can be defined in this field.
Code or Class Id **(O)**	This defines the complete classname of the Java applet and includes the package name.
File Name **(R)**	Name of the DLL file.
Java Package **(O)**	Specifies the Java packages required by this DLL.
Name **(R)**	Name of the DLL object, without the DLL filename extension.

Dock Object

Siebel Object Types > Dock Object

Represents rules or groups of rules for synchronizing the records of member tables to remote users.

Relates to information access. For more information, read *Siebel Developer's Reference*.

CAUTION: Do not modify this object type. Any modification can adversely affect performance and operation.

Properties

Property	Description / Usage
Active **(S)**	A TRUE or FALSE value. If TRUE, the dock object is active in the application.
Dock Code **(S)**	Code used for the dock object.
Name **(S)**	Abbreviated name of the primary table.
Primary Table Name **(S)**	The name of the driving table. All of the tables identified in dock object tables for a given dock object are related, through foreign keys in the data model, to one driving table.
User Name **(S)**	The full name of the dock object.
Visibility Level **(S)**	Specifies at a high level whether all records are to be transferred for a corresponding set of tables, or only a limited set.
Write DB cache **(S)**	A TRUE or FALSE value. If TRUE, the dock object writes explicitly to the S_DOCK_INST table.

See Also

"Dock Object Related DObj" on page 157
"Dock Object Table" on page 158
"Dock Object Visibility Rule" on page 159

Dock Object Related DObj

Siebel Object Types > Dock Object > Dock Object Related DObj

Relates to information access. For more information, read *Siebel Developer's Reference*.

CAUTION: Do not modify this object type. Any modification can adversely affect performance and operation.

Properties

Property	Description/Usage
Active **(S)**	A TRUE or FALSE value. If TRUE, the dock object is active in the application.
Check Visibility **(S)**	A TRUE or FALSE value. If TRUE, the visibility rules are checked for the user.
Name **(S)**	Name of dock object with related dock object.
Rel DObj Visibility Strength **(S)**	Integer value. Visibility strength of the related dock object.
Related Dock Object Name **(S)**	Name of the related dock object.
SQL Statement **(S)**	SQL script used to handle special exceptions for a certain rule.
Sequence **(S)**	Position of the dock object. Integer value.
Visibility Event Columns **(S)**	Name of the visibility event columns for the related dock object visibility rule.
Visibility Strength **(S)**	Visibility strength of the related dock object. Integer values between 0 and 100. A visibility strength of 100 means full visibility, while 0 means no visibility. Any value between 1 and 100 implies partial visibility.

See Also

"Dock Object" on page 156
"Dock Object Table" on page 158
"Dock Object Visibility Rule" on page 159

Dock Object Table

Used to specify the tables whose records are actually transferred in conjunction with the Dock Object object type.

Relates to information access. For more information, read *Siebel Developer's Reference*.

CAUTION: Do not modify this object type. Any modification can adversely affect performance and operation.

Properties

Property	Description/Usage
Dock Object Sub Type **(S)**	Sub type of the dock object.
Filter Sql Statement **(S)**	SQL script used to filter information regarding the dock object tables.
Name **(S)**	Name of table receiving the records.
Node Language Filter **(S)**	Used to allow language-based selective downloading on Translation Tables to the mobile clients based on the their preferred language(s). A TRUE or FALSE value. If TRUE, the node language filter applies to these translation tables. If set to FALSE, the node language filter is not applied, which means it is treated like a normal dock table.
Notify (S)	A TRUE or FALSE value. If TRUE, events are logged to the Siebel Remote Admin > Client Diagnostics screen.
Source Column Name **(S)**	The name of the column that is the source for the dock object table.
Table Name **(S)**	Name of the table transferred with the dock object.
Target Column Name **(S)**	Target column for the table transferring records.
Target Table Name **(S)**	Name of the table receiving records transferred with the dock object.
Visibility Event **(S)**	A TRUE or FALSE value. If TRUE, any changes on a particular dock table is a visibility event transaction. The Transaction Router performs visibility rules check on an visibility event.
Visibility Strength **(S)**	Visibility strength of the dock object table. Integer values between 0 and 100. A visibility strength of 100 means full visibility, while 0 means no visibility. Any value between 1 and 100 implies partial visibility.

See Also
"Dock Object" on page 156
"Dock Object Related DObj" on page 157
"Dock Object Visibility Rule" on page 159

Dock Object Visibility Rule

Siebel Object Types > Dock Object > Dock Object Visibility Rule

A set of rules that evaluates which logical records in a dock object display to a mobile user. This relates to information access. For more information, read *Siebel Developer's Reference*.

CAUTION: Do not modify this object type. Any modification can adversely affect performance and operation.

Properties

Property	Description / Usage
Active **(S)**	A TRUE or FALSE value. If TRUE, the dock object visibility rule is active in the application.
Category Column Name **(S)**	Name of the category column that decides whether a category is visible to a user.
Category Table Name **(S)**	Name of the category table that decides whether a category is visible to a user.
Check DObj Visibility Strength **(S)**	Property checks the visibility strength property of the dock object object type.
Check Dock Object Name **(S)**	Name of the object definition used to evaluate the records to see if it is related to another record that the user receives.
DBX SQL Statement **(S)**	SQL statement used for database extract.
Description **(S)**	Short description of the dock object visibility rule.
Dock Object Sub Type **(S)**	Sub type of the dock object.
Employee Column Name **(S)**	Name of the column that evaluates each record according to whether it has a foreign key to the mobile user's Employee record.
Employee Table Name **(S)**	Name of the employee table.
Name **(S)**	Name of dock object visibility rule.
Node Column Name **(S)**	Column name for the node dock object visibility rule.
Node Table Name **(S)**	Table name for the node dock object visibility rule.
Organization Column Name **(S)**	Column name for the organization dock object visibility rule.
Organization Table Name **(S)**	Table name for the organization dock object visibility rule.

Property	Description/Usage
Partial **(S)**	Not used.
Position Column Name **(S)**	Column name for the position dock object visibility rule.
Position Table Name **(S)**	Table name for the position dock object visibility rule.
Sequence **(S)**	Integer value that designates in what order the visibility rule is used.
Source Column Name **(S)**	Source column name for the check dock object visibility rule.
Source Table Name **(S)**	Source table name for the check dock object visibility rule.
SQL Statement **(S)**	SQL statement for SQL dock object visibility rule.
Target Column Name **(S)**	Target column name for the check dock object visibility rule.
Target Table Name **(S)**	Target table name for the check dock object visibility rule.
Type **(S)**	The type of the dock object visibility rule.
Visibility Event Columns **(S)**	Visibility event columns for the dock object visibility rule.
Visibility Strength **(S)**	Visibility strength of the dock object visibility rule. Integer values between 0 and 100. A visibility strength of 100 means full visibility, while 0 means no visibility. Any value between 1 and 100 implies partial visibility.

See Also
"Dock Object" on page 156
"Dock Object Related DObj" on page 157
"Dock Object Table" on page 158

Drilldown Object

Siebel Object Types > Applet > Drilldown Object

Provides the user with the ability to drill down on a field in a list applet and be taken to another view that displays more information about the field. Drilldown fields in a list view are identified by colored, underlined text, much like a hypertext link in a Web browser.

NOTE: You can configure Drilldown objects for list applets only.

Properties

Property	Description/Usage	Valid Values/ Examples
Business Component **(O)**	The business component that the user is drilling into (destination). If this property is left blank, then the destination business component is the same as the source business component (although the view typically changes). In a drilldown object, if the destination view should show the same active business components in the same context, and if you do not want to change the current query context, then do not specify a business component and do not specify the Source and Destination fields.	
Destination Field **(O)**	The field in the destination business component whose value equals the value of the source field in the source business component. If you do not specify a business component then do not specify the Destination field. If no value is specified, then it defaults to Id, which is the row ID of the destination business component.	
Hyperlink Field **(O)**	The field in the business component for the applet on which the drilldown occurs. This also determines which list column in the list applet is blue and underlined. If multiple drilldown objects for the applet are defined, a given field in the business component should be mentioned only once for all available drilldown objects. For a dynamic drilldown, the drilldown object that contains the dynamic drilldown destinations should have the Hyperlink Field property set.	Not required, but you should always enter a value—otherwise, no hyperlink appears for the user to drill down on.
Menu Text **(O)**	Not used.	Not applicable.
Name **(R)**	The name of the drilldown object. Must be unique within the scope of the applet.	
Sequence **(O)**	An integer value that determines the order of the items in the right-click shortcut menu. Also, the first item is the one that is used if the user double-clicks the row to drill down rather than single-clicking underlined text.	

Property	Description/Usage	Valid Values/ Examples
Source Field **(O)**	The field in the applet's business component (the source of the drilldown) whose value is applied as a search specification to the destination field in the destination business component (the business component that is specified by the Business Component property). Because the user is navigating from one view to another, there is a new active business component. The Source Field, Business Component, and Destination Field properties provide the linkage from a record in the source business component to an appropriate record or set of records in the destination business component. If you do not specify a business component then do not specify the Source field.	If this property is left blank, then the behavior is to drill down to the same row in the destination business component. In other words, the application applies a bookmark of the current business component to the destination.
View **(O)**	The view that appears when the user drills down. If the View property is left blank, you must specify a value in the Business Component property.	If this property is left blank, the current view is maintained. This is used when drilling down from accounts into subaccounts, for example.
Visibility Type **(O)**	A property that, when row ID-based drilldown is employed from an applet to a specific destination record, specifies a different visibility type (All, Sales Rep, Manager, or Personal) than the default visibility for the destination business component.	Generally a value of All is used if this feature is employed. This ensures that the destination record can be displayed.

See Also
"Applet" on page 29

Drilldown Object Locale

Siebel Object Types > Applet > Drilldown Object > Drilldown Object Locale

Represents language-specific overrides used with the Drilldown Object object type.

Properties

Property	Description/Usage	Valid Values/ Examples
Menu Text **(O)**	The text that appears in the right-click shortcut menu for a row for a specific language.	
Name **(R)**	The abbreviation of the language being used by the application.	Example: ENU.

See Also

"Drilldown Object" on page 160

Dynamic Candidate (H)

Siebel Object Types > Workflow Policy Object (H) > Assignment Object (H) > Dynamic Candidate (H)

Employee, position, and organization candidates for an assignment rule are dynamically specified from an attribute on the object.

Properties

Property	Description/Usage	Valid Values/ Examples
Attribute Id Column	Not used.	
Attribute Table	Not used.	
Display Name **(O)**	A display name for the Dynamic Candidate object.	
Name	A name for this object.	
Object Id Column	Not used.	
Score Column	Not used.	
Team Table	Not used.	
Team Table Attribute Id Column	Not used.	
Team Table Candidate Id Column	Not used.	
Team Type	Indicates whether the candidate team is Employee, Organization, or Position based.	

See Also

"Dynamic Candidate Component (H)" on page 164
"Dynamic Candidate Component Col (H)" on page 166
"Dynamic Candidate Locale (H)" on page 166

Dynamic Candidate Attribute (H)

Workflow Policy Object (H) > Assignment Object (H) > Dynamic Candidate (H) > Dynamic Candidate Attribute

NOTE: As of version 7.8, this object type is no longer used. Use "Dynamic Candidate Component (H)" on page 164 and "Dynamic Candidate Component Col (H)" on page 166 instead.

Properties

Property	Description/Usage	Valid Values/ Examples
Assignment Criterion	Not used.	
Team Attribute Column	Not used.	

Dynamic Candidate Component (H)

Siebel Object Types > Workflow Policy Object (H) > Assignment Object (H) > Dynamic Candidate (H) > Dynamic Candidate Component (H)

A list of joins from the assignment object to the candidate table.

Properties

Property	Description/Usage	Valid Values/ Examples
Additional Join Spec	Use this property to specify an additional join.	
Candidate Id Column	The column in the Team table containing the Id of the candidate (employee, position, or organization).	
Candidate Table	The name of the source table at the last level of a join. If candidate table is indicated, then Candidate Id Column property is required. **NOTE:** Before version 7.8, this property was called Team Table.	

Property	Description / Usage	Valid Values / Examples
Primary	Indicates where the join begins and is the primary table for the join. Typically, the primary is the table that the assignment object is stored in. For example, consider an account-contact-position dynamic candidate that has the S_ORG_EXT record marked as the primary. In this instance, Assignment Manager determines S_ORG_EXT is the first table in the join. If primary is not indicated, then Target Column Name and Target Component Name properties are required.	
Score Column	The column in the Team table containing scores of the candidates.	
Source Column Name **(R)**	The column in the source table that relates to another dynamic candidate component.	
Source Table Name **(R)**	The table that the dynamic candidate component is based on.	
Target Column Name	The column in the target component that relates to the column in the source column of the source table for this record. When the sql code is generated, the following occurs: inner join source table on source table.source column = (source table from target component).target column.	
Target Component Name	The target dynamic candidate component that joins to the source table in this dynamic candidate component.	

See Also

"Dynamic Candidate (H)" on page 163
"Dynamic Candidate Component Col (H)" on page 166
"Dynamic Candidate Locale (H)" on page 166

Dynamic Candidate Component Col (H)

Siebel Object Types > Workflow Policy Object (H) > Assignment Object (H) > Dynamic Candidate (H) > Dynamic Candidate Component (H) > Dynamic Candidate Component Col (H)

Maps columns in a source table, at a particular level in a join, to a team-based assignment criterion.

Properties

Property	Description/Usage	Valid Values/ Examples
Assignment Criteria Attribute Name	The name of the assignment criteria attribute that is compared to the value in the Attribute Column Name.	
Assignment Criteria Name	The name of the assignment criteria. **NOTE:** This property is automatically populated when you select the assignment criteria attribute name.	
Attribute Column Name **(R)**	The name of the column in the source table.	
Copy Column Name	The name of the column stamped on the team table when working in operational mode.	
Reporting Copy Column Name	The name of the column stamped on the Name reporting table when working in reporting mode.	

See Also

"Dynamic Candidate (H)" on page 163
"Dynamic Candidate Component (H)" on page 164
"Dynamic Candidate Locale (H)" on page 166

Dynamic Candidate Locale (H)

Siebel Object Types > Workflow Policy Object (H) > Assignment Object (H) > Dynamic Candidate (H) > Dynamic Candidate Locale

Represents language-specific overrides used with the Dynamic Candidate object type.

Properties

Property	Description/Usage	Valid Values/ Examples
Display Name **(O)**	A display name for the Dynamic Candidate Locale object.	

See Also
"Dynamic Candidate (H)" on page 163
"Dynamic Candidate Component (H)" on page 164
"Dynamic Candidate Component Col (H)" on page 166

Dynamic Drilldown Destination

Siebel Object Types > Applet > Drilldown Object > Dynamic Drilldown Destination

Configures dynamic drilldown behavior for a hyperlink field (and the corresponding list column or control).

Properties

Property	Description / Usage	Valid Values / Examples
Destination Drilldown Object **(O)**	The name of the drilldown object in the same applet that the hyperlink routes to when the field specified in the Field property matches the value specified in the Value property.	
Field **(O)**	The Type field in the business component of the applet, to be checked in the current record for the presence of a value that matches the setting in the Value property of the dynamic drilldown destination object definition. When the value matches, the hyperlink routes to the drilldown object specified in the Destination Drilldown Object property.	
Name **(R)**	Uniquely identifies each dynamic drilldown destination object definition within the parent drilldown object.	
Sequence **(O)**	Sets the search order for the children of one drilldown object, based on the integer values in this property from lowest to highest.	
Value **(O)**	The value to be matched in the Type field.	

Differences Between Static and Dynamic Drilldowns

As in a static drilldown configuration, the drilldown object definition identifies a hyperlink field and a view. These property settings continue to have the same purpose in dynamic drilldown, namely, to specify the list column or control that has hyperlink capabilities and the destination view when the hyperlink is clicked.

However, in dynamic drilldown, the drilldown object also has dynamic drilldown destination object definitions, each of which points to the type field in the business component and specifies a value to look for there. When the value in a dynamic drilldown destination is matched, the logic routes to a different drilldown object (typically with a different destination view).

See Also
"Applet" on page 29
"Drilldown Object" on page 160

EIM Explicit Primary Mapping

Siebel Object Types > EIM Interface Table > EIM Table Mapping > EIM Explicit Primary Mapping

Used by the Siebel Enterprise Integration Manager (EIM). For more information, read *Siebel Developer's Reference*.

CAUTION: Do not modify this object type. Any modification can adversely affect performance and operation.

Properties

Property	Description/Usage
Inf Tbl Exp Primary Flg Col Name (S)	Name of the column used for the primary mapping.
Name (S)	Name of the table used for the primary mapping.

EIM Interface Table

Siebel Object Types > EIM Interface Table

The EIM Interface Table Object type is an alternative representation of the Table object type, for tables of type Interface only.

Used by the Siebel Enterprise Integration Manager (EIM). For more information, read *Siebel Developer's Reference*.

CAUTION: Do not modify this object type. Any modification can adversely affect performance and operation.

Properties

Property	Description/Usage
Abbreviations 1 - 3 **(S)**	Specifies up to three alternative abbreviations for the table.
Alias **(S)**	Not used.
Base Table Name **(S)**	If the Type property is Ext or Journal, the name of the table that is extending or journaling.
EIM Delete Proc Column Name **(S)**	Name of the column being deleted during data integration.

Property	Description / Usage
EIM Export Proc Column Name **(S)**	Name of the column being exported during data integration.
EIM Merge Proc Column Name **(S)**	Name of the column being merged during data integration.
File **(S)**	A TRUE or FALSE value. TRUE indicates that the table is used to store file attachment information.
Name **(S)**	The unique name of the table.
Parent Interface Table Name **(S)**	Name of the parent interface table.
Parent Table Column 1 Name **(S)**	The name of the column that points to the primary parent if this is a child or intersection table.
Parent Table Column 2 Name **(S)**	The name of the column that points to the second parent if this is an intersection table.
Target Table Name **(S)**	Name of the target table.
Type **(S)**	A description of what the table is used for.
User Name **(S)**	The “user-friendly” name for the table.

See Also

“Table” on page 292

EIM Interface Table Column

Siebel Object Types > EIM Interface Table > EIM Interface Table Column

The EIM Interface Table Column object type is an alternative representation of the Column object type, for columns that are child object definitions of interface tables.

Used by the Siebel Enterprise Integration Manager (EIM). For more information, read *Siebel Developer’s Reference*.

CAUTION: Do not modify this object type. Any modification can adversely affect performance and operation.

Properties

Property	Description / Usage
Alias **(S)**	Not used.
Cascade Clear **(S)**	A TRUE or FALSE value. TRUE indicates to EIM how to handle this foreign key on delete of the related entity. Leave FALSE for extension columns.
Default **(S)**	The value used for this column in new records.
Denormalization Path **(S)**	The path that is being denormalized if this is a denormalization column. Applies to denormalization of user key columns into intersection tables.
EIM Processing Column Flag **(S)**	A TRUE or FALSE value. TRUE indicates that this interface table's column is reserved for EIM processing. This flag should not be used for customer data.
Force Case **(S)**	Indicates if new values should be forced to uppercase letters.
Foreign Key Table Name **(S)**	Specifies the table to which this column is a foreign key, used by EIM.
Lov Bounded **(S)**	A TRUE or FALSE value. If LOV Bounded is TRUE, EIM checks the values as it imports against the values contained in a list defined in LOV Type. In that case, LOV data should be imported first into S_LST_OF_VAL, and LOV Type must be specified. This property is read-only for standard columns in Siebel applications, but is editable for columns that have been added by way of the Database Extension Designer.
Lov Type **(S)**	Specifies the list of values domain in which this column is validated. Used in conjunction with the LOV Bounded property. List of values domains are defined in List of Values Administration in the client. This property is read-only for standard columns in Siebel applications but is editable for columns that have been added by way of the Database Extension Designer.
Name **(S)**	Specifies the name, which must be unique within the table.
Nullable **(S)**	A TRUE or FALSE value. TRUE indicates that NULL can be stored in this column by the database.
Physical Type **(S)**	Identifies the data type of the column in the database.
Precision **(S)**	The total number of digits in a number column. For non-integer columns, the precision is 22. For integer columns, the precision is 10.

Property	Description / Usage
Primary Child Col **(S)**	Used by EIM. For Oracle use only.
Primary Child Column Name **(S)**	The name of the column that contains the primary ID value.
Primary Child Table Name **(S)**	The name of the table from which the primary child is to be found.
Primary Inter Table Name **(S)**	Used by EIM. For Oracle use only.
Primary Join Column Name **(S)**	The name of the column on which to join to the base record.
Primary Key **(S)**	A TRUE or FALSE value. If TRUE, consider this column part of the primary key of the table.
Required **(S)**	A TRUE or FALSE value. If TRUE, the application requires this column. For example, even if the database allows NULLs, the application does not.
Scale **(S)**	The number of decimal digits in a number column, out of the total number of digits (precision). For non-integer columns, the scale is 7. For integer columns, the scale is 0.
Sequence Object **(S)**	A TRUE or FALSE value. TRUE indicates that a Sequence Object is specified, in which case column values are auto-generated. For Oracle use only. For each table, there should be only one foreign key for the foreign key table.
Text Length **(S)**	The length in characters for Text and Varchar columns.
Translation Table Name **(S)**	The name of the translation table.
Type **(S)**	Describes what this column is used for.
Use FKey **(S)**	A TRUE or FALSE value. TRUE indicates that this foreign key is to be used by synchronization in evaluating visibility. Leave FALSE in extension columns.
Use Key Sequence **(S)**	The sequence in the user key where this column fits.
User Name **(S)**	The “user-friendly” name for the column.
Valid Condition **(S)**	The expression for validating column values.

See Also

“Column” on page 130

EIM Table Mapping

Siebel Object Types > EIM Interface Table > EIM Table Mapping

Identifies a data table that is updated by the parent EIM interface table object definition.

Used by the Siebel Enterprise Integration Manager (EIM). For more information, read *Siebel Developer's Reference*.

CAUTION: Do not modify this object type. Any modification can adversely affect performance and operation.

Properties

Property	Description/Usage
Default Map Column Name Prefix (S)	The prefix for the default column being mapped during data integration.
Destination Table Name (S)	Name of the target table for the mapping.
EIM Exists Proc Column Name (S)	Name of the existing column being used during data integration.
EIM ROW_ID Proc Column Name (S)	Name of the row ID column being used during data integration.
EIM Status Proc Column Name (S)	Name of the status column being used during data integration.
EIM Unique Proc Column Name (S)	The unique name of the column being used during data integration.
Explicit Mapping (S)	A TRUE or FALSE value. If TRUE, the table is explicitly mapped during data integration.
Name **(S)**	Specifies the name of the target table.

See Also

"Attribute Mapping" on page 85
"Foreign Key Mapping" on page 191

Entity

Siebel Object Types > Entity Relationship Diagram > Entity

This object contains the definition of an entity's properties.

Properties

Property	Description/Usage	Valid Values/Examples
Business Component	A user-defined entity or an entity associated with an existing Siebel business component.	Business component can be associated to the entity through a drop-down list.
Name	Name of the entity.	Default Name, for example Entity 1. Can be renamed by user.

See Also

"Entity Attribute (H)" on page 173
"Entity Physical Relation (H)" on page 174
"Entity Relation" on page 175
"Entity Relationship Diagram" on page 175

Entity Attribute (H)

Siebel Object Types > Entity Relationship Diagram > Entity > Entity Attribute

Contains information on the properties of an attribute in the entity.

Properties

Property	Description/Usage	Valid Values/Examples
Business Component Field	Contains a pick list of the available properties on the business component that is mapped to the entity.	For example, if the Account BC is mapped to an entity, the Business Component Field pick list contains Account Competitors, Account Conditions, Account Markets, and so on.
Data Example	Free form text example of a needed attribute.	
Data Type	The attribute's data type.	This is a drop-down list that contains Character, Date, Date Time, Long, Number, Time, UTC Date Time, and Varchar.
Format String	The attribute string's format.	For example, if the data is a telephone number you enter "(XXX)XXX-XXXX".
Length	The attribute's maximum length.	
Name	The Entity Attribute's name.	

Property	Description/Usage	Valid Values/Examples
Pick List Domain	The set of legal values of an attribute that is defined from a pick list.	
Pick List Required	Check box indicating attribute is a bounded list.	
User Key Flag	Indicates if this attribute is a user key or not.	

See Also

"Entity" on page 172
"Entity Physical Relation (H)" on page 174
"Entity Relation" on page 175
"Entity Relationship Diagram" on page 175

Entity Physical Relation (H)

Siebel Object Types > Entity Relationship Diagram > Entity Relation > Entity Physical Relation

When you bind a relation in the ERD these fields become populated as read-only fields.

Properties

Property	Description/Usage	Valid Values/Examples
Business Component	This is the primary business component.	This is the BC name if the relation is a link and the parent BC if the relation is a join.
Join Alias	If the relation is a join, this is the join alias.	
Link Name	If the relation is a link, this is the link name.	
Name	The name of the repository object that is bound to the relation.	The name is the link name if the record represents a link. The name is the join alias if the record represents a join.

See Also

"Entity" on page 172
"Entity Attribute (H)" on page 173
"Entity Relation" on page 175
"Entity Relationship Diagram" on page 175

Entity Relation

Siebel Object Types > Entity Relationship Diagram > Entity Relation

Contains the definition of the Relation shown on the Canvas. You can define multiplicities, provide a name for the relation as well as specify a name for both ends of the relation.

Properties

Property	Description/Usage	Valid Values/Examples
Cardinality	Indicates the multiplicity of the relationship.	This drop-down list contains the values: 1:1 (one to one), 1:M (one to Many), M:1 (many to one), and M:M (many to many).
End Name 1	You can define a name for the start point of the relationship.	
End Name 2	You can define a name for the target end of the relationship.	
Entity 1	The entity at the start point of the relationship.	Can be either a user-defined entity or one that represents a Siebel business component.
Entity 2	The entity at the target end of the relationship.	Can be either a user-defined entity or one that represents a Siebel business component.
Name	The name of the relation.	The name can be the default name such as Relation0 or user defined such as “Owns” for Customer Owns Automobile.

See Also

“Entity” on page 172
“Entity Attribute (H)” on page 173
“Entity Physical Relation (H)” on page 174
“Entity Relationship Diagram” on page 175

Entity Relationship Diagram

Siebel Object Types > Entity Relationship Diagram

The Entity Relationship Designer is a tool for business analysts and developers to define a customer's business entities (independent of the Siebel data model) and then map the appropriate Siebel objects to these entities.

Properties

Property	Description/Usage	Valid Values/Examples
Name	The name of the Entity Relationship Diagram (ERD). Must be unique.	
Status	Indicates the status of the ERD relative to its development cycle	Approved Scope, Implementation Reviewed, Implemented, Production, Scope, To Be Implemented

See Also

"Entity" on page 172
"Entity Attribute (H)" on page 173
"Entity Physical Relation (H)" on page 174
"Entity Relation" on page 175

External Search Engine (H)

Siebel Object Types > External Search Engine (H)

Customizes full content-search from Siebel applications. Search collections from Hummingbird, Ltd. can be modeled. A search engine consists of one or more search tables that consist of one or more search fields.

Properties

Property	Description/Usage	Valid Values/Examples
Max Rows **(O)**	The maximum number of result rows to retrieve.	
Max Time **(O)**	The maximum time (seconds) allowed to do a search.	
Name **(R)**	The logical name of this search definition. This is specified as the value for the SearchDefName property in the CFG file.	
Sort Specification **(O)**	The sort expression used to order the records returned.	Examples: SCORE DESC, FILENAME ASC.

See Also
"Hidden Object Types and Properties" on page 21

Field

Siebel Object Types > Business Component > Field

Identifies and defines a field in a business component. All fields making up a business component record contain entries from both Single Value Field and Multi Value Field object types.

You can add user properties to field objects.

Properties

Property	Description/Usage	Valid Values/Examples
Calculated **(O)**	A TRUE or FALSE value.	TRUE — The field's value is calculated by the business component rather than retrieved from the database server.
Calculated Value **(O)**	The expression used to calculate the field's value if Calculated is TRUE.	
Column **(R+)**	The name of the database table's column. References to the field are done through its name. (+) This property is not used for calculated fields, fields of virtual business components, or MVFs, but is required for other fields.	The column name is case-sensitive and must be specified the same as it is in the database.
Currency Code Field **(O)**	The name of the currency code field. Used with a data type of DTYPE_CURRENCY.	The default field name is Currency Code.
Dest Field **(O)**	If the Multi Valued property is TRUE, the field in the destination business component defined in the Multi Value Link field. If Multi Valued is FALSE, this property is ignored.	
Exchange Date Field **(O)**	The name of the exchange date field; used with a data type of DTYPE_CURRENCY.	The default field name is Exchange Date.
Force Active **(O)**	A TRUE or FALSE value. For more information, read "Force Active" on page 183.	TRUE — The data value is always retrieved from the database.

Property	Description/Usage	Valid Values/Examples
Force Case **(O)**	The case for the field value.	Valid values are UPPER, LOWER, or FIRSTUPPER.
Hidden **(O)**	A TRUE or FALSE value.	TRUE — Makes the field invisible to all dynamically created list views, such as those used in the forecasting module. This property is used to support specialized functionality and is not intended for general configuration. For a business component data type of DTYPE_ID, the default is TRUE; otherwise, the default is FALSE.
Immediate Post Changes **(O)**	A TRUE or FALSE value. Field data is posted to the server when the focus moves off of the field and then the data is refreshed. Causes an immediate roundtrip to the server. When set to True the browser script PreSetFieldValue event is bypassed. Typically used for constrained drop-down lists and calculated fields. Excessive use affects performance.	TRUE — Changes to field object definitions are posted as made.
Internal Pick Applet **(O)**	Used by the workflow user interface. For the Decision Point step, you can set conditions on the business component fields. If the business component field on the UI normally has an applet, you can set this business component field attribute to make the workflow UI use that applet when displaying conditions for the field.	
Join **(O)**	The name of the join object definition used by this field. Used to retrieve data from the table defined by a join. Also refer to "Join" on page 212.	This property's value corresponds to the join's Name property.

Property	Description/Usage	Valid Values/Examples
Link Specification **(O)**	A TRUE or FALSE value. You must specify as TRUE if any business component used as the child of a link from this business component uses the Parent type of default values and expects to get a value.	TRUE — Specifies that the field's value is passed as a default value to a field in the child business component.
Multi Value Link **(O)**	If the Multi Valued property is TRUE, specifies the multi-value link for the field. If the field is not multi-valued, this property is ignored.	
Multi Valued **(O)**	A TRUE or FALSE value.	TRUE — The field is multi-valued. FALSE — The field is single-valued.
Name **(R)**	The user-defined name for the field. All references to the field are done through its name. Do not change the name of any Oracle-provided fields. Doing so can create upgrade problems and may affect intended behavior.	The name can contain spaces and must be unique within the business component.
No Copy **(O)**	A TRUE or FALSE value.	TRUE — Specifies that during a Based On Last operation, the field's value is not copied into the newly created record. If No Copy is set to FALSE in fields from a joined table, ensure that No Copy in the join field (identified in the Source Field property in the join specification) is also set to FALSE. Otherwise, in the user's Based On Last operation, the new record appears to copy the fields from the joined table, but when the record is required the fields are not populated.

Property	Description/Usage	Valid Values/Examples
Oracle Sequence Object **(O)**	The name of an Oracle sequence object. The sequence is used for field defaults when creating a new record. The Oracle sequence object has to be created manually in the Oracle Database. This is an exception to the normal way the Oracle sequence object is configured. Oracle sequence object is not supported in Siebel Remote Replication Environment.	
PickList **(O)**	The name of a picklist object definition used to display a list of valid values from which the user can choose.	
Post Default Value **(O)**	The value used for an empty field when the record is initially written to the database. This value is not used if the field is left empty on subsequent updates to the record. The post-default value for the field is used even when the field is not exposed in the user interface. This property is not validated for a calculated field. The Link Specification property needs to be set to TRUE on the Parent business component for the parent syntax to work in the child business component's linked field.	Limited to 255 characters.

Property	Description/Usage	Valid Values/Examples
Pre Default Value **(O)**	The value used for an empty field when a new record is created through an Add New Record or Copy Record operation. The value can be changed by the user before the record is written to the database. The pre-default value for the field is used even when the field is not exposed in the user interface. **NOTE:** When predefault value expressions depend on other fields, those fields must be populated when the record is initialized. For example, Pre Default Value of Expr: Field '[Id]' works because the Row Id or Id field is always populated during record initialization. However, the Pre Default Value on Expr: Field: '[Picklist Field]' does not work because the Picklist Field is populated *after* the record is initialized. This is true for Copy Record operation only if the NoCopy property is set to TRUE. This property is not validated for a calculated field.	Limited to 255 characters.
Precision **(O)**	The maximum number of digits, to the left and right of the decimal point, for a numeric field.	The default is 16. The maximum value available is database dependent.

Property	Description/Usage	Valid Values/Examples
Read Only **(O)**	A TRUE or FALSE value. In general, fields that are based on a joined table are always read-only. There are two exceptions to this rule—joins to extension tables and joins to active intersection tables. Certain fields are read-only even if the Read Only property is FALSE. The Effective Start Date field for the following business components is always read-only: ■ Account Product ■ Promotion ■ Promotion Account ■ Promotion Account Product ■ Promotion Administration ■ Promotion Product See also "Join" on page 212.	TRUE — Prevents the field value from being changed by the user.
Required **(O)**	A TRUE or FALSE value.	TRUE — Requires a value to be entered into the field before the record can be written to the database.
Scale **(O)**	The maximum number of digits to the right of the decimal point for a numeric field.	Valid values are 6 for numeric values or 0 for integers. The maximum value available is database dependent.
Text Length **(O)**	The maximum length of the text in the field. This property is usually ignored and the length retrieved from the underlying column definition. Note: Text Length is not ignored if a Text field has a user property named Text Length Override.	
Type **(R)**	The field's data type. For more information, read "Field Data Types" on page 184.	

Property	Description/Usage	Valid Values/Examples
Use Default Sensitivity **(O)**	A TRUE or FALSE value. For fields of type DTYPE_ID, queries are case-sensitive if Use Default Sensitivity is TRUE and the CaseInsensitive setting in the CFG file is FALSE. Queries are case-insensitive for this data type if Use Default Sensitivity is TRUE and the CaseInsensitive setting is TRUE. However, for Id fields not exposed in the business component (such as the ID field in Opportunity), the User Default Security property cannot be set.	TRUE — Causes the sensitivity mode that the application is running in (as defined in the data source) to be used in QBE searches that do not explicitly specify which sensitivity mode to use.
Validation **(O)**	The expression used to ensure data correctness when data is entered. Also, the validation is evaluated only for the field for which it is added. Should be used only with single value fields, not MVFs.	Limited to 255 characters.

Force Active

Default is FALSE. When Force Active is TRUE, the field is queried each time the business component is instantiated, even when the field is not exposed on the user interface. Force Active should only be set to TRUE when there is a need to script a field in the active business component and the field is not exposed on the active applet. Setting this property to TRUE may reduce performance.

For information about user properties, read *Siebel Developer's Reference.* For information about virtual business components, read *Siebel Developer's Reference* or *Overview: Siebel Enterprise Application Integration*.

Virtual Business Components

The following properties of Field objects are supported in virtual business components:

- Calculated
- Calculated Value
- Currency Code Field
- Exchange Date Field
- Force Case
- Hidden

- Internal Pick Applet
- Pick List
- Precision
- Read Only
- Required
- Scale
- Text Length
- Type

The following properties of Field objects are not supported in virtual business components:

- Column
- Dest Field
- Force Active
- Join
- Link Specification
- Multi Value Link
- Multi Valued
- No Copy
- Oracle Sequence Object
- Post Default Value
- Pre Default Value
- Use Default Sensitivity
- Validation

Field Data Types

Siebel Object Types > Business Component > Field

The Type property of a Field object type definition corresponds to the physical type of the column in the underlying table.

Notes:

- All values in the Type property have a prefix of DTYPE_.
- Standard Siebel applications come with predefined data types, which you must use.
- When mapping columns to a data type, make sure to use a similar type. For example, map an integer column to DTYPE_INTEGER. Do not map to incompatible types, such as mapping a Long column to DTYPE_BOOL.

The following table lists data types for the Field object type.

Field Type	Definition
DTYPE_BOOL	■ Data stored as Y or N ■ Often displayed as TRUE/FALSE or checked/unchecked ■ Mapped to column with physical type/length of Character(1)
DTYPE_CURRENCY	■ Refers to data as currency ■ Mapped to column with physical type of Number
DTYPE_DATE	■ Refers to data as a date; ignores additional stored data such as time ■ Mapped to column with physical type of Date
DTYPE_DATETIME	■ Refers to date and time ■ Field must be mapped to column with physical type of DateTime
DTYPE_ID	■ Application-generated key ■ Mapped to column with physical type/length of Varchar(15)
DTYPE_INTEGER	■ Refers to data as an integer ■ Mapped to column with physical type of Number
DTYPE_NOTE	■ Refers to data as a long string ■ Mapped to column with physical type of Long
DTYPE_NUMBER	■ Refers to the data as a number ■ Mapped to column with physical type of Number
DTYPE_PHONE	■ Refers to data as a phone number ■ Mapped to column with physical type/length of Varchar(40)
DTYPE_TEXT	■ Refers to the data as a string of length 2000 or less ■ Defaults to mixed-case as entered by the user ■ The ForceCase property on a Field can be used to force text to Upper, Lower, or FirstUpper ■ Mapped to column with physical type of Varchar
DTYPE_TIME	■ Refers to the data as a time ■ Additional stored information such as date is ignored ■ Mapped to column with physical type of Time

See Also
“Business Component” on page 91
“Multi Value Field” on page 234
“Field Locale” on page 186
“Field User Prop” on page 186

Field Locale

Siebel Object Types > Business Component > Field > Field Locale

Represents language-specific overrides used with the Field object type.

Properties

Property	Description/Usage	Valid Values/Examples
Validation Message **(R)**	The language-specific text that should be displayed when validation fails. The Validation property is configured on the parent field.	An error message that should be displayed to the user in the language specified in the Language Code property.

See Also
“Business Component” on page 91
“Field” on page 177

Field User Prop

Siebel Object Types > Business Component > Field > Field User Prop

Communicates a property value to specialized C++ code. The values of user properties can be changed at configuration time. These values persist in the repository and the Siebel repository file.

Properties

Property	Description/Usage	Valid Values/Examples
Name **(R)**	The name of the user property.	
Value **(O)**	The value of the user property.	

Changing Undocumented User Properties

Siebel Tools developers should use and modify only documented user properties. User properties are subject to change in future Siebel releases, and as such, only documented user properties are supported.

See Also

"Business Component" on page 91
"Field" on page 177

For information about user properties, read *Siebel Developer's Reference*.

Find

Siebel Object Types > Find

Creates an entry that can be added to the Find dialog box for applications.

Properties

Property	Description/Usage	Valid Values/ Examples
Applet **(R)**	The applet to be used by the find. This applet must be included in the view specified for every Find View object definition for the Find object.	
Destination Field **(R)**	The destination field in the business component (of the applet) that is used to drill down into a particular record. If nothing is specified, the row ID field is used.	
Display Name **(O)**	A display name for the Find object.	
Drilldown View **(O)**	The view that appears when the user drills down. If the drilldown is not specified, the Find View is used.	
Name **(R)**	The name of the Find object definition. This is referenced when setting up application find object definitions that use this Find object definition.	
Preview **(O)**	The view that appears (pops up in a separate browser window) when a user invokes the Preview button in the Search Center. The Preview view is usually configured to be a read-only view.	Example: For the Find Object Accounts, the Preview attribute is the Accounts Preview View.

See Also

"Find Field" on page 188
"Find View" on page 191

Find Field

Adds a search field to the dynamically generated Find dialog box.

Properties

Property	Description / Usage	Valid Values / Examples
Display Name **(O)**	The text (or key to localized text) that displays for this find field when its parent Find object definition is active in the Find dialog box.	
Drilldown Business Component	Information not available.	
Drilldown Destination Field	Information not available.	
Drilldown Source Field	Information not available.	
Drilldown View	Information not available.	
Field **(R)**	The field being searched in the business component.	
Name **(R)**	Name of the search field added.	
Sequence - Language Override **(O)**	Provides the ability to change the sequence property when a particular language/locale is active.	Integer value.
Sequence **(O)**	The position of this find field in the list of fields displayed in the Find dialog box for the parent find object definition.	Integer value.
Title **(O)**	Reserved for future use.	Not applicable.

Field Restriction

The field specified in a find field must be a field that exists in the business component for the applet for the find object definition.

Query Differences

The Find dialog box and a query by example (QBE) can have different behavior for multi-value fields because the Find dialog box generates the same query as if you did an EXISTS query on the field in QBE. You can verify this by embedding your QBE expression with Exists (); for example, `Exists ("Aga*")`.

QBE supports both types of querying, but to keep the Find dialog box simple, Siebel applications implement a query that works in all cases. If you want to mimic the behavior of the non-Exists version with the Find dialog box, create a join from the primary ID field for the MVLink to the table containing the data. Then add a field on the table you joined to and point your find field to this new field.

See Also

"Find" on page 187
"Find View" on page 191

Find Field Locale

Siebel Object Types > Find > Find Field > Find Field Locale

Represents language-specific overrides used with the Find Field object type.

Properties

Property	Description/Usage	Valid Values/ Examples
Display Name **(O)**	The text (or key to localized text) that displays for this find field when its parent find object definition is active in the Find dialog box for a specific language.	
Name **(R)**	The abbreviation of the language being used by the application.	Example: ENU.
Sequence **(O)**	The position of this find field in the list of fields displayed in the Find dialog box for the parent find object definition for a specific language.	Integer value.

See Also

"Find Field" on page 188

Find Locale

Siebel Object Types > Find > Find Locale

Represents the language-specific overrides used with the Find object type.

Properties

Property	Description/Usage	Valid Values/ Examples
Display Name **(O)**	A display name for the Find object for a specific language.	
Name **(R)**	The abbreviation of the language being used by the application.	Example: ENU.

See Also

"Find" on page 187

Find Pick View

Siebel Object Types > Find > Find Pick View

Identifies the applet in the Pick View that can receive selected items from the Find results list.

Properties

Property	Description/Usage	Valid Values/ Examples
Applet Name **(O)**	Specifies the name of an applet in the view that has an associated applet that allows users to associate child records to the parent.	
Field Name **(O)**	Specifies the MVG field. (Not supported in Siebel 7.)	
Name **(R)**	Property is hidden and its value is defaulted from the View Name property.	
View Name **(O)**	Specifies the name of the view.	

Find results can be associated to an applet as long as there is a relationship between the applet and its parent, and the search result and the Pick Applet use the same business components. In Siebel Tools, this relationship is implemented by creating a link between the two Business Components with an intersection table.

An example of the Pick View and Pick Applet for the Find category Opportunities would be View: Contact Detail—Opportunities View and Applet: Opportunity List Applet—Basic. An Opportunity Find result can be attached to the Opportunity List Applet—Basic, as long as there is a many-to-many relationship between the parent Contact applet and the child applet, which is Opportunity List Applet—Basic.

See Also

"Applet" on page 29

Find View

Siebel Object Types > Find > Find View

Specifies the view that is presented when performing a find operation with the Find dialog box.

Properties

Property	Description/Usage	Valid Values/ Examples
Sequence **(O)**	The order in which views are to be considered for usage for the find object definition. For more information, read "Visibility Considerations" on page 191.	
View **(R)**	The name of the view to be used for the find operation when this find view is used.	

Visibility Considerations

For a given user, the find operation uses the first find view object definition that is in the user's responsibility list. Because different views apply different visibility, you would want to have the least restrictive views first in the list and the most restrictive last. This order enables users with different responsibilities to find items using the most broad visibility that they are allowed to use.

See Also

"Find" on page 187
"Find Field" on page 188

Foreign Key Mapping

Siebel Object Types > EIM Interface Table > EIM Table Mapping > Foreign Key Mapping

Each Foreign Key Mapping object definition identifies a foreign key column in the destination table that is to be populated from the interface table.

Used by the Siebel Enterprise Integration Manager (EIM). For more information, read *Siebel Developer's Reference*.

CAUTION: Do not modify this object type. Any modification can adversely affect performance and operation.

Properties

Property	Description / Usage
EIM Foreign Key Proc Column **(S)**	Name of the column being used as the foreign key by EIM.
Foreign Key Column Name **(S)**	Name of the column being used as the foreign key.
Name **(S)**	Name of the foreign key column.
Override Ref Column **(S)**	Name of the column used as a possible override for the EIM Foreign Proc Column property.
User Key Name **(S)**	Name of the user key column that is found in the user key index.

See Also

"EIM Interface Table" on page 168
"EIM Table Mapping" on page 172

Foreign Key Mapping Column

Siebel Object Types > EIM Interface Table > EIM Table Mapping > Foreign Key Mapping > Foreign Key Mapping Column

Each Foreign Key Mapping Column object definition identifies a piece of the user key; that is, one of the attribute columns used to locate rows in the table the foreign key points to. The user key columns, taken together, uniquely identify rows in that table.

Used by the Siebel Enterprise Integration Manager (EIM). For more information, read *Siebel Developer's Reference*.

CAUTION: Do not modify this object type. Any modification can adversely affect performance and operation.

Properties

Property	Description / Usage
Interface Table Data Column Name **(S)**	The name of the column in the interface table that supplies the data.
Name **(S)**	The name of the user key column.
Self Ref Mapping Column Name **(S)**	Name of the column.
Self Reference Join **(S)**	A TRUE or FALSE value. If TRUE, a join operation is allowed between the self reference mapping column and the foreign key mapping column.
User Key Attribute Name **(S)**	Name of the attribute column.

See Also
"EIM Interface Table" on page 168
"EIM Table Mapping" on page 172
"Foreign Key Mapping" on page 191

Help Id

Siebel Object Types > Help Id

Specifies the HTML file name that contains the help topics for a screen in a Siebel application.

A Help Id object definition may be referenced by entering its name as the property text in the Help Identifier property of a view or screen object definition. The Help Id object definition, in turn, references a help topic file (HTML format) associated with the application through the HTML Help URL property.

When the user chooses Help > Contents from the application-level menu to invoke context-sensitive online help, the application calls the Siebel Web Engine (SWE) GotoPage method, which uses SWE code to display the correct help topic in a separate browser window.

For more information about the help implementation in Siebel applications, read *Siebel Object Interfaces Reference*.

Properties

Property	Description/Usage	Valid Values/Examples
HTML Help URL (O)	Specifies the URL to display when the help documentation for a particular Help Id is requested. The URL is relative to the public directory on the Siebel Web server installation.	Example: help/siebaccounts.htm
Name **(R)**	The string that identifies the help topic in Siebel Tools.	All caps, with no punctuation characters other than the underscore (_) symbol. The context string (Name) is in the format ID_*type*_*objdefname*, where *type* is VIEW or SCREEN, and *objdefname* identifies the specific view or screen. Example: ID_SCREEN_ACCOUNTS for the Accounts screen help.
Type **(O)**	The object type that can use this help topic.	Examples: View, Report.
Value **(O)**	Not used.	Not applicable.

See Also
"Screen" on page 256
"View" on page 325

HTML Hierarchy Bitmap

Siebel Object Types > HTML Hierarchy Bitmap

The HTML Hierarchy Bitmap object is a top-level object used to capture the set of bitmaps to be used for rendering hierarchical information.

For all the properties in the HTML Hierarchy Bitmap object type, the Bitmap object must belong to the HTML Hierarchy Icons Bitmap Category.

Properties

Property	Description / Usage	Valid Values / Examples
Arrow Down Bitmap **(O)**	Specifies the name of the Bitmap object that defines the attributes of a down arrow image.	
Arrow Up Bitmap **(O)**	Specifies the name of the Bitmap object that defines the attributes of an up arrow image.	
Bar Bitmap **(O)**	Specifies the name of the Bitmap object that defines the attributes of a vertical bar image.	
Close Bitmap **(O)**	Specifies the name of the Bitmap object that defines the attributes of a closed folder image.	
Collapse Bitmap **(O)**	Specifies the name of the Bitmap object that defines the attributes of a collapsed bitmap image.	
Collapse Elbow Bitmap **(O)**	Specifies the name of the Bitmap object that defines the attributes of a collapsed elbow bitmap image.	
Collapse Tee Bitmap **(O)**	Specifies the name of the Bitmap object that defines the attributes of a collapsed tee bitmap image.	
Elbow Bitmap **(O)**	Specifies the name of the Bitmap object that defines the attributes of an elbow bitmap image.	
Expand Bitmap **(O)**	Specifies the name of the Bitmap object that defines the attributes of an expanded bitmap image.	
Expand Elbow Bitmap **(O)**	Specifies the name of the Bitmap object that defines the attributes of an expanded elbow bitmap image.	
Expand Tee Bitmap **(O)**	Specifies the name of the Bitmap object that defines the attributes of an expanded tee bitmap image.	
Leaf Bitmap **(O)**	Specifies the name of the Bitmap object that defines the attributes of a document image.	

Property	Description/Usage	Valid Values/ Examples
Name **(R)**	Defines the name of this object.	
Open Bitmap **(O)**	Specifies the name of the Bitmap object that defines the attributes of an open folder image.	
Space Bitmap **(O)**	Specifies the name of the Bitmap object that defines the attributes of a white space image.	
Tee Bitmap **(O)**	Specifies the name of the Bitmap object that defines the attributes of a "|-" image.	

See Also

"List" on page 220
"Tree" on page 318

Icon

Siebel Object Types > Icon Map > Icon

An icon uses a bitmap object to define the image for the icon.

Object type is used by the Siebel Web Engine.

Properties

Property	Description/Usage	Valid Values/Examples
Bitmap **(O)**	Bitmap to be used for the icon.	
Bitmap Category **(O)**	Bitmap category to which the icon bitmap belongs.	
Name **(R)**	Name of the icon that is compared to the field value.	

See Also

"Icon Map" on page 195

Icon Map

Siebel Object Types > Icon Map

Defines a collection of named icon objects that are used in controls and list columns to map a field value to an icon. The field value is compared with the icon name to select the icon to be used to display the field value.

Object type is used by the Siebel Web Engine.

Properties

Property	Description/Usage	Valid Values/Examples
Name **(R)**	Name of the icon map.	

See Also

"Icon" on page 195

Import Aux Field

Siebel Object Types > Import Object > Import Aux Field

Defines the auxiliary key field name for the business component defined in the import object.

Properties

Property	Description/Usage	Valid Values/Examples
Aux Field Name **(R)**	The auxiliary key field name.	
Name **(R)**	The name within the import object.	

Duplicate Keys

If there is more than one duplicate based on the key field, a second search with auxiliary key fields (for example, Account and Account Location for Contact) is done.

See Also

"Import Object" on page 199

Import Field

Siebel Object Types > Import Object > Import Field

Defines the field that the data is imported to.

Properties

Property	Description/Usage	Valid Values/ Examples
Display Name **(O)**	Either the value to use as the column heading or the key to the international message table.	
Field **(R)**	The name of the field to which the data is imported.	
Name **(R)**	The name within the import object.	

See Also

"Import Object" on page 199

Import Field Column

Siebel Object Types > Import Object > Import Source > Import Field Map > Import Field Column

Defines the column in the import source from which the data is imported. The data is imported into the parent import field of the corresponding data map.

Properties

Property	Description/Usage	Valid Values/ Examples
Column **(R)**	The name of the column.	
Name **(R)**	The name within the import object.	

See Also

"Import Field Map" on page 198
"Import Object" on page 199
"Import Source" on page 199

Import Field Locale

Siebel Object Types > Import Object > Import Field > Import Field Locale

Represents language-specific overrides used with the Import Field object type.

Properties

Property	Description/Usage	Valid Values/ Examples
Display Name **(O)**	Either the value to use as the column heading or the key to the international message table for a specific language.	
Name **(R)**	The abbreviation of the language being used by the application.	Example: ENU.

See Also

"Import Field" on page 196

Import Field Map

Siebel Object Types > Import Object > Import Source > Import Field Map

Defines the mapping between the field in the business component and the column in the import source.

Properties

Property	Description/Usage	Valid Values/Examples
Country Code Column **(O)**	The name of the country code column in the import source.	
Display Name **(O)**	Either the value to use as the column heading or the key to the international message table.	
Field **(R)**	The name of the field to which the data is imported.	
Name **(R)**	The name within the import source.	
Parse Middle Name **(O)**	A TRUE or FALSE value.	TRUE — Parses the middle name out of the full name.
Separator Character **(O)**	The character inserted between the parts of multiple-input data.	

See Also

"Import Field" on page 196
"Import Object" on page 199
"Import Source" on page 199

Import Key Field

Siebel Object Types > Import Object > Import Key Field

Defines the key field name for the business component defined in the import object. Key fields (for example, First Name, Middle Name, and Last Name for the contact import object) are used to search for duplicates.

Properties

Property	Description / Usage	Valid Values / Examples
Key Field **(R)**	The key field name.	
Name **(R)**	The name within the import object.	

See Also

"Import Object" on page 199

Import Object

Siebel Object Types > Import Object

Defines the information for importing external data into the specified business component.

Properties

Property	Description / Usage	Valid Values / Examples
Business Component **(R)**	The name of the business component into which the data is imported.	
Name **(R)**	The name of the import object.	

Contact Imports

Siebel applications support contact business components only for client-side imports.

Import Source

Siebel Object Types > Import Object > Import Source

Defines the source from which data is imported: for example, ACT! and ECCO.

Properties

Property	Description/Usage	Valid Values/Examples
Name **(R)**	The name of the import source.	

See Also

"Import Object" on page 199

Index

Siebel Object Types > Table > Index

Creates a new index to improve performance of queries.

Properties

Property	Description/Usage	Valid Values/Examples
Alias **(O)**	The short name for the index.	
Allow Reverse Scan **(O)**	A TRUE or FALSE value. Applies to the DB2/UDB platform only.	TRUE — Indicates that the index is built with a backward page reference in addition to a forward page reference allowing the ability to read the index backwards.
Cluster **(O)**	A TRUE or FALSE value. Applies to the DB2/UDB platform only.	TRUE — The table is clustered on this index (for example, the rows are physically stored in the order of the index columns).
Cluster 2 **(O)**	A TRUE or FALSE value. This property applies to the MSSql platform and is applicable to unique indexes only.	TRUE — All other indexes have this cluster key appended to them.
Name **(R)**	The unique name.	

Property	Description/Usage	Valid Values/Examples
Number of Unique Columns **(S)**	If this value is populated then this equals the number of columns that participate in the uniqueness constraint. The columns that do not participate in the uniqueness constraint have the Include Only property set to TRUE. In such a case, the number of columns in the index is more than this (Number of Unique Columns) value. Applies to the DB2/UDB platform and is applicable to unique indexes only.	
Type **(O)**	A description of the index.	Valid values are Primary Key, User Key, Extension, and System.
Unique **(O)**	A TRUE or FALSE value. A unique index prevents multiple rows with the same values in each of the index columns.	TRUE — Indicates that this index is unique
User Name **(R)**	The user-friendly name for the index.	
User Primary Key **(O)**	A TRUE or FALSE value. Property is used by the Siebel Enterprise Integration Manager (EIM).	TRUE — EIM enforces uniqueness on this index.

Performance

Use caution when creating new indexes, as they can easily decrease performance as well as take up additional disk space.

See Also

"Table" on page 292
"Column" on page 130
"Data Source (H)" on page 155
"Index Column" on page 201
"User Key" on page 322

Index Column

Siebel Object Types > Table > Index > Index Column

Creates a new index column.

Properties

Property	Description/Usage	Valid Values/ Examples
Column Name **(R)**	The name of the column in the index.	
Name **(R)**	The column name within the index.	
Sequence **(R)**	The position of the column within the index.	
Sort Order **(O)**	Represents the way the column of an index is sorted. The approach to populate the 7.0 repository is as follows: 1 Set as NOT NULL with Asc as default. 2 If the index column is of Date type, set to Desc.	

See Also

"Index" on page 200
"Table" on page 292

Integration Component

Siebel Object Types > Integration Object > Integration Component

Structures an inbound or outbound message string into a format that is recognizable to both the Siebel application and an external application. The integration component is an intermediate structure that helps resolve differences between external application data formats and Siebel data formats.

Properties

Property	Description/Usage	Valid Values/Examples
Adapter Info **(S)**	Used by Oracle for synchronization by Integration Object Wizard.	
Business Component **(O)**	Name of the business component.	
Cardinality **(O)**	Number of integration components.	Valid values are One, One or More, Zero or More and Zero or One.
External Name **(S)**	The name of the structure that is recognizable to the external application.	
External Sequence **(O)**	Specifies the order in which integration components are processed.	Integer value.

Property	Description/Usage	Valid Values/Examples
Name **(R)**	The name of the integration component.	
Parent Integration Component **(O)**	The name of this component's parent integration component, if any. Only one active integration component within an integration object can be the root component.	If this component is the root component, this should be left blank.
XML Container Element **(O)**	XML Tag used to enclose elements that represent this integration component.	
XML Sequence **(O)**	Specifies the order in which integration components are presented in an XML document.	Integer value.
XML Tag **(O)**	The XML tag name that is generated for this component when the component is converted into an XML document.	

See Also

"Integration Component Field" on page 203
"Integration Object" on page 209

Integration Component Field

Siebel Object Types > Integration Object > Integration Component > Integration Component Field

Field of an Integration Component object.

Properties

Property	Description/Usage	Valid Values/Examples
Adapter Info **(S)**	Used by Oracle for synchronization by Integration Object Wizard.	
Business Component Field **(O)**	Name of the business component field.	
Column **(O)**	Name of the column.	
External Data Type **(R)**	The data type of the field from the external application.	

Property	Description/Usage	Valid Values/Examples
External Length **(O)**	The length of the external data element that corresponds to the Integration component field.	
External Name **(O)**	The external name of a data component or element that is recognizable to an external application.	
External Precision **(O)**	The number of decimal places, if any, supported by the external data element, if it is defined as a number.	
External Required **(O)**	A TRUE or FALSE value.	TRUE — The external data element is required by an external application. For example, a ROWID could be required by an external application that uses ROWID as a primary key field in a non-Siebel database.
External Scale **(O)**	The total length of the number field, including the decimal places, for the external data element.	Example: If the scale of a number is 6 and the precision 2, the field can represent a number of the format 9999.99.
External Sequence **(O)**	Influences the order of particular adapter processes' integration component fields. Semantics differ from adapter to adapter.	

Property	Description/Usage	Valid Values/Examples
Field Type **(O)**	The field type.	Maximum length is 30 characters: ■ Data — Contains raw field data that can be formatted in a user-defined fashion. XML documents are formatted as delimited strings and stored in this type of field. ■ System — Contains data formatted in the Siebel internal format. Used to specify primary rows of an MVG. Also used to specify an operation for an integration object instance.
Group **(O)**	Not used.	Not applicable.
Length **(O)**	The length of the Siebel field that corresponds to a Siebel table column.	
Name **(R)**	The name of the field.	Maximum length is 75 characters.
Physical Data Type **(O)**	Data type of the integration field in the integration space.	
Precision **(O)**	Number of decimal places, if any, supported by a given field, if it is defined as a number.	
Required **(O)**	Indicates that the integration component is required in an integration object.	
Scale **(O)**	The total length of the number field, including the decimal places.	Example: If the scale of a number is 6 and the precision 2, the field can represent a number of the format 9999.99.
User Visible **(O)**	A TRUE or FALSE value.	TRUE — Indicates that the integration component is visible to DTE users.

Property	Description/Usage	Valid Values/Examples
XML Literal Value **(O)**	Specifies default value that is presented in an XML document if the value for this integration component field is not specified.	
XML Parent Element **(O)**	The name of the parent element to which this component belongs. The parent element becomes an XML tag.	
XML Sequence **(O)**	Specifies the order in which integration component fields are presented in an XML document.	Integer value.
XML Style **(O)**	The name of the XML style. Indicates whether this field is formatted as an XML attribute or an XML element when an XML document is generated for the integration object that contains this field.	Values are Attribute or Element.
XML Tag **(O)**	A specific XML element that represents this integration component field. If not designated, a system-generated XML tag is assigned to the integration component field when output to an external application.	Maximum length is 75 characters.

See Also
"Integration Component" on page 202
"Integration Object" on page 209

Integration Component Field User Prop

Siebel Object Types > Integration Object > Integration Component > Integration Component Field > Integration Component Field User Prop

Communicates a property value to C++ code that implements special integration component field behavior. The value of these properties can be changed at configuration time. These values persist in the repository and in the Siebel repository file.

Properties

Property	Description/Usage	Valid Values/ Examples
Name **(R)**	Name of the integration component field user prop.	
Value **(O)**	Value of the integration component field user prop.	

Changing Undocumented User Properties

Customer developers can change only those user properties that have been documented. Only Oracle developers who create or change specialized C++ code should create or delete user properties that are not documented.

See Also

"Integration Component" on page 202
"Integration Component Field" on page 203
"Integration Object" on page 209

For information about user properties, read *Siebel Developer's Reference.*

Integration Component Key

Siebel Object Types > Integration Object > Integration Component > Integration Component Key

Defines a key for an integration component. A key has one or more key fields (defined in the Integration Component Key Field type) which are used to detect duplicate rows.

Properties

Property	Description/Usage	Valid Values/Examples
Key Sequence Number **(O)**	Specifies the order in which integration component keys of the same type are processed.	Integer value.
Key Type **(R)**	Type of integration component key.	Valid values are Foreign Key, Hierarchy Parent Key, Hierarchy Root Key, Status Key, Target Key, and User Key.

Property	Description/Usage	Valid Values/Examples
Name **(R)**	Name of this integration component key.	Maximum length is 75 characters.
Target Key Name **(O)**	Used only for keys of type Foreign Key. References the target key that this foreign key is pointing to. Fields are mapped by sequence.	

See Also

"Integration Component" on page 202
"Integration Component Field" on page 203
"Integration Component Field User Prop" on page 206
"Integration Component Key Field" on page 208
"Integration Component User Prop" on page 209
"Integration Object" on page 209

Integration Component Key Field

Siebel Object Types > Integration Object > Integration Component > Integration Component Key > Integration Component Key Field

Defines an integration field as a key field for a key within an integration object. The Integration Component Key Field object is implemented as an intersection table between integration fields and integration component keys.

Properties

Property	Description/Usage	Valid Values/ Examples
Field Name **(R)**	Integration component field referenced by its name.	
Name **(R)**	Name of the integration component key field.	
Sequence **(O)**	Specifies the order in which integration component key fields are processed.	Integer value.

See Also

"Integration Component" on page 202
"Integration Component Field" on page 203
"Integration Component Field User Prop" on page 206
"Integration Component Key" on page 207
"Integration Component User Prop" on page 209
"Integration Object" on page 209
"Integration Object User Prop" on page 210

Integration Component User Prop

Siebel Object Types > Integration Object > Integration Component > Integration Component User Prop

Communicates a property value to C ++ code that implements special integration component behavior. The value of these properties can be changed at configuration time. These values persist in the repository and in the Siebel repository file.

Properties

Property	Description / Usage	Valid Values / Examples
Name **(R)**	Name of the integration component user prop.	
Value **(O)**	Value of the integration component user prop.	

Changing Undocumented User Properties

Customer developers can change only those user properties that have been documented. Only Oracle developers who create or change specialized C++ code should create or delete user properties that are not documented.

See Also

"Integration Component" on page 202
"Integration Component Field" on page 203
"Integration Component Field User Prop" on page 206
"Integration Component Key" on page 207
"Integration Component Key Field" on page 208
"Integration Object" on page 209

For information about user properties, read *Siebel Developer's Reference.*

Integration Object

Siebel Object Types > Integration Object

Provides a logical representation of external application data, or of Siebel data that needs to map to an external application.

Properties

Property	Description/Usage	Valid Values/Examples
Adapter Info **(O)**	Used by Oracle for synchronization by Integration Object Wizard.	
Base Object Type **(R)**	One of several predefined object types, upon which this integration object is based.	Valid values are None, SAP BAPI Input, SAP BAPI Output, SAP IDOC, SQL, Siebel Business Object, and Table.
Business Object **(O)**	Name of the business object.	
External Major Version **(O)**	Not used.	Not applicable.
External Minor Object **(O)**	Not used.	Not applicable.
External Name **(R)**	The name of the object that an external function can recognize.	
Name **(R)**	The name of the integration object.	
XML Tag **(R)**	The XML tag name that is generated for this object when the object is converted to an XML document.	

See Also

"Business Object" on page 102
"Integration Component" on page 202
"Integration Component Field" on page 203

Integration Object User Prop

Siebel Object Types > Integration Object > Integration Object User Prop

Communicates a property value to C ++ code that implements special integration object behavior. The value of these properties can be changed at configuration time. These values persist in the repository and in the Siebel repository file.

Properties

Property	Description / Usage	Valid Values / Examples
Name **(R)**	Name of the integration object user prop.	
Value **(O)**	Value of the integration object user prop.	

Changing Undocumented User Properties

Customer developers can change only those user properties that have been documented. Only Oracle developers who create or change specialized C++ code should create or delete user properties that are not documented.

See Also

"Integration Component" on page 202
"Integration Component Field"
"Integration Component Field User Prop" on page 206
"Integration Component Key" on page 207
"Integration Component Key Field" on page 208
"Integration Component User Prop" on page 209
"Integration Object" on page 209

For information about user properties, read *Siebel Developer's Reference.*

Interface Table User Key Usage

Siebel Object Types > EIM Interface Table > Interface Table User Key Usage

Provides support for alternative user keys for base tables. An interface table user key usage object definition defines the use of a nontraditional user key for a given base table in a specific interface table.

Used by the Siebel Enterprise Integration Manager (EIM). For more information, read *Siebel Developer's Reference*.

CAUTION: Do not modify this object type. Any modification can adversely affect performance and operation.

Properties

Property	Description / Usage	Valid Values / Examples
Application Table Name **(S)**	Name of the base table in an interface table.	

Property	Description/Usage	Valid Values/ Examples
Name **(S)**	Name of the user key.	
User Key Name **(S)**	Name of the user key.	

Join

Siebel Object Types > Business Component > Join

Defines the logical join between a business component's base table and another table.

Properties

Property	Description/Usage	Valid Values/Examples
Name **(R)**	The name of the join object definition. This defaults to the table specified in the Table property. If there are two joins to the same table, then you need to change the name of one of them.	
Outer Join Flag **(O)**	A TRUE or FALSE value.	TRUE — Indicates a left outer join. FALSE — Indicates an inner join.
Table **(R)**	The name of an SQL table. The name is case sensitive and must be specified the same as it is in the Siebel database.	

SQL Restrictions

Refer to an SQL language reference manual for a complete explanation of and a list of restrictions on left, self, and simple joins.

Left Outer Joins

In defining a left outer join, the outer join is applied to the table defined by the join's Table property. Brings values back even if no values are returned for the join.

Fields on Joined Tables

Implicitly joined fields (unlike their explicit counterparts) can be updated.

See Also
"Business Component" on page 91
"Join Specification" on page 213

Join Constraint

Siebel Object Types > Business Component > Join > Join Constraint

A join constraint is a constant-valued search specification applied to a column during a join. It is for use with outer joins.

Properties

Property	Description/Usage	Valid Values/ Examples
Destination Column **(O)**	Column of the destination table to apply the constraint to.	Example: LANG_CD.
Name **(R)**	Name of the join constraint.	
Value **(O)**	A non-null constant value or constant function, such as LoginId(), PostionId(), and DivisionId(). GetProfileAttr can be used, but be aware that profile attributes can be persistent or dynamic. If dynamic, this property must contain a value before the join fires. If the property does not contain a value, then GetProfileAttr returns a null and the join fails.	Example: Language().

See Also
"Join" on page 212

Join Specification

Siebel Object Types > Business Component > Join > Join Specification

Specifies the columns to join from the tables on the left and right sides of the join.

Properties

Property	Description/Usage	Valid Values/ Examples
Destination Column **(O+)**	The name of the column in the destination table on which the join is performed. The name is case-sensitive and must be specified the same as it is in the database. (+) This property is required if the join occurs on a column other than ROW_ID.	
Name **(R)**	The name of the join specification object definition, which must be unique within the join. The name defaults to the Source Field value for the join specification. For more information, read "Join Specification Naming" on page 214.	
Source Field **(O+)**	The name of a field (in the parent business component of the join object definition) that the destination table is joined on. The Source Field property must point to a field that represents a database column. A computed field is an example of a field not based on a database column, and is therefore not suitable as a source field for a join specification. The field has to be a foreign key with respect to Destination Column. (+)This is required if the join occurs on a field other than ID.	

Join Specification Naming

The Name property of this object type requires some special consideration. By default, the name of a join specification is the name of the source field specified for the object definition. You should not change this value unless you have two join specification object definitions for the same join that use the same source field. Although this name is not referenced anywhere else in the repository, it is used for the merge/upgrade utility to determine matching entries.

See Also

"Business Component" on page 91
"Join" on page 212

Link

Specifies the relationship between two business components. The relationship can be one-to-many (1:M), such as parent/child, or many-to-many (M:M).

Properties

Property	Description/Usage	Valid Values/ Examples
Cascade Delete **(O)**	Indicates what action to perform on detail business component records of the link if the master is deleted. Do not set to DELETE if the child business component in this link is also a child business component in another link. In this case, use CLEAR instead. Also for DELETE, you have to check to see if the Table referenced by the child business component is used by another business component that is the child business component in another link. For example, assume you have two links: Contact/Action and Contact/Activity Plan. Action and Activity Plan are based on the same table. You have to make sure the setting of Cascade Delete is Clear for both links. Cascade Delete must be set to NONE for many-to-many links. With a many-to-many link, Siebel applications delete the intersection record but leave the child record intact as it may have many other parents. When you delete a record that is pointed to by foreign keys of other tables, the references to it may or may not be deleted. If those references are not deleted, the user is left with row IDs that point to nonexistent records. In the case of multi-value groups, sometimes these foreign keys are converted to “No Match Row Id”. In other words, if multiple links exist that refer to the same tables, you must make sure that the Cascade Delete property is consistent among all of those links.	Valid values: CLEAR — The foreign key reference is removed. DELETE — The child records are deleted. NONE — No operations are performed on the child record.
Child Business Component **(R)**	The name of the business component whose data is determined by the business component defined by the Parent Business Component property.	

Property	Description / Usage	Valid Values / Examples
Destination Field **(R+)**	The name of a child business component's field that is used to logically join to the parent business component. (+) Required for 1:M links.	
Inter Child Column **(R+)**	The name of the column of the database intersection table that links the intersection table to the child business component. The name is case-sensitive and must be specified the same as it is in the database. (+) Required for M:M links.	
Inter Child Delete **(O)**	A TRUE or FALSE value.	TRUE — Deletes both the association and the child record when deleted through the child record's applet. FALSE — Only the child record's association is deleted. Specified in M:M links only.
Inter Parent Column **(R+)**	The name of the column of the database intersection table that links the intersection table to the parent business component. The name is case-sensitive and must be specified the same as it is in the database. (+) Required for M:M links.	
Inter Table **(R+)**	The name of the database intersection table. The name is case-sensitive and must be specified the same as it is in the database. (+) Required for M:M links.	
Name **(R)**	A name for the link. The name can contain spaces and must be unique among links. All references to the link are done through its name. In general, this property is internally managed and should not be changed unless necessary.	

Property	Description / Usage	Valid Values / Examples
No Associate **(O)**	A TRUE or FALSE value. If TRUE, the association applet is prevented from appearing when creating a New record for the child in a M:M relationship. Accordingly, the user is forced to enter a new record and is not given the option of associating an existing one. Specified for M:M links only.	TRUE — Specifies that no new associations can be created through the link. New detail records may be added. This property is ignored if the current view is in Admin Mode.
No Delete **(O)**	A TRUE or FALSE value.	TRUE — Prevents deletion of child records through the link.
No Insert **(O)**	A TRUE or FALSE value. Specified for 1:M links only. See also the No Associate property for this object type.	TRUE — Prevents creation of child records through the link.
No Inter Delete **(O)**	A TRUE or FALSE value. Applies only to M:M relationships.	TRUE — Associated intersection table records are never deleted. FALSE (default) — Deletes the associated intersection table records when either a parent or a child record is deleted.
No Update **(O)**	A TRUE or FALSE value.	TRUE — Updating is not allowed.
Parent Business Component **(R)**	The name of the business component whose data determines what records, if any, are retrieved by the child business component (defined by the Child Business Component property).	
Primary Id Field **(O)**	Specifies the name of the field in the master business component that holds the row ID values pointing to primary records in the detail business component.	
Search Specification **(O)**	A conditional expression used to restrict the records retrieved. Applied to the child business component when the link is active.	

Property	Description / Usage	Valid Values / Examples
Sort Spec **(O)**	A sorting expression used in an association list to order the detail records retrieved. Specified for M:M links only.	
Source Field **(O)**	The name of a parent business component's field that is used to link to the child business component. The source field is a unique identifier for rows in the parent business component. Typically it is the field that maps to the ROW_ID column in the parent business component's base table and that has the name Id. This is not a field that appears in the list of fields in the Object List Editor for the business component. The Id field appears only in drop-down lists for properties such as Source Field in object types like Link. The default value for Source Field is Id, which means that a blank value for this property is the same as selecting Id.	
Visibility Auto All **(O)**	A TRUE or FALSE value.	TRUE — If the user has access to any "All" views, then visibility for child business component is All; otherwise visibility is determined by the Visibility Type property.

Property	Description/Usage	Valid Values/ Examples
Visibility Rule Applied **(O)**	A value that indicates how visibility rules should be applied when in a view using this link. Visibility settings in a view apply to the applet specified in the Visibility Applet property. If there is a master-detail relationship between this applet and an applet displaying detail records, then use the Visibility Rule Applied property to allow users to see detail records that they would not see because of visibility rules. Users are not allowed to drill down to see the complete records.	Valid values are: ■ Always — Allows visibility rules in the detail records when the current master-detail view is based on this link, even though the view does not have active visibility settings in the Visibility Applet and Visibility Applet Type properties. ■ Never — Disables visibility rules in the detail records when the current view is based on this link.
Visibility Type **(O)**	Determines the visibility type when the business component appears as a child in the given link. For more information, read "Visibility Auto All (O)" on page 219.	

Naming

When naming a link, the Siebel application convention is to use the parent business component and child business component names separated by a slash (/).

Calendar Business Component

The Calendar business component should not be used as the master or detail business component in a link.

List

Siebel Object Types > Applet > List

Holds some information specific to list applets. For a list applet, there must be one and only one list object definition. If the applet is not a list applet, there should be no list object definitions.

Properties

Property	Description / Usage	Valid Values / Examples
Alpha Tab Search Field **(O)(H)**	Name of the alpha tab search field. The field specified is the field in the business component for the applet that the alpha search should be applied to. If specified, causes the list applet to have an alphabet tab search bar.	
Alpha Tab Search Field - Language Override **(O)(H)**	Provides the ability to change the alpha tab search field property when a particular language/locale is active.	
HTML Hierarchy Bitmap **(O)**	Name of the HTML hierarchy bitmap. For more information, read "HTML Hierarchy Bitmap" on page 194.	
HTML Multi Row Edit **(O)**	A TRUE or FALSE value. Used by the Siebel Web Engine.	TRUE — While in the Edit List mode of the applet, you can edit multiple rows without saving each row. FALSE — Each row needs to be saved to the database before editing another row. FALSE is the default.
HTML Multi Row Select **(O)**	A TRUE or FALSE value. Used by the Siebel Web Engine.	TRUE — Supports marking multiple rows in the applet for invoking a method on. FALSE — Methods get invoked only on one row (the currently active one). FALSE is the default.
Name **(R)**	Specifies the name within the applet.	Should be List.
Total Displayed **(O)**	A TRUE or FALSE value. For the Siebel application to display the total when Total Displayed is TRUE, it must scan the entire data set each time the user navigates to or refreshes the view. This can adversely affect performance.	TRUE — Causes the display of running totals immediately upon entry into the applet. FALSE — Causes the totals to be calculated but not displayed (and are therefore available for manipulation by a script or other calculated fields).

Property	Description / Usage	Valid Values / Examples
Total Required **(O)**	A TRUE or FALSE value. This property is relevant only if the applet is a list applet.	TRUE — Sums and displays those columns whose Total Required property is TRUE at the bottom of the list.
Variable Row Height **(O)(H)**	A TRUE or FALSE value that, if TRUE, indicates that the list applet supports variable row heights (for example, Notes). By default, a variable row displays the first three lines. For the first three lines to appear completely, the Text Lines Per Note Row setting in the List tab in the Options window in Siebel applications needs to be set to 3 on client workstations.	TRUE — Indicates that the list applet supports variable row heights (for example, Notes).

See Also

"Applet" on page 29
"Hidden Object Types and Properties" on page 21
"List Column" on page 222

List Column

Siebel Object Types > Applet > List > List Column

Associates one list column to a list applet. Lists and list columns are usually created and edited in the Applet Designer.

You can add user properties to list columns.

Properties

Property	Description/Usage	Valid Values/Examples
Available - Language Override **(O)**	A TRUE or FALSE value. Provides the ability to change the list column can be associated with a list applet when a particular language/ locale is active.	Same as for available property.
Available **(O)**	A TRUE or FALSE value. If Available = FALSE, the list column does not show up in the list or the Columns Displayed dialogue window. See also "Show In List (O)" on page 227.	The list column can be shown in a list applet.
Bitmap Column Heading **(O)(H)**	A TRUE or FALSE value that is used only for Boolean list columns that specify a value for the Check Bitmap Identifier.	TRUE — The bitmap specified in the Check Bitmap Identifier displays as the list column heading. FALSE — The normal name/display name is the list column heading.
Check Bitmap Identifier **(O)(H)**	The bitmap to display in the TRUE state if the list column is of the type CheckBox.	
Content Fixup Name **(O)**	Determines if the links and images within the HTML page should be "fixed-up" to be proxied through the Siebel server (Inside Application) or left alone to be requested by the client's browser directly (Outside Application).	
Detail Applet **(O)**	An applet that displays related information for the field specified by the list column from the same business component.	
Display Format **(O)**	The format mask to apply to the data.	Example:$#,###,###,###.
Display Name **(O)**	Either the value to use as the list column heading or the key to the international message table.	Example: Expected Value.
Field **(O)**	The field for which the list column is displaying data.	Example: Amount Expected Value.

Property	Description/Usage	Valid Values/Examples
Field Retrieval Type **(O)**	Specifies that the field data is either rich content HTML data (Field Data), a standard reference to a web page (URL), or a specially constructed URL integrating data taken from the Siebel database and supporting the external content service (Symbolic URL).	
HTML Attribute **(O)**	Can be used to add HTML tag attributes to the HTML tags that the Standard Interactivity client creates to render the list column. Used with the Siebel Web Engine.	
HTML Display Mode **(O)**	Used with controls that are field values. Controls how the field value is displayed. Used with the Siebel Web Engine.	Valid values are: ■ DontEncodeData — Field value shown as in the database. ■ EncodeData — This is the default. Any markup language reserved characters in the field data are encoded so as to render them correctly. ■ FormatData — In addition to encoding markup language reserved characters, the field value is
HTML Height - Language Override **(O)**	Provides the ability to change the height of list column in pixels in the HTML thin client when a particular language/locale is active.	
HTML Height **(O)**	Height of list column in pixels in the HTML thin client.	
HTML Icon Map **(O)**	Can be set to an icon map object. Used to map field values to icons as defined in the icon map. Used with the SWE Engine.	Example: CHECK.

Property	Description/Usage	Valid Values/Examples
HTML List Edit **(O)**	A TRUE or FALSE value. Used with the Siebel Web Engine.	TRUE — This control is shown as a data input control (if the control allows updates) when rendering the applet in the Edit List mode. This is the DEFAULT. FALSE — Is shown as read-only.
HTML Max Chars Displayed - Language Override **(O)**	Provides the ability to change the maximum number of characters displayed in a list column when a particular language/locale is active.	
HTML Max Chars Displayed **(O)**	The maximum number of characters to be displayed in a list column. Only applies when the parent applet is in base mode or the list column is read only. Property does not limit the number of characters that can be entered into a text control in edit mode.	
HTML Only **(O)**	A TRUE or FALSE value. Used with the Siebel Web Engine.	TRUE — The control is used only in the HTML thin client. FALSE — The control is also used in the dedicated client.
HTML Row Sensitive **(O)**	Causes the Web engine to position the applet on the current record before invoking the method defined in the Method Invoked property. Used with the Siebel Web Engine.	Examples of methods that require this to be set to TRUE are EditRecord, DeleteRecord, and Drilldown.
HTML Sequence - Language Override **(O)**	Provides the ability to change the (order) position of this list column relative to other list columns in the applet in the HTML thin client when a particular language/locale is active.	
HTML Sequence **(O)**	The (order) position of this list column relative to other list columns in the applet in the HTML thin client. Indicates tab sequence.	

Property	Description/Usage	Valid Values/Examples
HTML Type **(O)**	Defines how the list column is to be rendered. If not set, the application uses the Type property. Used with the Siebel Web Engine.	Example: CheckBox.
HTML Width - Language Override **(O)**	Provides the ability to change the width of the list column in pixels in the HTML thin client when a particular language/locale is active.	
HTML Width **(O)**	The width of the list column in pixels in the HTML thin client (maximum is 2048).	
MVG Applet **(O)**	The applet to use for the pop-up MVG if the field for the list column is a multi-value field and the Runtime property is set to TRUE for the list column.	
Name **(R)**	The name of the list column.	
Pick Applet **(O)**	The applet to use for the pop-up picklist if the field for the list column has a picklist specified and the Runtime property is set to TRUE. This property (and corresponding applet) is ignored for list columns based on fields with static picklists, with one exception: it is used for picklists that have the Long List property set to TRUE.	
Popup Edit **(O)(H)**	A TRUE or FALSE value that, if TRUE, specifies that an ordinary text field requires a pop-up edit box.	
Prompt Text **(O)**	Not used.	Not applicable.
Read Only **(O)**	A TRUE or FALSE value. If the underlying field has a property setting of Read Only = TRUE, the list column is also read-only even if the list column's Read Only property is FALSE.	TRUE — Prevents data from being edited.

Property	Description/Usage	Valid Values/Examples
Runtime **(O)**	A TRUE or FALSE value.	TRUE — Makes a run-time check to see if a picklist, calculator, calendar, or MVG pop-up button is provided. FALSE — It is assumed that there is no picklist or MVG or that the field is Date or Time in DateTime.
Sequence - Language Override **(O)(H)**	Provides the ability to change the position of this list column in the list when a particular language/locale is active.	Integer value.
Sequence **(R)(H)**	The position of this list column in the list.	Integer value.
Show In List - Language Override **(O)**	Provides the ability to change the list column is shown in a list when a particular language/locale is active.	
Show In List **(O)**	A TRUE or FALSE value. If Show In List is FALSE and Available is TRUE, the list column is not visible in the applet unless the user selects it from the Columns Displayed window.	TRUE — The list column is shown in a list.
Show Popup **(O)**	A TRUE or FALSE value.	TRUE — Specifies that a new browser window is opened before invoking the method. FALSE — Specifies that the method is invoked in the same browser window.
Text Alignment - Language Override **(O)**	Provides the ability to change the text alignment of the list column when a particular language locale is active.	
Text Alignment **(O)**	The text alignment of the list column.	Valid values are Left, Center, and Right.

Property	Description/Usage	Valid Values/Examples
Text Alignment-Label - Language Override **(O)**	Provides the ability to change the Text-Alignment Label property for a list column when a particular language or locale is active. Note that this occurs automatically for locales that read right-to-left (such as Hebrew).	Center, Left, and Right.
Text-Alignment Label **(O)**	This item is used to align the text of a list column's label on a list applet. This is now a separate property from the text alignment of a list column field.	Center, Left, and Right.
Total Currency Code Expression **(O)**	The currency of the total value. Can be used to display the list column total in a converted currency.	For example, an opportunity with a currency code of dollars might have two products, one with a price in YEN and the other in USD. The list total could show the total product prices in the currency of the opportunity if the list column's Total Currency Code Expression were set to the following: Parent: 'Opportunity.Currency Code'.
Total Required **(O)**	A TRUE or FALSE value. This property also requires the Total Required property to be set on the list object definition.	TRUE — The list column is totaled. FALSE — The list column is not totaled.
Type **(O)(H)**	The list column type.	Valid values are CheckBox or TextBox.
Width - Language Override **(O)(H)**	Provides the ability to change the list column width when a particular language/locale is active.	
Width **(O)(H)**	The list column width.	

See Also

"Applet" on page 29
"Hidden Object Types and Properties" on page 21
"List" on page 220

For information about user properties, read *Siebel Developer's Reference.*

List Column Locale

Siebel Object Types > Applet > List > List Column > List Column Locale

Represents language-specific overrides used with the List Column object type.

Properties

Property	Description/Usage	Valid Values/Examples
Available **(O)**	A TRUE or FALSE value.	TRUE — The list column can be associated with a list applet.
Display Name **(O)**	A symbolic string for the list column heading.	
HTML Height **(O)**	Height of list column in pixels in the HTML thin client.	
HTML Max Chars Displayed **(O)**	The maximum number of characters to be displayed in a list column.	
HTML Sequence **(O)**	The (order) position of this list column relative to other list columns in the applet in the HTML thin client when the list applet is in Edit, Query or New Mode. Indicates tab sequence.	
HTML Width **(O)**	The width of the list column in pixels in the HTML thin client (maximum is 2048).	
Name **(R)**	The abbreviation of the language being used by the application.	Example: ENU.
Prompt Text **(O)**	Not used.	Not applicable.
Sequence **(O)(H)**	The position of this list column in the list.	Integer value.
Show In List **(O)**	A TRUE or FALSE value.	TRUE — The list column is shown in a list.
Text Alignment **(O)**	The text alignment of the list column.	Valid values are Left, Center, and Right.

Property	Description/Usage	Valid Values/Examples
Text-Alignment Label **(O)**	Provides the ability to change the Text-Alignment Label property when a particular language or locale is active. This is linked directly to the Text Alignment Label Language Override property for a list column.	
Width **(O)(H)**	The list column width.	

See Also
"Hidden Object Types and Properties" on page 21
"List Column" on page 222

List Column User Prop

Siebel Object Types > Applet > List > List Column > List Column User Prop

Communicates a property value to specialized C++ code. The values of user properties can be changed at configuration time. These values persist in the repository and the Siebel repository file.

Properties

Property	Description/Usage	Valid Values/Examples
Name **(R)**	The name of the user property.	
Value **(R)**	The value of the user property.	

Changing Undocumented User Properties

Customer developers can change user properties that have been documented. Only Oracle developers who create or change specialized C++ code should create or delete user properties that are not documented.

See Also
"Applet" on page 29
"List" on page 220
"List Column" on page 222

For information about user properties, read *Siebel Developer's Reference.*

List Locale

Siebel Object Types > Applet > List > List Locale

Represents language-specific overrides used with the List object type.

Properties

Property	Description/Usage	Valid Values/ Examples
Alpha Tab Search Field **(O)(H)**	If this property contains a value, causes the list applet to have an alphabet tab search bar. Also, the field specified is the field in the business component for the applet that the alpha search should be applied to.	
Name **(R)**	The abbreviation of the language being used by the application.	Example: ENU.

See Also

"Hidden Object Types and Properties" on page 21
"List" on page 220

Menu

Siebel Object Types > Menu

Provides a named menu that users can access in Siebel applications and to which menu items can be associated or removed in Siebel applications.

Properties

Property	Description/Usage	Valid Values/Examples
Name **(R)**	The name of the menu.	

See Also

"Menu Item" on page 231

Menu Item

Siebel Object Types > Menu > Menu Item

Associates a Command object definition with a Menu Item object definition. This association places a menu whose invoked method is specified in the Command object definition on the specified menu in a given position.

Properties

Property	Description/Usage	Valid Values/ Examples
Caption **(O)**	The text displayed in the menu.	Example: &Insert Record.
Command **(O)**	The name of the Command object definition that is to provide the method or accelerator for the menu item.	Example: Create Record.
Name **(R)**	The name of the menu item.	Example: Edit - Insert Record.
Position **(R)**	Identifies the ordinal position of the menu item. The top level positions are 1, 2, 3, and so on. The dot notation separates the ancestors of that menu item.	Example: 3.1.

See Also

"Menu" on page 231

Menu Item Locale

Siebel Object Types > Menu > Menu Item > Menu Item Locale

Represents language-specific overrides used with the Menu Item object type.

Properties

Property	Description/Usage	Valid Values/Examples
Caption **(O)**	The text displayed in the menu.	
Name **(R)**	The abbreviation of the language being used by the application.	Example: ENU.

See Also

"Menu" on page 231
"Menu Item" on page 231

Message

Siebel Object Types > Message Category > Message

Used with the Query function, in the user interface, to display text when a query is submitted and when the results are received. Used with <swe:messages> tag of the Siebel Web Engine.

Properties

Property	Description/Usage	Valid Values/Examples
Name **(R)**	Name of the message.	Example: Query Mode Prompt.
Text **(O)**	The text of the message.	Example: Enter Query.

See Also

"Message Category" on page 233

Message Category

Siebel Object Types > Message Category

The category of messages found in the user interface, for examples <swe:messages> tag, used by the Siebel Web Engine, and user defined error messages.

Properties

Property	Description/Usage	Valid Values/Examples
Name **(R)**	Name of the message category.	Example: User Defined Errors.

See Also

"Message" on page 232

Message Locale

Siebel Object Types > Message Category > Message > Message Locale

Represents the language-specific overrides used with the Message object type.

Properties

Property	Description/Usage	Valid Values/ Examples
Name **(R)**	The abbreviation of the language being used by the application.	Example: ENU.
Text **(O)**	The text of the message.	

See Also
"Message" on page 232
"Message Category" on page 233

Multi Value Field

Siebel Object Types > Business Component > Multi Value Field

Displays the child records of a parent record. When the button of a multi-value field is clicked, Siebel applications display the child records of the parent record. The data displayed in a multi-value field is that of the actively selected parent record.

Properties

Property	Description/Usage	Valid Values/Examples
Calculated **(O)**	A TRUE or FALSE value.	TRUE — Specifies that the field's value is calculated by the business component rather than retrieved from the database server.
Field **(R)**	The name of a field in the business component defined by the multi-value link object definition.	
Force Active **(O)**	A TRUE or FALSE value.	TRUE — Causes the data value to be always retrieved from the database.
Hidden **(O)**	A TRUE or FALSE value.	TRUE — Makes the field invisible to all dynamically created list views. For a business component data type of DTYPE_ID, the default is TRUE; otherwise, the default is FALSE.
Link Specification **(O)**	A TRUE or FALSE value. This is necessary if any business component used as the child of a link from this business component uses the Parent type of default values and expects to get a value.	TRUE — Specifies that the field's value is passed as a default value to a field in the child business component.
Multi Value Link **(R)**	The name of the multi-value link used when retrieving the data.	
Name **(R)**	The user-defined name for the field. The name must be unique within the business component. All references to the field are made through its name.	The name can contain spaces.

Property	Description/Usage	Valid Values/Examples
No Copy **(O)**	A TRUE or FALSE value that, if TRUE, specifies that during a Based On Last operation, the field's value is not copied into the newly created record. Also note that field value is not copied if Field's Multi Value Link object specifies No Copy as TRUE or corresponding Source Field object specifies No Copy as TRUE.	TRUE — Specifies that during a Based On Last operation, the field's value is not copied into the newly created record.
PickList **(O)**	The name of a picklist object definition used to display a list of valid values from which the user can choose.	
Read Only **(O)**	A TRUE or FALSE value.	TRUE — Prevents the field value from being changed by the user.
Required **(O)**	Not currently used.	
Use Default Sensitivity **(O)**	A TRUE or FALSE value.	TRUE — Causes the sensitivity mode that the application is running in (as defined in the data source) to be used in QBE searches that do not explicitly specify which sensitivity mode to use.

See Also

"Business Component" on page 91
"Field" on page 177

Multi Value Link

Siebel Object Types > Business Component > Multi Value Link

Retrieves the records displayed through a multi-value field.

Properties

Property	Description/Usage	Valid Values/Examples
Auto Primary **(O)**	A setting that determines how row ID values are populated in the primary ID field, based on a Siebel application-supplied list column labeled Primary in the multi-value group applet. The user can manually select the primary. Auto Primary determines how, if at all, the primary selection is defaulted.	DEFAULT — The first record automatically becomes the primary. NONE. — The user must manually specify the primary. SELECTED — Selection of a primary on one MVL causes the selection of a primary on the others. For example, as soon as a primary Shipping Address is indicated, it also becomes the primary Billing Address. This behavior is also true when the primary is not set. It is not true if the primaries already have values. SELECTED applies only when there are several multi-value links pointing to the same detail business component.
Check No Match **(O)**	A TRUE or FALSE value. In the parent Business Component, the foreign key field can house a value of No Match.	TRUE — If the application finds a No Match field, it should execute a separate query anyway. If the Auto Primary property is set to Default, it sets the first record returned as the primary. If the Auto Primary property is set to SELECTED, it ascertains whether any other multi-value link to this business component has indicated a primary, and set that record as the primary of this multi-value link. FALSE — The application should not execute a
Destination Business Component **(R)**	The name of the child business component.	
Destination Link **(R)**	The name of the link to use.	

Property	Description / Usage	Valid Values / Examples
Name **(R)**	The name for the multi-value link. The name must be unique among the multi-value links for this business component. All references to a multi-value link are made through its name.	The name can contain spaces, but cannot contain numbers or special characters.
No Associate **(O)**	A TRUE or FALSE value.	TRUE — Specifies that no associations can be created through the link. New detail records may be added. This property is ignored if the current view is in Admin Mode.
No Copy **(O)**	A TRUE or FALSE value. When FALSE, the copy operation on the child business component has the potential to fail due to unique indexes violations. In addition, the No Copy property only copies directly related multi-value group records. A multi-value link on a child business component with the No Copy property set to FALSE is ignored, and grandchild records are not copied using the Deep Copy feature.	FALSE — Specifies that child records through this link are copied.
No Delete **(O)**	A TRUE or FALSE value.	TRUE — Prevents deletion.
No Insert **(O)**	A TRUE or FALSE value.	TRUE — Prevents inserts.
No Update **(O)**	A TRUE or FALSE value.	TRUE — Prevents updates.
Popup Update Only **(O)**	A TRUE or FALSE value.	TRUE — Requires that the multi-value group window be popped up for any of the MVG values to be updated.

Property	Description/Usage	Valid Values/Examples
Primary Id Field **(O)**	Specifies the name of the field in the master business component that holds the row ID values pointing to primary records in the detail business component.	
Source Field **(O+)**	The name of a source field used to link the child business component and the source business component. The source business component is the business component that declares the multi-value link object definition. (+)This field is required if the source field is different from the Id.	
Type Field **(O)**	The name of a field in the DestBusComp business component used as a filter to retrieve a subset of records. This field name is used in conjunction with the Type Value property, which identifies the data value to use.	
Type Value **(O)**	The data value to use in filtering a subset of records. This value is used in conjunction with the Type Field property, which identifies the field to use.	

Property	Description / Usage	Valid Values / Examples
Use Primary Join **(O)**	A TRUE or FALSE value. Do not set both the Cache Data property of the Business Component object, and the Use Primary Join property of the Multi Value Link object type, to TRUE. (For more information, consult the Cache Data property of the Business Component object type.)	TRUE — Executes a primary join using the Primary Id Field property value to retrieve the primary child's data. The user must explicitly type EXISTS in queries to retrieve a non-primary MVF value. FALSE — The second query that retrieves all child records is executed. The user does not need to type EXISTS. In this case, the Siebel application generates an EXISTS query automatically.
Name **(R)**	The user-defined name for the field. The name must be unique within the business component. All references to the field are made through its name.	The name can contain spaces.
No Copy **(O)**	A TRUE or FALSE value that, if TRUE, specifies that during a Based On Last operation, the field's value is not copied into the newly created record. Also note that field value is not copied if Field's Multi Value Link object specifies No Copy as TRUE or corresponding Source Field object specifies No Copy as TRUE.	TRUE — Specifies that during a Based On Last operation, the field's value is not copied into the newly created record.
PickList **(O)**	The name of a picklist object definition used to display a list of valid values from which the user can choose.	
Read Only **(O)**	A TRUE or FALSE value.	TRUE — Prevents the field value from being changed by the user.

Property	Description/Usage	Valid Values/Examples
Required **(O)**	Not currently used.	
Use Default Sensitivity **(O)**	A TRUE or FALSE value.	TRUE — Causes the sensitivity mode that the application is running in (as defined in the data source) to be used in QBE searches that do not explicitly specify which sensitivity mode to use.

MVF Pick Map

Siebel Object Types > Business Component > Multi Value Field > MVF Pick Map

Specifies instructions for copying fields when a drop-down list is used by a field in a business component.

Properties

Property	Description/Usage	Valid Values/Examples
Constrain **(O)**	A TRUE or FALSE value.	TRUE — Defines the pick map as a constraint pick map. FALSE — Defines the pick map as a copy pick map.
Field **(R)**	The field in the parent business component with which the pick map is associated.	
No Clear **(O)**	A TRUE or FALSE value.	TRUE — Causes the field specified in the pick map not to be set to NULL when no value is picked from the drop-down list. FALSE — Clears the pick mapped field. However, the pick mapped field is not cleared when the drop-down list is unbounded.
Pick List Field **(O)**	The field in the picklist business component with which the pick map is associated.	

Copy and Constraint Pick Maps

There are two kinds of pick maps: copy pick maps and constraint pick maps. The type of pick map is determined by the value in the Constrain property. If a pick map is a copy pick map, then the value of the field in the picklist business component specified by the Pick List Field property is copied into the field of the parent business component specified by the Field property. If the pick map is a constraint pick map, then the value from the field in the parent business component specified by field is applied as a search specification on the field in the picklist business component specified by Pick List Field.

Constraining Pick Maps on Multi-Value Fields

You cannot use a field in the parent business component to constrain a picklist on a multi-value field. You can use only multi-value fields that are part of the detail business component.

See Also

"Business Component" on page 91
"Field" on page 177
"Multi Value Field" on page 234
"Pick List" on page 244
"Pick Map" on page 247
"Pick Map UpdOnlyIfNull" on page 248
"SVF Pick Map" on page 287

MVF Pick Map UpdOnlyIfNull

Siebel Object Types > Business Component > Multi Value Field > MVF Pick Map > MVF Pick Map UpdOnlyIfNull

Marks a copy pick map (read "Pick Map" on page 247) to perform only the copy operation if the field specified in the pick map UpdOnlyIfNull is not null. Constrain pick maps are also copy pick maps. In cases where the constraint is active, this results in no operation (no-op). In cases where the UpdOnlyIfNull causes the constraint to be ignored, the copy operation works. To avoid the copy operation use a calculated field as the field for the constrain pick map.

Properties

Property	Description/Usage	Valid Values/ Examples
Field **(R)**	The field in the parent business component to check for a NULL value before performing the copy operation specified by the parent pick map.	

See Also

"Business Component" on page 91
"Multi Value Field" on page 234
"MVF Pick Map" on page 240
"Pick List" on page 244
"Pick Map" on page 247
"Pick Map UpdOnlyIfNull" on page 248
"SVF Pick Map UpdOnlyIfNull" on page 288

Page Tab

Siebel Object Types > Application > Page Tab

Defines which screens are accessible through page tabs for an application.

Properties

Property	Description/Usage	Valid Values/Examples
Bitmap Category **(O)(H)**	Reserved for future use.	Not applicable.
Screen **(R)**	The screen to be exposed through a page tab.	Example: Accounts Screen.
Sequence **(O)**	The order of the page tabs for an application.	Integer value.
Text **(R)**	The text (or key to localized text) displayed on the page tab.	Example: Accounts.

See Also

"Application" on page 45
"Hidden Object Types and Properties" on page 21
"Screen Menu Item" on page 258

Page Tab Locale

Siebel Object Types > Application > Page Tab > Page Tab Locale

Represents language-specific overrides used with the Page Tab object type.

Properties

Property	Description/Usage	Valid Values/Examples
Name **(R)**	The abbreviation for the language being used by the application.	Example: ENU.
Text **(O)**	The text (or key to localized text) displayed on the page tab	

See Also

"Page Tab" on page 242

Pager Object

Siebel Object Types > Pager Object

Defines properties of business components that can be used to fill in default information in the Send Page applet.

Properties

Property	Description/Usage	Valid Values/Examples
Alpha Template **(O)**	The template used to generate default text for paging alphanumeric pagers. It can contain placeholders for the fields in the business components that are substituted with real values at run time. The template is stored in the MSGTEMPL\<language> (for example, msgtempl\enu) directory of the Siebel application directory.	The name of any template file available in the user's MSGTEMPL\<language> directory.
Business Component **(R)**	The business component that owns the paging-related properties.	Any buscomp that contains references to employees or pager-related fields.
Employee Field **(O)**	The field in the business component containing the login name of the employee to be paged. This login name is looked up in the Employee business component for the default phone number and pager ID to use. The Phone Field and the PIN Field properties, if defined, take precedence over the Employee Field property.	A field that contains the ID of an employee.
Name **(S)**	For Oracle use only. Do not modify.	

Property	Description/Usage	Valid Values/Examples
Numeric Template **(O)**	The template used to generate default message text for paging numeric pagers. It can contain placeholders for the fields in the business components that are replaced with real values at run time. The template is stored in the MSGTEMPL\<language> (for example, msgtempl\enu) directory of the Siebel application directory.	The name of any template file available in the user's MSGTEMPL\<language> directory.
Phone Field **(O)**	The field in the business component that contains the computer paging phone number to use. The computer paging phone number is different from the personal phone number assigned to some pagers. It is also different from the phone number that can be dialed to page with a touch-tone phone. Usually there is only one computer paging phone number available for each pager company. This number must support the TAPI protocol used in computer paging.	A field that contains the computer paging phone number.
PIN Field **(O)**	The field in the business component that contains the PIN number (or pager ID) of the pager to page. This is usually a six- or seven-digit number used to identify a pager.	A field that contains the pager PIN number.

Pick List

Siebel Object Types > Pick List

Lists valid values for a specific control or list column.

Properties

Property	Description/Usage	Valid Values/Examples
Bounded **(O)**	A TRUE or FALSE value.	TRUE — Requires any value entered into a field bounded by the drop-down list to be among the drop-down list's values.
Business Component **(R)**	The name of the business component whose data makes up the list of valid values.	

Property	Description/Usage	Valid Values/Examples
Long List **(O)**	A TRUE or FALSE value that specifies whether the Siebel application should attempt to position the cursor on the current value in a long list of values. When you specify Long List = TRUE for a picklist list, the Siebel application does not keep focus on the current record. A value of TRUE means that the picklist returns a large set of values, so to scroll through that list of values and position the cursor on the current value in the field could be very costly from a performance standpoint. For this reason, Long List does not keep focus.	TRUE — Recommendation is to set this attribute to TRUE if pick list is expected to contain more than 500 records.
Name **(R)**	The name of the picklist. The name must be unique among picklists. All references to the picklist are made through its name.	The name can contain spaces.
No Delete **(O)**	A TRUE or FALSE value.	TRUE — Prevents the user from deleting an existing picklist value.
No Insert **(O)**	A TRUE or FALSE value.	TRUE — Prevents the user from inserting an existing picklist value.
No Merge **(O)**	A TRUE or FALSE value.	TRUE — Prevents the user from merging two existing picklist values into a single value.
No Update **(O)**	A TRUE or FALSE value.	TRUE — Prevents the user from changing an existing picklist value.
Search Specification **(O)**	The name of the search specification to use in addition to the business component's usual search specification.	
Sort Specification **(O)**	The sort to apply for the picklist instead of the sort specification defined for the business component. Unlike the search specification, this does replace the business component sort specification.	

Property	Description/Usage	Valid Values/Examples
Static **(O)**	A TRUE or FALSE value.	TRUE — The pick list values come from a predefined list of values. These pick lists are called Static pick lists and they typically use PickList Generic and PickList Hierarchical business components. These business components display data entered by the administrator in the List of Values Administration Views. Static pick lists are rendered as drop-down menus in the interface. For more information, read “Hierarchical PickLists” in *Siebel Developer’s Reference*. FALSE — The pick list values come from the business component specified in the Business Component property. These pick lists are called dynamic pick lists.
Type Field **(O)**	The name of a picklist business component’s field to use in retrieving a subset of records. This field name is used in conjunction with the Type Value property, which is used to identify the data value to use.	
Type Value **(O)**	The data value to use in retrieving a subset of records. This value is used in conjunction with the Type Field property.	
Visibility Auto All **(O)**	A TRUE or FALSE value.	TRUE — If user has access to any “All” views, then visibility for a pick list business component is All; otherwise visibility is determined by the Visibility Type property.
Visibility Type **(O)**	Determines the visibility type when the business component is used as a pick list. For more information, see Visibility Auto All.	Valid values are All, Catalog, Group, Manager, Organization, Personal, Sales Rep, and Sub-Organization.

Source and Destination Fields

You do not specify source and destination field information here. This is done through pick map object definitions on specific fields.

See Also

“Pick Map” on page 247
“Pick Map UpdOnlyIfNull” on page 248

Pick Map

Specifies instructions for copying fields when a picklist is used by a field.

Properties

Property	Description/Usage	Valid Values/Examples
Constrain **(O)**	A TRUE or FALSE value.	TRUE — Defines the pick map as a constraint pick map. FALSE — Defines the pick map as a copy pick map.
Field **(R)**	The field in the parent business component that is connected to the pick map.	
No Clear **(O)**	A TRUE or FALSE value indicating how to set the Field property if no value is selected from the picklist.	FALSE — The Field property is set to NULL. TRUE — The Field property is not set to NULL.
Pick List Field **(O)**	The field in the picklist business component that is connected to the pick map.	
Sequence **(O)**	Integer value that specifies the order in which the field values are copied back to the business component after a particular row has been picked. Note that Siebel applications do not always use the Sequence property of Pick Map. Pick List Source Id always gets set first, overriding Sequence in the Pick Map.	

Copy and Constraint Pick Maps

There are two kinds of pick maps: copy pick maps and constraint pick maps. The type of pick map is determined by the value in the Constrain property. If a pick map is a copy pick map, then the value of the field in the picklist business component specified by the Pick List Field property is copied into the field of the parent business component specified by the Field property. If the pick map is a constraint pick map, then the value from the field in the parent business component specified by the Field property is applied as a search specification on the field in the picklist business component specified by the Pick List Field property.

Constraining Picklists on Multi-Value Fields

You cannot use a field in the parent business component to constrain a picklist on a multi-value field. You can use only multi-value fields that are part of the detail business component.

See Also
"Business Component" on page 91
"Field" on page 177
"MVF Pick Map" on page 240
"Pick List" on page 244
"Pick Map UpdOnlyIfNull" on page 248
"SVF Pick Map" on page 287

Pick Map UpdOnlyIfNull

Siebel Object Types > Business Component > Field > Pick Map > Pick Map UpdOnlyIfNull

UpdOnlyIfNull for a Copy Pickmap

Marks a copy pick map to perform the copy operation only if the values of the fields specified in the pick map UpdOnlyIfNull are null. For more information, read "Pick Map" on page 247.

UpdOnlyIfNull for a Constraint Pickmap

Acts as a copy pickmap if the value of the constraining field is null and the values of the fields specified in the constraint pick UpdOnlyIfNull are null. To disable the constraint when the constraining field value is null, use a calculated field as the field for the constraint pickmap.

Properties

Property	Description/Usage	Valid Values/ Examples
Field **(R)**	The field in the parent business component to check for the null condition before performing the copy operation specified by the parent pick map.	

Constrain By

Each pick map object definition has a child object definition of type Pick Map UpdOnlyIfNull. If you add one of these child object definitions to whichever pick map constrains the picklist, then Siebel applications ignore the constraint if the constraining value is NULL. You might still have a non-NULL picklist field that still returns no records, but this prevents you from seeing no records when the picklist field is NULL.

See Also
"Business Component" on page 91
"Field" on page 177
"MVF Pick Map UpdOnlyIfNull" on page 241
"Pick List" on page 244
"Pick Map" on page 247
"SVF Pick Map UpdOnlyIfNull" on page 288

Project

Siebel Object Types > Project

Groups other object definitions. It is also the center of the locking mechanism; that is, locking a project allows you to modify all object definitions within that project.

Properties

Property	Description/Usage	Valid Values/Examples
Allow Object Locking	Provides the ability to lock granular objects such as an applet or a business component.	
Language Locked **(S)**	If Locked is TRUE, records the language in which this project was locked. A project is always locked in a single language, which is the language Siebel Tools was in when the project was first locked.	Example: ENU, JPN, DEU.
Locked **(O)**	A TRUE or FALSE value that displays the lock status of the project.	TRUE — The project is currently locked.
Locked By Name **(O)**	If Locked is TRUE, displays the name of the person who has the project locked for editing. Otherwise, it is empty.	
Locked Date **(O)**	If Locked is TRUE, displays the date and time when the project was locked for editing. Otherwise, it is empty.	
Name **(R)**	Specifies the name uniquely.	
Owner Branch	Reserved for future use.	

Object Definitions and Projects

Every object definition in the repository (except for repositories and projects) needs to belong to a project. If an object definition has a parent (for example, a field of a business component), then the child belongs to the same project as its parent. You should group object definitions that tend to be modified together into the same project.

Report

Siebel Object Types > Report

Defines a paper or electronic representation of the information displayed by Siebel applications. The user cannot update the report's data.

The use of certain properties depends on whether the report is static or dynamic. Dynamic reports are tied to one or more specific views and are available only when the views are active; a dynamic report's data may also be global (that is, all data is reported on) or specific to the current record of the active view. Static reports are always available, even when there is no active view. Unlike the data in dynamic reports, however, a static report's data is only global.

Properties

Property	Description/Usage	Valid Values/ Examples
Access Base DB Name **(O)**	For Actuate reports, the name of the executable file (less the ROX extension) that Siebel applications run when this report is selected. Siebel applications look in the *Siebel_client_root*\reports*language_code* directory. For Access reports, the base Access database file that contains the Access report's report object definition. The default is base.mdb and is located in the *Siebel_client_root*\reports directory.	
Business Component **(R)**	The name of the business component whose data is the basis for the report.	
Business Object **(R)**	This property applies to static reports only. The name of the business component's business object.	
Class **(R)**	The class of the report.	Valid values are: CSSActuateReportViewer for an Actuate report. CSSCrystalReport and CSSWECrystalReport for a Crystal Reports report. CSSAccessReport and CSSReport are not supported.

Property	Description / Usage	Valid Values / Examples
Client Only **(O)**	A TRUE or FALSE value. A Client Only report cannot be scheduled to run at a later time, and cannot be run from a thin client.	TRUE — Specifies that the report cannot be run on the reports server.

Command Line **(O)**	A job to execute after exporting report data. Use only if the Class property value is CSSReport.	Examples:
		
		`C:\MSOFF95\EXCEL\EXCEL.EXE`
		
		`C:\SIEBEL\REPORTS\OPPORTUN.CSV`

Current Record Only **(O)**	This property applies to dynamic reports only. A TRUE or FALSE value.	TRUE — Causes the business component to export/report only on the current record.
Dynamic View **(O)**	This property applies to dynamic reports only. A TRUE or FALSE value. Indicates if the report inherits search criteria, sort criteria, and visibility from the view.	TRUE — The report is dynamic based on the view and queries run on that view. For example, Dynamic View is TRUE and the report is based on an Account business object. The user runs a query for accounts that begin with the letter T. If the user runs this report, it shows only accounts beginning with T.
Export System Fields **(O)**	A TRUE or FALSE value. If system fields are not required in your report, you should set this to FALSE.	TRUE — Exports the ID, Created, Created By, Updated, and Updated By fields. If FALSE, only the ID field is exported for the report.
Exported Table **(R)**	The name of the table to which the data is exported. Applicable only to CSSReport classes.	

Property	Description / Usage	Valid Values / Examples
Help Identifier **(O)**	A topic name for context-sensitive help.	The name is a constant character string, all uppercase, with no blanks. Example: `ID_REPORT_OPPORTUNITY_LIST`
Menu Text **(R)**	The submenu command text displayed under the menu command defined by the report category to which this report is assigned.	
Name **(R)**	The name of the report.	
ODBC Source Name **(O)**	The ODBC data source name for exporting report data. The default ODBC source names are specified in the CFG file.	
Parameter Applet **(O)**	Used to specify the name of an applet that is shown as the report parameters applet at run-time. Also determines if a report is parameterized or not: for example, if this is not empty, the specified applet is shown.	
Patch Field Names **(O)**	A TRUE or FALSE value.	TRUE — Causes a multiword field name to be patched into a single-word field name by replacing characters such as spaces, octothorpes (#), and slash marks (/) with underscores (_). It also appends _X to the end of the field name, and converts the resulting name to uppercase letters.
Preview **(O)**	Reserved for future use.	Not applicable.

Property	Description / Usage	Valid Values / Examples
Search Specification **(O)**	This property applies to static reports only. The conditional expression used to retrieve a subset of records. Using the Search Specification property on a dynamic report makes the report lose this behavior and ignore any predefined queries used on the current view.	
Sort Specification **(O)**	This property applies to static reports only. The sort expression used to order the records returned. You can set a sort specification on a report object definition to send the rows to the Actuate report driver via the data stream in sorted order. Otherwise the rows are sent in an unsorted, or default, sort order. Ordering the data coming from Siebel applications improves performance when the report runs and possibly is required to make the report work.	
Status Text **(O)**	Not currently used.	

Property	Description / Usage	Valid Values / Examples

Template Name **(O)**	This property applies to Actuate reports only. This property is left blank for Access reports.
	
	The name of the datastream library generated in Siebel Tools when the Tools menu option Actuate Report is invoked.
	
	Creates a Report Object Library (ROL) file to use in Actuate development and places this file in the location specified by the Tools CFG file.
	
	`[ActuateReports]`
	`ActuateDevWBDir = \Actuate7\eRDPro`
	`TemplateDestDir = D:\sea77\Tools\rptsrc\ENU\lib`
	
	The CFG file parameter does not specify the file's extension, which is ROL.
	
	The ROL file supplies information to Actuate about this report.	

View Mode **(O)**	This property applies to static Actuate reports only. The type of visibility that should be applied to the report's business component. This property only works when the business object is defined for the report object.	

Parameter Applet

This is the second step (using Siebel Tools) in creating a parameterized report. (The first step would be creating the parameter applet itself.) Next, the report design files need to be modified, to create parameter variables that correspond to the parameter applet's fields. This results in the parameter values chosen by the user at run time being written out to the ROV file in Actuate for the report to pick up during execution.

See Also

"Report Field" on page 254

Report Field

Siebel Object Types > Report > Report Field

Defines the field to be exported to or otherwise made available in the report.

Properties

Property	Description/Usage	Valid Values/Examples
Field **(R)**	The name of the field to be exported for the report.	
Name **(R)**	The name within the report.	

Business Component Restriction

A report field is a child of a parent report. These fields must be defined in the business component of the parent report. If no field is specified for a report, then all nonhidden fields are exported.

See Also

"Report" on page 250

Report Locale

Siebel Object Types > Report > Report Locale

Represents language-specific overrides used with the Report object type.

Properties

Property	Description/Usage	Valid Values/ Examples
Menu Text **(O)**	The submenu command text displayed under the menu command defined by the report category to which this report is assigned.	
Name **(R)**	The abbreviation of the language being used.	Example: ENU.
Status Text **(O)**	Not currently used.	

See Also

"Report" on page 250

Repository

Siebel Object Types > Repository

Consists of tables in a database where object definitions are stored.

CAUTION: Do not modify this object type. Any modifications can adversely affect performance and operation.

Properties

Property	Description / Usage
Name **(S)**	Name of the repository.

Schema Maintenance

All Schema Maintenance object types are for Oracle use only. The following is a list of the Schema Maintenance objects.

- Schema Maintenance Dependent Step
- Schema Maintenance Phase
- Schema Maintenance Phase Usage
- Schema Maintenance Platform Path
- Schema Maintenance Process
- Schema Maintenance Process Variable
- Schema Maintenance Step
- Schema Maintenance Step From
- Schema Maintenance Step Database Implementation
- Schema Maintenance Step Implementation
- Schema Maintenance Step To

Screen

Siebel Object Types > Screen

Groups or categorized views.

Properties

Property	Description/Usage	Valid Values/Examples
Bitmap Category **(O)(H)**	The name of the Bitmap Category object that contains the definitions of the Bitmap objects used in the screen. For example: For the Accounts screen, there is an Accounts bitmap category that contains bitmap object definitions for Logo and Screen Tab Icon. The Logo bitmap is displayed at the top of the view bar in the Accounts screen. The Screen Tab Icon bitmap is displayed on the tab of the Accounts screen.	Examples: Accounts, Contacts.
Default View **(O)**	The view that is used when the user clicks on a page tab for the screen. The view must be an entry in the screen view of the current screen.	Examples: Account List View, Contact Detail View.
Help Identifier **(O)**	A help ID for the screen for context-sensitive help.	Examples: ID_SCREEN_ACCOUNTS, ID_SCREEN_CONTACTS.
Name **(R)**	The name of a screen. All references to a screen are made through its name.	Example: Accounts Screen, Contacts Screen.
Unrestricted Viewbar **(O)(H)**	A TRUE or FALSE value. When Unrestricted Viewbar is FALSE, views outside of the business object for the default view of the screen does not appear in the thread bar. If you require the thread bar to show these views, do not use the Unrestricted Viewbar property.	TRUE — Displays all views in the screen in the view bar, even if they use different business objects.
Upgrade Behavior **(R)**	This property is set by Oracle and cannot be modified. Indicates if the object should be ignored if the repository merge is run with the Incorporate Custom Layout option.	Admin or Null Admin indicates that the object should be ignored. Null indicates that the object is to be processed by the Incorporate Custom Layout option.
Viewbar Text **(O)**	The label that appears below the optional Logo bitmap in the view bar.	Examples: Accounts, Contacts.

See Also
“Application” on page 45
“Bitmap Category” on page 87
“Hidden Object Types and Properties” on page 21
“Page Tab” on page 242
“Screen Menu Item” on page 258
“Screen View” on page 259

Screen Locale

Siebel Object Types > Screen > Screen Locale

Represents language-specific overrides used with the Screen object type.

Properties

Property	Description/Usage	Valid Values/Examples
Name **(R)**	The abbreviation of the language being used by the application.	Example: ENU.
Viewbar Text **(O)**	The label that appears below the optional Logo bitmap in the view bar.	

See Also
“Screen” on page 256

Screen Menu Item

Siebel Object Types > Application > Screen Menu Item

Adds items to the Screen menu for the application.

Properties

Property	Description/Usage	Valid Values/ Examples
Screen **(R)**	The name of the screen that is accessed with the menu item. If the screen definition contains multiple screen views, the menu item expands when highlighted to show each of those views.	

Property	Description/Usage	Valid Values/ Examples
Sequence **(O)**	The position of the menu item on the Screen menu for the application.	
Text **(R)**	The text to use for the menu item. If there is localized text for this menu item, then the value is a key into the message table to find the localized text.	

See Also

"Application" on page 45
"Screen" on page 256

Screen Menu Item Locale

Siebel Object Types > Application > Screen Menu Item > Screen Menu Item Locale (H)

Represents language-specific overrides for the Screen Menu Item object type.

Properties

Property	Description/Usage	Valid Values/Examples
Name **(R)**	The abbreviation of the language being used by the application.	Example: ENU.
Text **(O)**	The text to use for the menu item.	

See Also

"Screen Menu Item" on page 258

Screen View

Siebel Object Types > Screen > Screen View

Defines the collection of views for a screen.

Properties

Property	Description/Usage	Valid Values/Examples
Category **(O)**	A value that groups views in a screen with the same Category value into the same view bar category. Categories appear hierarchically in the view bar.	
Category Default View **(O)**	Defines the default view for this Screen View record of Type Aggregate Category or Detail Category. If this is not defined, SWE uses the first available view for the Category or the Default View defined for the screen.	Valid values are only those Views where the Parent Category is the same as this Category.
Category Name **(O)** **(R)** (see description)	The name that describes the Category that is used to group similar views on a screen. This is required for Screen View records where the Type is Aggregate Category or Detail Category.	
Client Restriction **(O)**	Specifies the availability of the view to different client platforms.	Example: Handheld client.
Display In Page **(O)**	When checked, this Screen View record is available in the runtime client on the relevant screen. This should always be checked for Screen View records where Type is Aggregate Category or Detail Category.	
Display In Site Map **(O)**	When checked, this Screen View record is available on the Site Map in the runtime client.	
Menu Text **(O)**	The text displayed in the Site Map for this particular View or Category.	

Property	Description / Usage	Valid Values / Examples
Object Manager Restriction **(O)**	The availability of the view when using different object managers. The view might be automatically suppressed under UNIX due to the UNIX Support property of the applet or business component classes in use, rendering the NT Only setting unnecessary.	Valid values: ■ NT Only — This view in this screen is available when the object manager is running in Windows-based server environments, but not when running in UNIX. ■ NULL — This view in this screen is available in all object manager environments.
Parent Category **(O)** **(R)** (see Description)	Defines how a view is associated to or grouped within a category or how a detail category is associated to an aggregate category. This is required for Screen View records where Type is Detail View or Detail Category. This is optional for records of Type Aggregate View. This must be left blank for records of Type Aggregate Category.	
Sequence **(O)**	The order that the view appears in the submenu for the screen. You can sequence the views displayed on the view bar, but not the categories (grouped views). Categories always appear last in the list.	Integer value.
Status Text **(O)**	Not used.	Not applicable.
Type **(R)**	Defines the type of screen view. Screen views are either actual views or are groupings of views or categories. The Type (in addition to the Parent Category property and applet visibility rules if relevant) generally defines where the view or category is displayed at runtime.	Aggregate Category, Aggregate View, Detail Category, and Detail View.
View **(R)**	The view that is being included in the screen.	Example: Account Detail - Contacts View.
Viewbar Text **(O)**	Text that appears on a view tab, a drop-down list, or link for a View or Category.	

Category Menu Text and Category Viewbar Text

For example, Account Detail-A View, Account Detail-B View, and Account Detail-C View all belong to the Category Detail. You want to show A, B, and C in the Site Map, but only A and B in the view bar. The attributes can be as follows:

	Category	Category Menu Text	Category Viewbar Text
Account Detail-A	Detail	Account Detail	Detail
Account Detail-B	Detail	Account Detail	Detail
Account Detail-C	Detail	Account Detail	

See Also

"Hidden Object Types and Properties" on page 21
"Screen" on page 256
"View" on page 325

For information about configuring the Mobile Web Client, read *Siebel Developer's Reference*.

Screen View Locale

Siebel Object Types > Screen > Screen View > Screen View Locale

Represents language-specific overrides used with the Screen View object type.

Properties

Property	Description / Usage	Valid Values / Examples
Category Menu Text **(O)**	Caption of the Category View that shows on the Site Map. If empty, view hidden on Site Map.	
Category Viewbar Text **(O)**	Caption of the Category View that shows on the view bar. If empty, view hidden on the view bar.	
Menu Text **(O)**	The text displayed in the Site Map for this particular View or Category.	
Name **(R)**	The abbreviation of the language being used by the application.	Example: ENU.
Status Text **(O)**	Not currently used.	
Viewbar Text **(O)**	Text that appears on a view tab, a drop down box, or link for a View or Category.	

See Also
"Screen View" on page 259

Search Category

Siebel Object Types > Search Category

A search category maps to a search index. Each entry defines a category that can be searched against. Search categories are not engine-specific and can be shared by different search definitions.

Properties

Property	Description/Usage	Valid Values/ Examples
Destination Field **(O)**	The field in the destination business component whose value equals the value of the result row ID, in the index business component. If no value is specified, then it defaults to Id, which is the row ID of the destination business component.	
Drilldown Buscomp **(O)**	The business component that the user is drilling into (destination). If this property is left blank, then it is assumed that the result category does not map to a database table.	
Drilldown View **(O)**	The view that appears when the user drills down. If this property is left blank, then it is assumed that the result category does not map to a database table.	
Name **(R)**	Logical name for the category.	
Preview **(O)**	The view that appears (pops up in a separate browser window) when a user invokes the Preview button in the Search Center. The Preview view is usually configured to be a read-only view.	Example: For the Find Object Accounts, the Preview attribute is the view Accounts Preview View.
Result Identifier **(O)**	An acronym that is used in the Summary column in the search result.	
Search Index **(R)**	Name of the search index.	Valid values are Auction Item, Catalog Category, Decision Issue, External Document, Literature, Product, Product News, Product by Price List, Resolution Documents, Solution, and Template.

See Also
"Search Pick View" on page 274
"Search Visibility View" on page 276

Search Custom Result Field

Siebel Object Types > Search Engine > Search Definition > Search Custom Result Field

Specifies the search execution and display properties for the result fields under the engine.

Properties

Property	Description/Usage	Valid Values/ Examples
Display Format **(O)**	The format mask to apply to the data.	
Display Name **(O)**	The name to display in the user interface.	
Name **(R)**	The logical name of the field.	
Scaling Factor **(O)**	The multiplication factor for numeric columns. The value specified here is simply multiplied with the actual value for the column.	
Sequence **(R)**	The order in which columns should appear.	Integer value.
Sequence - Language Override **(O)**	Provides the ability to change the sequence order when a particular language/locale is active.	
Text Alignment **(O)**	The alignment of the text in the column.	
Text Alignment - Language Override **(O)**	Provides the ability to change the text alignment in a column when a particular language/locale is active.	
Use In Search **(O)**	Indicates if this column is to be included in a search.	
Visible **(O)**	A TRUE or FALSE value.	TRUE — The field is displayed.
Visible - Language Override **(O)**	Provides the ability to change the visibility of a field when a particular language/locale is active.	

Property	Description/Usage	Valid Values/ Examples
Width **(R)**	The width in pixels of the display column.	
Width - Language Override **(O)**	Provides the ability to change the width in pixels of the display column when a particular language/locale is active.	

See Also

"Search Definition" on page 265
"Search Definition Category" on page 266

Search Custom Result Field Locale

Siebel Object Types > Search Engine > Search Definition > Search Custom Result Field > Search Custom Result Field Locale

Represents language-specific overrides used with the Search Custom Result Field object type.

Properties

Property	Description/Usage	Valid Values/ Examples
Display Name **(O)**	The name to display in the user interface.	
Name **(R)**	The abbreviation of the language being used by the application.	Example: ENU.
Sequence **(R)**	The order in which columns should appear.	Integer value.
Text Alignment **(O)**	The alignment of the text in the column.	
Visible **(O)**	A TRUE or FALSE value.	TRUE — The field is displayed.
Width **(R)**	The width in pixels of the display column.	

See Also

"Search Custom Result Field" on page 264

Search Definition

Siebel Object Types > Search Engine > Search Definition

Identifies a group of search categories that can be searched against.

Properties

Property	Description/Usage	Valid Values/Examples
Default Search	Information not available.	
Name **(R)**	Logical name of this search definition. (This is specified as a value for the SearchDefName property in the CFG file.)	
Sort Specification **(O)**	A sort expression used to order the records returned. Only result field names can be specified here.	Examples: SCORE DESC, FILENAME ASC.

See Also

"Search Engine" on page 267

Search Definition Category

Siebel Object Types > Search Engine > Search Definition > Search Definition Category

An entry here maps a Search Category to a Search Definition.

Multiple search categories that refer to the same search index cannot be associated to the same search definition. For example, if Solution External and Solution Internal categories refer to the same search index Solution, and only differ in their filter search specifications, they cannot be specified under the same search definition. The application returns an error at run time. This is due to a limitation with the way Hummingbird SearchServer handles searches.

Properties

Property	Description/Usage	Valid Values/Examples
Display Name **(R)**	The actual display name of the category. This name appears in the category drop-down list in the Basic Search Applet/ Advanced Search Applet.	
Filter Search Spec **(O)**	Specifies the search spec to be applied on filter fields.	
Name **(R)**	Identifies a category that can be searched on for the selected search definition.	
Sequence **(R)**	Specifies the order in which the search categories are listed in the search applet for the parent application.	Integer value.

Property	Description/Usage	Valid Values/Examples
Sequence - Language Override **(O)**	Provides the ability to change the order in which the search categories are listed in the search applet for the parent application when a particular language/locale is active.	
Use Filtered Search **(O)**	A TRUE or FALSE value. Not currently used.	TRUE — Set this flag to TRUE, if you want to turn on filtered searching for this category.

See Also

"Search Definition" on page 265
"Search Engine" on page 267

Search Definition Category Locale

Siebel Object Types > Search Engine > Search Definition > Search Definition Category > Search Definition Category Locale

Represents language-specific overrides used with the Search Definition Category object type.

Properties

Property	Description/Usage	Valid Values/Examples
Display Name **(R)**	The actual display name of the category. This name appears in the category drop-down list in the Basic Search Applet/Advanced Search Applet.	
Name **(R)**	The abbreviation of the language being used by the application.	Example: ENU.
Sequence **(O)**	Specifies the order in which the search categories are listed in the search applet for the parent application.	

See Also

"Search Definition Category" on page 266

Search Engine

Siebel Object Types > Search Engine

Represents an external search engine that may be integrated with Siebel applications.

Properties

Property	Description/Usage	Valid Values/Examples
Class **(R)**	The class name of the search adapter. For example, CSSFulcrumSearchAdapter handles all the low-level interactions with the Hummingbird SearchServer, be they administrator or execution functions.	
Name **(R)**	The name of the search engine.	Example: Hummingbird SearchServer.

See Also

"Search Definition" on page 265
"Search Result Field" on page 275

Search Engine Field (H)

Siebel Object Types > External Search Engine (H) > Search Engine Table (H) > Search Engine Field (H)

Specifies the individual fields within a search table.

Properties

Property	Description/Usage	Valid Values/ Examples
Column **(R)**	The vendor-specific name of the physical column in the search index.	
Display Format **(O)**	The format mask to apply to the data.	
Display Name **(R)**	The name to display in the user interface.	
Name **(R)**	The logical name of the field.	
Scaling Factor **(O)**	The multiplication factor for numeric columns.	
Sequence **(R)**	The order of columns.	Integer value.
Text Alignment **(O)**	The alignment of the text in the column.	
Text Length **(R)**	The amount of storage (bytes) allocated to a field.	
Type **(R)**	A picklist with valid values displayed.	

Property	Description/Usage	Valid Values/ Examples
Use In Search **(O)**	A value that indicates if this column is to participate in a search.	
Visible **(O)**	A TRUE or FALSE value.	TRUE — Displays the field.
Width **(R)**	The width in pixels of the display column.	

Field Types

Two field types are mandatory for all search tables: Internal ID and FQ File Name. Two more field types are required for database search tables: Source and Row ID. The seven available field types are as follows:

- **External File.** The actual content of the document.
- **FQ File Name.** The fully qualified filename of the document.
- **File/Summary.** Relevant information about the filename and/or the summary.
- **Internal ID.** An internally generated ID.
- **Row ID.** The database row ID of this row.
- **Source.** The source (table) of the data.
- **Summary.** The summary of the indexed row.

See Also

"Hidden Object Types and Properties" on page 21
"Search Engine Table (H)" on page 270

Search Engine Field Locale (H)

Siebel Object Types > External Search Engine (H) > Search Engine Table (H) > Search Engine Field (H) > Search Engine Field Locale (H)

Represents language-specific overrides used with the Search Engine Field object type.

Properties

Property	Description/Usage	Valid Values/ Examples
Display Name **(R)**	The name to display in the user interface.	
Name **(R)**	The abbreviation of the language being used by the application.	Example: ENU.

See Also
"Hidden Object Types and Properties" on page 21
"Search Engine Field (H)" on page 268

Search Engine Pick View (H)

Siebel Object Types > External Search Engine (H) > Search Engine Pick View (H)

Identifies the applet that can receive selected items from the search result list. Applies only to specialized applets that can handle such items. An example of the specialized applets that can be specified as search engine pick views are the Service Request Resolution and Resolution Items Administration views.

Properties

Property	Description/Usage	Valid Values/Examples
Applet **(R)**	The name of the applet.	
Name **(R)**	The logical name of the applet.	
View **(R)**	The name of the view containing the applet.	

See Also
"External Search Engine (H)" on page 176
"Hidden Object Types and Properties" on page 21

Search Engine Table (H)

Siebel Object Types > External Search Engine (H) > Search Engine Table (H)

Models a vendor-specific search collection that is then used by the Siebel applications client. It consists of one or more search field definitions.

Properties

Property	Description/Usage	Valid Values/ Examples
DB Table Flag **(O)**	A TRUE or FALSE value.	TRUE — Indicates that this table is a database table.
Drilldown Applet **(O)**	Applicable only if DB Table Flag is TRUE. The name of the applet in the drilldown view that contains the row ID in the search result row.	

Property	Description/Usage	Valid Values/ Examples
Drilldown View **(O)**	Applicable only if DB Table Flag is TRUE. The name of the view navigated to when you double-click a row in the search result list.	
File Name Prefix **(O)**	The prefix given to the search results filename. Used for saving database search results. The search results filename is generated from the text in this property and the ROW_ID value.	If the File Name Prefix for service requests is SR:, search results from the service requests table have filenames such as SR:1-ABC.
Name **(R)**	The (vendor-specific) physical name of the table/ collection in the search index.	
Result Identifier **(O)**	The abbreviation in the search result summary.	
Sequence **(O)**	A sequence number used to order the tables.	Integer value.
Title **(O)**	The table used in the source selection list of the Search dialog box.	

One Search Engine Per Table

Use the Search Engine Table object type with caution. The search field should be defined for only one search table in a search definition, because the search tables are restricted to having the same fields in them.

See Also

"External Search Engine (H)" on page 176
"Hidden Object Types and Properties" on page 21

Search Engine Table Locale (H)

Siebel Object Types > External Search Engine (H) > Search Engine Table (H) > Search Engine Table Locale (H)

Represents language-specific overrides used with the Search Engine Table object type.

Properties

Property	Description/Usage	Valid Values/ Examples
Name **(R)**	The abbreviation of the language being used by the application.	Example: ENU.
Title **(O)**	The table used in the source selection list of the Search dialog box.	

See Also

"Hidden Object Types and Properties" on page 21
"Search Engine Table (H)" on page 270

Search Filter Field

Siebel Object Types > Search Index > Search Table > Search Filter Field

Identifies and defines a field of the business component that is to be indexed separately as a filter field.

Properties

Property	Description/Usage	Valid Values/ Examples
Column Name **(R)**	The name of the physical column in the search index, which varies with the vendor of your database software.	
Data Type **(O)**	Specifies the data type of the column.	Valid values are Varchar and Character.
Field Type **(O)**	Specifies the field type of the column.	Valid values are Constraint and Key.
Index Mode **(O)**	Specifies the index mode of the column. The index mode specifies the type of indexing that is applied on the values in this field.	Valid values are Literal, None, and Normal.
Name (also referred to as Business Component Field) **(R)**	The active fields of the business component that is associated with the parent Search Index object.	
Sequence **(R)**	The order in which columns appear.	Integer value.
Text Length **(O)**	The amount of storage, in bytes, allocated to a field.	

See Also

"Search Index Field Map" on page 273
"Search Table" on page 276

Search Index

Siebel Object Types > Search Index

A logical name for a group of search engine specific entities.

The search index for Hummingbird SearchServer is a Fulcrum index table. For example, the search index Service Request maps to, in the case of Hummingbird SearchServer, an index table called FUL_SRREQ.

Properties

Property	Description/Usage	Valid Values/Examples
DB Table **(O)**	A TRUE or FALSE value.	TRUE — Indicates that the object maps to a database table.
Index Buscomp **(O)**	Business component that the index maps to. Index Business component property can be null if based on an external directory.	
Name **(R)**	Logical name for the index.	

See Also

"Search Table" on page 276

Search Index Field Map

Siebel Object Types > Search Index > Search Table > Search Index Field Map

Identifies a field of the business component that is to be indexed.

Also specifies the result field (for the parent engine) that it maps to. Only fields marked as Data (Public) are visible here. Contents of the index fields are indexed in the Search Engine Result Fields.

Properties

Property	Description/Usage	Valid Values/Examples
BC Field **(R)**	The active fields of the applet (or business component) that is associated with the parent Search object. These fields are included in the Search.	
Field Type **(O)**	Specifies the type of the Index Field. For example, File Attachment, File Revision Number, and Organization Visibility.	Valid values are Attachment, Group, Organization, Rev Num, and Row Id.
Name **(R)**	Identifies a name for the field.	
Result Field **(O)**	Specifies the result field that maps to the index field. For example, you may want SR Abstract to be indexed into the SUMMARY result field. By default, the contents of the index field are automatically indexed into the DOCUMENT_TEXT result field.	
Sequence **(O)**	Not used.	Not applicable.

See Also

"Search Table" on page 276

Search Pick View

Siebel Object Types > Search Category > Search Pick View

Identifies the applet that can receive the selected item from the Search Results List as an association record. This applies only to applets that can handle such associations.

Properties

Property	Description/Usage	Valid Values/ Examples
Applet Name **(R)**	Name of the applet.	
Field Name **(O)**	Not used.	
Name **(R)**	Logical name of the pick view.	
View Name **(R)**	Name of the view containing the applet.	

See Also
"Search Category" on page 263

Search Result Field

Siebel Object Types > Search Engine > Search Result Field

Lists all fields that are included with or used by every search index table created for the search engine.

Properties

Property	Description/Usage	Valid Values/Examples
Column **(R)**	The name of the physical column in the search index, which varies with the vendor of your database software.	
Create Column **(O)**	A TRUE or FALSE value. A flag that denotes which columns are actually created in the Search Table.	TRUE — Column to be created in Search Table.
Data Type **(O)**	Specifies the data type of the column.	Valid values are Varchar, Character, and ApVarchar.
Index Mode **(O)**	Specifies the index mode of the column.	Valid values are Literal, None, and Normal.
Name **(R)**	The logical name of the field.	
Sequence **(R)**	Specifies the sequence in which the columns are created and used.	Integer value.
Text Length **(O)**	The amount of storage, in bytes, allocated to a field.	
Type **(R)**	A picklist with valid values displayed.	Valid values are Data (Private), Data (Public), and System. Data (Private) fields are fields that are created by the user. Fields marked System or Data (Public) cannot be deleted.

See Also
"Search Engine" on page 267

Search Table

Siebel Object Types > Search Index > Search Table

Each entry specifies the physical index table or collection, depending on the search engine.

Properties

Property	Description/Usage	Valid Values/Examples
Engine Name **(R)**	Name of the search engine.	Example: Hummingbird SearchServer
Name **(R)**	Actual name of the index table or the collection.	

See Also

"Search Index" on page 273

Search Visibility View

Siebel Object Types > Search Category > Search Visibility View

Search Visibility View object properties are not used in the current version of the application.

Properties

Property	Description/Usage	Valid Values/ Examples
Name **(R)**	Name of the search visibility view.	
Sequence **(R)**	The order of this view relative to other views.	Integer value.
Visibility Applet **(O)**	Identifies which of the applets in the view is the one where visibility is controlled. Usually this is the list (upper) applet in a list-form view or the master applet in a master-detail view.	
Visibility View **(O)**	Identifies which view is the one where visibility is controlled.	

Server Component Event Subtype

Siebel Object Types > Server Component Type > Server Component Event Type > Server Component Event Subtype

Event subtypes are code references that define the event. Each event subtype is defined to a specific security level, so when an associated event occurs, the event has an intrinsic severity level to which it is associated.

CAUTION: Do not modify this object type. Any modifications can adversely affect performance.

Properties

Property	Description/Usage
Code Symbol **(S)**	Uniquely identifies the event subtype as an enum. This provides for faster searching than using the full descriptive name.
Description **(S)**	Same as the Comments property.
Display Name **(S)**	Name of the event subtype as it appears in the server manager.
Name **(S)**	Name of the object definition.

Server Component Event Type

Siebel Object Types > Server Component Type > Server Component Event Type

Events are logged at the server level and the component level. Component-level event types are events that relate to a specific server component.

CAUTION: Do not modify this object type. Any modifications can adversely affect performance.

Properties

Property	Description/Usage
Code Symbol **(S)**	Uniquely identifies the event type as an enum. This provides for faster searching than using the full descriptive name.
Description **(S)**	Same as the Comments property.
Display Name **(S)**	Name of the event type as it appears in the server manager.
Name **(S)**	Name of the object definition.

Server Component Parameter

Siebel Object Types > Server Component Type > Server Component Parameter

Specifies parameters within a server component type.

CAUTION: Do not modify this object type. Any modifications can adversely affect performance.

Properties

Property	Description / Usage
Code Symbol **(S)**	Uniquely identifies the parameter as an enum for use in CcfGetParam calls. This provides for faster searching than using the full descriptive name.
Component Definition Settable **(S)**	See Enterprise Settable property.
Component Settable **(S)**	See Enterprise Settable property.
Data Type **(S)**	Data type of the values the parameter stores.
Default Value **(S)**	Specify a default value for the parameter here. The parameter assumes this value if unchanged in the server manager.
Description **(S)**	Longer text description of the parameter, explaining its purpose.
Display Name **(S)**	Name of the event type as it appears in the server manager.
Enterprise Settable **(S)**	A TRUE or FALSE value. The settable flags specify the levels at which this parameter may be set, from enterprise down through task. When Enterprise Settable is set to TRUE, the parameter is set for the whole enterprise. If a parameter is set at a particular level, its value is inherited at all lower levels. Restricting where it is settable ultimately determines the range of levels at which it can be used as well as set.
Name **(S)**	Name of the object definition.
Performance Impact	Information not available.
Pick List **(S)**	Name of the picklist.
Requires Component Reconfig **(S)**	A TRUE or FALSE value. Process by which the configuration of a component can be changed affecting new tasks but allowing old tasks to continue running, unaffected, using the old configuration. In general, configuration changes are picked up automatically by new tasks, but there are a small number of arguments for which changes do not take effect unless the component is taken through this reconfiguration process. This flag identifies the arguments for which this is true.
Server Settable **(S)**	See Enterprise Settable property.
Task Settable **(S)**	See Enterprise Settable property.
Text Length **(S)**	For arguments with a data type of Text. Specifies the maximum length in characters.
Visibility	Information not available.

Server Component State Value

Siebel Object Types > Server Component Type > Server Component State Value

State values contain information about the current operation of a task or the component for which the task is running. Component tasks periodically update their state values to indicate information about their current processing, such as the current phase of operation. Component-level state values refer to the state of the component as a whole.

CAUTION: Do not modify this object type. Any modifications can adversely affect performance.

Properties

Property	Description/Usage
Code Symbol **(S)**	Uniquely identifies the server component state value as an enum. This provides for faster searching than using the full descriptive name.
Data Type **(S)**	Data type of the values the component state value stores.
Description **(S)**	Same as Comments property.
Display Name **(S)**	Name of the component state value type as it appears in the server manager.
Level **(S)**	Specifies the level (Component, Server, or Task) for which the state value can be set.
Name **(S)**	Name of the object definition.
Text Length **(S)**	For data type of Text, specifies the maximum length in characters.
Visibility	Information not available.

Server Component Statistic

Siebel Object Types > Server Component Type > Server Component Statistic

Various statistics are recorded at the task level for all server component tasks. These statistics are used to monitor the progress and performance of a component or optimize system performance.

CAUTION: Do not modify this object type. Any modifications can adversely affect performance.

Properties

Property	Description / Usage
Calculated **(S)**	A TRUE or FALSE value. If TRUE, the component statistic object definition is calculated.
Calculated Value **(S)**	The script for the calculation.
Code Symbol **(S)**	Uniquely identifies the server component statistic as an enum. This provides for faster searching than using the full descriptive name.
Data Type **(S)**	Data type of the values being stored for this object.
Description **(S)**	Same as Comments property.
Display Name **(S)**	Name of the component statistic as it appears in the server manager.
Name **(S)**	Name of the object definition.
Visibility	Information not available.

Server Component Subsystem

Siebel Object Types > Server Component Type > Server Component Subsystem

A subsystem of the server component type object.

CAUTION: Do not modify this object type. Any modifications can adversely affect performance.

Properties

Property	Description / Usage
Name **(S)**	Name of the subsystem.
Subsystem **(S)**	Name of the subsystem.

Server Component Type

Siebel Object Types > Server Component Type

Siebel Server supports multiple component types; each type performs a specific function or job. A component type is configured with a set of arguments that determine its behavior to create a defined component (or simply component).

CAUTION: Do not modify this object type. Any modifications can adversely affect performance.

Properties

Property	Description/Usage
Code Symbol **(S)**	Uniquely identifies the server component type as an enum. This provides for faster searching than using the full descriptive name.
Description **(S)**	Same as Comments property.
Display Name **(S)**	Name of the component type as it appears in the server manager.
Key Based Routing **(S)**	A TRUE or FALSE value. If TRUE, some Batch Mode Components can control the partitioning and routing of requests. These components can register a key or set of keys so that they get only the requests that match these keys.
Name **(S)**	Name of the object definition.

Single Value Field

Siebel Object Types > Business Component > Single Value Field

Displays the properties of field object definitions that are applicable to only those fields that are not multi-value fields. When selected, only records that are not multi-value fields are displayed.

Properties

Property	Description/Usage	Valid Values/Examples
Calculated **(O)**	A TRUE or FALSE value.	TRUE — Causes the field's value to be calculated by the business component.
Calculated Value **(O)**	An expression to calculate the field's value.	
Column **(R)**	The name of the database table's column. The column's name is case-sensitive and must be specified the same as it is in the database. Every reference to the field is made through its name. This property is not allowed for calculated fields but is required for other fields.	
Currency Code Field **(O)**	The name of the currency code field. Used with a data type of DTYPE_CURRENCY.	Default is Currency Code.

Property	Description / Usage	Valid Values / Examples
Exchange Date Field **(O)**	The name of the exchange date field. Used with a data type of DTYPE_CURRENCY.	Default is Exchange Date.
Force Active **(O)**	A TRUE or FALSE value. For best performance, do not use Force Active except where necessary.	TRUE — Specifies that the data value is always retrieved from the database.
Force Case **(O)**	The case for the field value.	Valid values are UPPER, LOWER, and FIRSTUPPER.
Hidden **(O)**	A TRUE or FALSE value that governs whether the field is displayed. For a business component data type of DTYPE_ID, the default is TRUE; otherwise, the default is FALSE.	TRUE — Makes the field invisible to all dynamically created list views.
Join **(O)**	The name of the join object definition used by this field. Used when the value of the field is retrieved from the table defined by the join. See also "Join" on page 212.	
Link Specification **(O)**	A TRUE or FALSE value. A link specification is necessary if any business component used as the child of a link from this business component uses the Parent type of default values and expects to get a value. For best performance, only use if you need to predefault the parent value to the child value through the link, because this setting results in the column always being retrieved any time this object is active. Similar to ForceActive.	TRUE — Passes the field's value as a default value to a child business component's field.
Name **(R)**	The user-defined name for the field. The name must be unique within the business component. All references to the field are made through its name.	The name can contain spaces.

Property	Description/Usage	Valid Values/Examples
No Copy **(O)**	A TRUE or FALSE value.	TRUE — During a Based On Last operation, causes the field's value not to be copied into the newly created record.
Oracle Sequence Object **(O)**	The name of an Oracle sequence object. The sequence is used for field defaults when creating a new record.	
PickList **(O)**	The name of a picklist object definition used to display a list of valid values from which the user can choose.	
Post Default Value **(O)**	Used to fill in a default value for a field if the user leaves the field blank and saves the record. Just before the row is committed, if the field is blank, the Post Default Value is used for the field. If the field has a user-entered value, the Post Default Value is not used. This property is not validated for a calculated field.	Limited to 255 characters.
Pre Default Value **(O)**	The value used for an empty field when a new record is created through an Add New Record or Copy Record operation. The value might be changed by the user before the record is written to the database. This property is not validated for a calculated field.	Limited to 255 characters.
Precision **(O)**	The maximum number of digits, to the left and right of the decimal point, for a numeric field.	Default is 16. The maximum value available is database dependent.

Property	Description / Usage	Valid Values / Examples
Read Only **(O)**	A TRUE or FALSE value. Certain fields are read-only even if the Read Only property is FALSE. The Effective Start Date field for the following business components is always read-only: ■ Account Product ■ Promotion ■ Promotion Account ■ Promotion Account Product ■ Promotion Administration ■ Promotion Product In general, fields that are based on a joined table are always read-only. There are two exceptions to this rule: joins to extension tables and joins to active intersection tables. See also "Join" on page 212.	TRUE — Prevents the field value from being changed by the user. Note that it is also possible to make a field read-only at the applet level.
Required **(O)**	A TRUE or FALSE value. There is a user property available to control Required based on an expression. For more information, read *Siebel Developer's Reference*.	TRUE — Requires a value to be entered into the field before the record can be written to the database.
Scale **(O)**	The maximum number of digits to the right of the decimal point for a numeric field.	Values are 6 for numeric values and 0 for integers. The maximum value available is database dependent.
Text Length **(O)**	The maximum length of the text in the field. This property is usually ignored, and the length is retrieved from the underlying column definition.	
Type **(R)**	The field's data type. Be sure that the type you select corresponds to the data type in the physical layer.	

Property	Description/Usage	Valid Values/Examples
Use Default Sensitivity **(O)**	A TRUE or FALSE value.	TRUE — Causes the sensitivity mode that the application is running in (as defined in the data source) to be used in QBE searches that do not explicitly specify which sensitivity mode to use.
Validation **(O)**	The expression used to ensure data correctness when data is entered. Should be used only with single value fields, not MVFs.	The property text is limited to 255 characters.

See Also
"Business Component" on page 91
"Field" on page 177
"Multi Value Field" on page 234

String (H)

Siebel Object Types > String Map (H) > String

The String object is for Oracle use only.

String Map (H)

Siebel Object Types > String Map

The String Map object is for Oracle use only.

Sub Report

Siebel Object Types > Report > Sub Report

Manages the child records of a report (where the report manages the parent records). Subreports are specific to Microsoft Access, Actuate, and general reports.

Properties

Property	Description / Usage	Valid Values / Examples
Business Component **(R)**	The name of the business component whose data is the basis for the subreport. This business component is linked to the business component of the previous subreport declaration (if any) or the top-level report declaration. The business component must be declared as the Child business component to the business component specified in the report object. Typically the report object business component is synonymous with a specific business object and the business components that are available to subreport are business components that exist within the business object with a valid link declared. Reports are similar to views in that if the relationship from the subreport to the parent report cannot be configured into a view, then the subreport cannot be used.	
Exported Table **(R)**	(CSSReport and CSSCrystalReport classes only.) The filename or the table that the subreport data is exported to.	
Inter Table **(O)**	The name of the intersection table to use when there is a many-to-many relationship between this business component and the parent business component.	
Inter Table Name **(O)**	(CSSReport and CSSCrystalReport classes only.) The name of the table or the filename that the subreport intersection table data is exported to. When naming the intersection table, the convention is to use the parent and child business component separated by a slash mark (/).	
Position **(O)**	The ordinal position of the subreport. The first-level positions are 1, 2, 3, 4, and so on. The next level identifies the corresponding parent of the subreport; if Position is 1, then all successive subreport positions would be 1.1, 1.2, 1.3, and so on.	
Search Specification **(O)**	An additional search expression that can be applied to the subreport's business component.	
Sort Specification **(O)**	An additional sort specification that can be applied to the subreport's business component.	

Ordinal Position

The parent-child relationship between subreports is based on the ordinal position of the subreport's declaration. The report object is the first position in the ordinal, so it is 1. Subreports are one level deeper, so they get their own unique ordinal. For example, if you have three subreports, then they are numbered 1.1, 1.2, and 1.3. Subreports where the business component is a child are assigned successive whole-digit ordinals (2, 3, and so on). Subreports to these are assigned ordinals such as 2.1, 2.2, and so on.

See Also

"Business Component" on page 91
"Report" on page 250

Sub Report Field

Siebel Object Types > Report > Sub Report > Sub Report Field

Defines the field to be exported to or otherwise made available in the subreport.

Properties

Property	Description/Usage	Valid Values/Examples
Field **(R)**	The name of the field to be exported for the subreport.	
Name **(R)**	The subreport's name.	

Business Component Restriction

A subreport field is linked to a parent subreport. These fields must be defined in the business component of the parent subreport. If no field is specified for a subreport, then an error occurs when creating the Actuate library.

See Also

"Report" on page 250
"Sub Report" on page 285

SVF Pick Map

Siebel Object Types > Business Component > Single Value Field > SVF Pick Map

Specifies instructions for copying fields when a picklist is used by a field.

Properties

Property	Description/Usage	Valid Values/Examples
Constrain **(O)**	A TRUE or FALSE value.	TRUE — Defines the pick map as a constraint pick map. FALSE — Defines the pick map as a copy pick map.
Field **(R)**	The field in the parent business component that is connected to the pick map.	
No Clear **(O)**	A TRUE or FALSE value indicating how to set the Field property if no value is selected from the picklist.	FALSE — The Field property is set to NULL. TRUE — The Field property is not set to NULL.
Pick List Field **(O)**	The field in the picklist business component that is connected to the pick map.	

Copy and Constraint Pick Maps

There are two kinds of pick maps: copy pick maps and constraint pick maps. The type of pick map is determined by the value in the Constrain property. If a pick map is a copy pick map, then the value of the field in the picklist business component specified by the Pick List Field property is copied into the field of the parent business component specified by the Field property. If the pick map is a constraint pick map, then the value from the field in the parent business component specified by the Field property is applied as a search specification on the field in the picklist business component specified by the Pick List Field property.

See Also

"Business Component" on page 91
"Field" on page 177
"MVF Pick Map" on page 240
"Pick List" on page 244
"Pick Map" on page 247
"Single Value Field" on page 281
"SVF Pick Map UpdOnlyIfNull" on page 288

SVF Pick Map UpdOnlyIfNull

Siebel Object Types > Business Component > Single Value Field > SVF Pick Map > SVF Pick Map UpdOnlyIfNull

Marks a copy pick map (read “SVF Pick Map” on page 287) to perform the copy operation only if the field specified in the SVF pick map UpdOnlyIfNull is not null. Constrain pick maps are also copy pick maps. In cases where the constraint is active, this results in no operation (no-op). In cases where the UpdOnlyIfNull causes the constraint to be ignored, the copy operation works. To avoid the copy operation use a calculated field as the field for the constrain pick map.

Properties

Property	Description/Usage	Valid Values/ Examples
Field **(R)**	The field in the parent business component to check for the null condition before performing the copy operation specified by the parent pick map.	

See Also

“Business Component” on page 91
“MVF Pick Map UpdOnlyIfNull” on page 241
“Pick List” on page 244
“Pick Map” on page 247
“Pick Map UpdOnlyIfNull” on page 248
“Single Value Field” on page 281
“SVF Pick Map” on page 287

Symbolic String (H)

Siebel Object Types > Symbolic String

Encapsulates a piece of text that can appear anywhere on the Siebel user interface. Any Siebel Object that has a translatable string property (such as the caption of a Control object) can obtain a desired string display value by establishing a reference to the appropriate Symbolic String object through its “String Reference” property (for example, “Caption - String Reference”). The actual string content for a Symbolic String is stored in one or more child Symbolic String Locale objects, each of which represents the value of the string in a different language. For example, the Symbolic String named ‘SBL_ACCOUNTS-0910145428-0HN’ might have two Symbolic String Locale children, one containing the display value for English (‘Account’) and one for German (‘Kunde’).

Properties

Property	Description/Usage	Valid Values/ Examples
Approved **(O)**	For Oracle use only.	
Current String Value	Calculated field indicating what the actual display value exposed to the UI would be if a compile were performed using the active Siebel Tools language mode and the corresponding compiled SRF file used. The calculated value displayed in this field is the 'String Value' attribute of the corresponding Symbolic String Locale child object instance for the active Siebel Tools language mode. For example, if the selected Symbolic String is 'SBL_ACCOUNTS-0910145428-0HN' and the Tools language mode is set to German, then 'Kunde' (and not 'Accounts') is displayed in this field in Tools. Switching the Tools language mode to English changes the calculated value to 'Accounts'. Although this property is a calculated field, it is editable. Modifying this value modifies the String Value property of the Symbolic String Locale whose language corresponds to the active Tools language mode.	Accounts, Kunde (the German translation for the word 'Accounts')
Definition **(O)**	Text definition of the Symbolic String.	State: U.S. (or other country) sub-national geographic entity
Glossary **(O)**	For Oracle use only.	
Name **(R)**	The name of the Symbolic String. All references to the Symbolic String are made using this name, which must be unique among Symbolic Strings. The names of all Symbolic Strings delivered with the Siebel application begin with "SBL_" and contain an uppercase version of the ENU string value for the Symbolic String. For example, the ENU value for the Symbolic String named "SBL_ACCOUNTS-0910145428-0HN" is "Accounts".	SBL_ACCOUNTS-0910145428-0HN
Type **(O)**	A text field that can be used as desired to categorize Symbolic Strings into groups or families. All Symbolic Strings delivered with the Siebel application were produced through an internal conversion process, and as such are marked with the type "Conversion".	"Conversion", "Private"

See Also

"Symbolic String Locale" on page 291

Symbolic String Locale

Siebel Object Types > Symbolic String (H) > Symbolic String Locale

Child object of Symbolic String that defines the language-specific display values for a particular Symbolic String instance. For example, for the parent symbolic string that means ‘Account’, there may be multiple entries, one for each language.

Properties

Property	Description / Usage	Valid Values / Examples
Language **(R)**	Language of the particular display value.	ENU, DEU
Parent Symbolic String **(R)**	Used to associate string display values stored in the Symbolic String Locale with their parent Symbolic String.	Valid values: any existing Symbolic String name attribute. Example: SBL_#_OF_CONCURRENT_LICENSES-1004225906-00M (any string value is allowed).
String Value **(R)**	Actual string literal value that is displayed in the UI for a particular language.	Accounts’, ‘Kunde’

See Also

“Symbolic String (H)” on page 289

System Activity Object

Siebel Object Types > System Activity Objects

Defines the properties of a business component that are used in generating activities by certain commands such as Send Mail and Send Page.

Properties

Property	Description/Usage	Valid Values/Examples
Account Id Field **(O)**	The field that contains the row ID of the account that should be associated with the generated activity.	
Business Component **(R)**	The business component for which the system activity properties are defined.	Any business component that contains information useful for activity generation. For example, if a System Activity Object is defined on the Service Request buscomp, information in the service requests records such as account Id and contact Id can be used in the generated activities when sending communications.
Contact Id Field **(O)**	The field that contains the row ID of the contact that should be associated with the generated activity.	
Defect Id Field **(O)**	A field that contains a defect row ID value that can be used to populate the Activity Defect Id field of the new activity.	
Opportunity Id Field **(O)**	The field that contains the row ID of the opportunity that should be associated with the generated activity.	
Service Request Id Field **(O)**	The field that contains the row ID of the service request that should be associated with the generated activity.	
Source Id Field **(O)**	A field that contains a campaign row ID value that can be used to populate the Campaign Id field of the new activity.	

Table

Siebel Object Types > Table

Creates a new table to store additional attributes.

Properties

Property	Description/Usage	Valid Values/Examples
Abbreviation 1-6 **(O)**	Specifies up to 6 alternative abbreviations for the table. The abbreviations are used by the EIM Generator/Wizard in determining how to name interface table-columns used in foreign key mappings and should not be modified by customers.	
Alias **(O)**	Not used.	Not applicable.
Append Data **(O)**	A TRUE or FALSE value. Specifies to the database that new rows are to be inserted at the end of the table and cannot be turned on if there is a clustered index on that table.	TRUE — New rows are inserted at the end of the table.
Base Table Name **(R)**	If the Type property is Ext or Journal, the name of the table this table is extending or journaling.	
Dflt Mapping Col Name Prefix **(S)**	The default name prefix to use when generating interface table column names. For Oracle use only.	
External API Write	Boolean property used to indicate whether or not inserts, updates, or deletes to external tables should be handled by an external API.	If this property is set to TRUE, the BusComp_PreWriteRecord should be scripted to publish the insert, update, or delete to an external API.
File **(O)**	A TRUE or FALSE value.	TRUE indicates that the table is used to store file attachment information.
Group **(O)**	Solely used to set table groups for IBM System 390. Tables are primarily placed into groups that correspond to dock objects, with further differentiation taken into account for table size, access frequency, and so on.	

Property	Description / Usage	Valid Values / Examples
Key Generation Business Service	The Business Service used to generate the primary key (Id field) for the Business Component. If this is not specified, the default row id generator is used for tables that have a configured column corresponding to the “Id” system field.	
Key Generation Service Method	The name of the Business Service method to be invoked when generating the primary key of the Business Component.	
Name **(R)**	The unique name of the table.	
Owner Organization Specifier	Information not available.	
Parent Table Column1 Name **(O)**	The column that points to the primary parent if this is a child or intersection table.	
Parent Table Column2 Name **(O)**	The column that points to the second parent if this is an intersection table.	
Status **(O)**	The current status of a table. States whether tables from previous versions of Siebel business applications can be used in the most recent version of the application.	ACTIVE — Can still be used in latest version of the application. INACTIVE — Discontinued and not supported in the latest release. Extension columns that reside on inactive tables must be moved to alternate tables. Fro example, S_EMPLOYEE should be moved to S_CONTACT, S_USER, or S_EMP_PER. EOL (End of life) — Not used in the latest release. The table should be deleted because it is no longer used. For example, the old S_*_IF tables. NOT USED — Not used in latest release by the application, but you may continue to use the table. For example, table can be used as is for docking or EIM.

Property	Description/Usage	Valid Values/Examples
Type **(S)**	A description of what the table is used for.	
User Name **(R)**	The name for the table.	
Volatile **(O)**	A TRUE or FALSE value.	TRUE — Specifies to the database not to use the statistics produced by runstats for that table.

See Also

"Column" on page 130
"Data Source (H)" on page 155
"Index" on page 200
"User Key" on page 322

Task

Siebel Object Types > Task

The Task object is an executable version of an organization's customer-centric business process. It provides a wizard-like user interface that guides the user through task execution, allows navigation both back and forward within task execution, and enables task execution to be paused and resumed. A task consists of one or more task steps such as start steps, subtasks, decision points, and business service steps.

Properties

Property	Description/Usage	Valid Values/Examples
Allow Pause **(R)**	Flag that indicates if the task instance can be paused and resumed.	TRUE — the task instance can be paused. FALSE — the task instance cannot be paused.
Business Object **(R)**	The name of the associated business object.	The value for this property is chosen from a drop-down list of business objects. Only business objects with a defined primary component appear in this drop-down list.
Description **(O)**	A text narrative describing the purpose of the task.	

Property	Description/Usage	Valid Values/Examples
Display Name **(R)**	Name that is displayed in the Current/Context Task pane. Display Name is shown in the pane unless the Task Group has an Override Display Name set for the Task.	The value for this property is determined by the Display Name - String Reference property, which is selected from a drop-down list of available translatable strings. For more information about the String Reference property, see "Properties with Consistent Meanings" on page 16.
Inbox Disposition **(R)**	Flag that indicates whether an inbox item will be left in inbox after the task instance completes.	Save A Record — Always keep the inbox item after the task instance completes. Discard Record — Always delete the inbox item after the task instance completes. Do Not Care (Fastest) — This option minimizes the number of database operations required to maintain the task inbox items. An inbox item is created upon task completion only if the task has never been paused before.
Is Subtask **(R)**	Flag that indicates if the task is a main task or a subtask.	TRUE — The task is a subtask. FALSE — The task is a main task.
Name **(S)**	The name of the object. The system sets the value of this property using the value of the Task Name property.	
On Conflict **(R)**	Flag that controls conflict resolution behavior when a conflict is detected while writing data to the database. For information on conflict resolution schemes, see the *Siebel Business Process Framework: Task UI Guide*.	Cancel Operation — Abort write and return errors to the user. Continue Operation — Resolve conflict using assigned resolution scheme, and continue write.
Status **(S)**	The current status of the task.	In Progress — editing in progress. Completed — editing completed and ready for activation. Not In Use — not in use and can be deleted.

Property	Description/Usage	Valid Values/Examples
Task Name **(R)**	The internal name of the task that the task controller uses to identify the task.	
Transactional **(R)**	Flag that indicates whether task execution is transactional.	TRUE — task execution is transactional. FALSE — task execution is not transactional.
Transient BC **(O)**	The name of the transient business component associated with the task.	This value is selected from a drop-down list of available transient BCs.
Version **(S)**	The version number of the task.	Read-only. The default version is 0.

See Also

"Task Branch" on page 297
"Task Chapter" on page 300
"Task Event" on page 301
"Task Group" on page 303
"Task Locale" on page 305
"Task Metric" on page 305
"Task Property" on page 306
"Task Step" on page 307

Task Branch

Siebel Object Types > Task > Task Branch

Connects two task steps. The Task Branch controls the data and process flow of a task.

Properties

Property	Description/Usage	Valid Values/Examples
Label **(O)**	Text to be displayed in the task flowchart as a label on the branch.	
Name **(R)**	The name of the branch.	

Property	Description / Usage	Valid Values / Examples
Parent Name **(S)**	The name of the task step that the branch belongs to.	
Type **(R)**	The type of the branch.	Condition — Indicates that a condition is defined for the branch. Default — Indicates that if nothing else is satisfied, this branch is followed. Additionally, conditions defined for the branch are ignored. Connector — Indicates that there is no condition branching involved. Error Exception — Use to define exception handling. This connector type captures system errors, such as an error noting that the Assignment Manager server component is not available. User Defined Exception — Use to define exception handling. This connector type captures user-defined errors, such as an error noting that an order being submitted is incomplete.

See Also

"Task" on page 295
"Task Branch Criteria" on page 298
"Task Branch Criteria Value" on page 299
"Task Property" on page 306
"Task Step" on page 307

Task Branch Criteria

Siebel Object Types > Task > Task Branch > Task Branch Criteria

The Task Branch Criteria object type specifies a list of conditions. At runtime, the task controller evaluates the conditions and when the outcome is true, the task takes the path defined by the branch.

Properties

Property	Description/Usage	Valid Values/Examples
Applet Field Name	The name of the field within the named applet. This is a required field when the Compare To value is set to Applet.	The drop-down list displays the fields defined for the selected applet.
Applet Name	Name of the applet to compare. This is a required field when the Compare To value is set to Applet.	The drop-down list displays the available applets.
Business Component	The name of the business component within the business object of the task. This is a required field when the Compare to value is set to Business Component.	The drop-down list displays the business components defined for the business object selected for the task.
Business Component Field	The name of a field within the business component. This is a required field when the Compare To value is set to Business Component.	The drop-down list displays the fields defined for the selected business component.
Compare To **(R)**	Indicates where the comparison value is coming from.	Applet, Business Component, Expression, and Task Property.
Name **(S)**	Name of the object.	
Operation **(R)**	Identifies the comparison operation.	For a description of the available comparison operations, see the section on building expressions in the *Siebel Business Process Framework: Workflow Guide*.
Parent Name **(S)**	The name of the parent branch.	
Property Name **(R)**	Identifies the specific task property on which to base the condition.	The drop-down list displays the task properties of the task.

See Also

"Task" on page 295
"Task Branch" on page 297
"Task Branch Criteria Value" on page 299
"Task Property" on page 306
"Task Step" on page 307

Task Branch Criteria Value

Siebel Object Types > Task > Task Branch > Task Branch Criteria > Task Branch Criteria Value

This object type is for internal use only.

Task Chapter

The Task Chapter object type defines a logical grouping of task steps. When a task step is executed, the task step name and its associated chapter name are displayed in the Current Task Pane. Task chapters are displayed in the order defined by the sequence property.

Properties

Property	Description/Usage	Valid Values/ Examples
Color **(R)**	Color in which the chapter's task steps is displayed in the task editor designer canvas when Show Chapter mode is turned on. The color can be set only in the Multi Value Property Window of the parent task object.	
Display Name **(O)**	Chapter name that is displayed in the Current Task pane. If left blank, the chapter is not displayed.	
Name **(R)**	Name of the chapter. Chapter names should be unique within a task.	
Parent Name **(S)**	The name of the task that the chapter belongs to.	
Sequence **(R)**	Chapter sequencing number. This property defines the order in which this chapter is displayed in the current task pane with respect to other chapters with the same parent task object.	Non-negative integer values.

See Also

"Task" on page 295
"Task Chapter Locale" on page 300

Task Chapter Locale

Represents language-specific overrides used with the Task Chapter object type.

Properties

Property	Description/Usage	Valid Values/ Examples
Display Name **(O)**	Name displayed for a task chapter for a specific language.	
Name **(R)**	The abbreviation of the language being used by the application.	Example: ENU.

See Also

"Task" on page 295
"Task Chapter" on page 300

Task Event

Siebel Object Types > Task > Task Event

Task Event defines event handlers for certain task events (such as Cancel, Delete, Pause, and Resume). The Task Event defines the actions to take when such task events occur.

Properties

Property	Description/Usage	Valid Values/Examples
Business Service Method	The name of the method to invoke on the service. This is a required field when Business Service Name is specified.	The drop-down list displays methods defined for the selected business service.
Business Service Name	The name of the service to invoke. This is a required field when Workflow Process is not specified.	The drop-down list displays business services available in Siebel Tools with the Hidden flag set to FALSE.
Name **(R)**	The name of the task event.	Cancel, Delete, Pause, Resume
Parent Name **(S)**	The name of the parent task.	
Workflow Process	The name of the workflow process to invoke. This is a required field when Business Service Name is not specified.	The drop-down list displays available workflow processes.

See Also

"Task" on page 295
"Task Event IO Argument" on page 302
"Task Property" on page 306

Task Event IO Argument

Task Event IO Argument defines the input arguments to the event handlers. The object is used for configuring both the input and output arguments of the event handler. When the Input/Output flag is set to Input, an input argument name of the event handler, whichever type being configured in the parent Task Event object, should be entered into the Argument property. When the Input/Output flag is set to Output, an output argument name of the event handler should be entered into the Argument property.

Task Event defines event handlers for certain task events, i.e., Cancel, Delete, Pause, and Resume. In other words, Task Event defines what actions to take when such task events occur.

Properties

Property	Description/Usage	Valid Values/Examples
Argument **(O)**	The name of the output argument of the Workflow or Business service called by the event handler. This property should be left empty for input arguments.	
Business Component	The name of the business component within the business object of the task. This is a required field when the Type is set to Business Component.	
Business Component Field	The name of a field within the business component. This is a required field when the Type is set to Business Component.	The dropdown list displays all fields defined for the selected business component.
Input/Output **(R)**	Flag that indicates whether the argument is an input argument or an output argument.	Input — input argument Output — output argument
Name **(S)**	For input arguments, specify the input argument name of the business service method. For output arguments, this property should be left empty. A unique name is generated by the system.	If a Workflow process is chosen as the event handler, the value of this property should be either RowId or ProcessName.
Parent Name **(S)**	Name of the parent Task Event object.	
Preferred Sequence **(O)**	Evaluation sequence of the argument. Currently, evaluation sequence is not supported for this object, and the property is being ignored.	Non-negative integer values.

Property	Description/Usage	Valid Values/Examples
Property Name	The name of the task property that is referenced in the argument. This is a required field when the Type is set to Task Property.	The drop-down list includes all task properties of the task.
Type **(R)**	The type of the argument.	Business Component, Literal, Expression, Task Property
Value **(O)**	This property stores a literal value or expression when the Type is set to Literal or Expression.	

See Also

"Task" on page 295
"Task Event" on page 301

Task Group

Siebel Object Types > Task > Task Group

The Task Group object represents a collection of tasks, based on a common business component that can be displayed on the Task Pane. For more information about configuring task groups, see the *Siebel Business Process Framework: Task UI Guide*.

Properties

Property	Description/Usage	Valid Values/Examples
Business Component **(R)**	The common business component shared by the tasks in this task group.	
Display Name **(R)**	Title under which this group of tasks is displayed in the Task Pane.	

See Also

"Task" on page 295
"Task Group Item" on page 303
"View Task Group" on page 332

Task Group Item

Siebel Object Types > Task > Task Group > Task Group Item

Specifies a task and the sequence in which it is shown within a task group.

Properties

Property	Description / Usage	Valid Values / Examples
Sequence	Specifies the order in which the task is displayed within the parent task group.	The value of this property is a numerical value.
Task **(R)**	Specifies the task to display in the Task Pane, which can be launched by the user. Adding a task to the Task Group Item does not mean that it is shown in the Task Pane. Logic is applied to ensure the following: ■ The Task Group's Application matches the running application. ■ The user has access to run this task, as specified in Task Administration. ■ If "Context Required" is set to true, the task's business component appears in the current view. ■ The Task is licensed for this module.	

See Also

"Task" on page 295
"Task Group" on page 303

Task Group Locale

Siebel Object Types > Task > Task Group > Task Group Locale

Represents language-specific overrides used with the Task Group object type.

Properties

Property	Description / Usage	Valid Values / Examples
Display Name **(O)**	Name displayed for a Task Group for a specific language.	
Name **(R)**	The abbreviation of the language being used by the application.	Example: ENU.

See Also
"Task" on page 295
"Task Group" on page 303
"Task Group Item" on page 303

Task Locale

Siebel Object Types > Task > Task Locale

Represents language-specific overrides used with the Task object type.

Properties

Property	Description/Usage	Valid Values/Examples
Display Name **(O)**	Name displayed for a Task for a specific language.	
Name **(R)**	The abbreviation of the language being used by the application.	Example: ENU.

See Also
"Task" on page 295

Task Metric

Siebel Object Types > Task > Task Metric

Defines metrics to be collected during task execution.

Properties

Property	Description/Usage	Valid Values/Examples
Metric Name **(R)**	The name of the metric.	The dropdown list displays a list of pre-defined metrics.
Name **(R)**	The name of the object (pre-defaulted to the Metric Name by system).	
Property Name **(R)**	The name of the task property that the metric is mapped to.	The dropdown list displays a list of task properties in the task.

See Also
"Task" on page 295
"Task Property" on page 306

Task Property

Siebel Object Types > Task > Task Property

A Task Property defines a field to store a value that you can use in task steps, either as input and output arguments, or for performing evaluations.

Properties

Property	Description/Usage	Valid Values/Examples
Access Mode **(R)**	Defines how the task property can be accessed during task execution.	Read — task property is read only. R/W — task property can be read and updated.
Data Type **(R)**	The type of data that can be stored.	Binary — For variant or binary information. Binary types must be assigned the None value in the In/Out field. Date — For dates. Hierarchy — Data type used by Enterprise Application Integration (EAI) to store data from a property set. Integration Object — Data type used by Enterprise Application Integration (EAI) to store data from an integration object. Number — For numeric data. String — For alphanumeric data. Strongly Typed Integration Object — Data type used by Enterprise Application Integration (EAI) to store data from a strongly typed integration object.
Default **(O)**	Pre-default value of the task property.	

Property	Description/Usage	Valid Values/Examples
In/Out **(R)**	Describes whether or not the task property is passed in or out of the task, passed into the task and returned, or used only within the task.	In — The task property is passed into the task. (Binary types cannot be assigned this value.) Out — The task property is passed out of the task. (Binary types cannot be assigned this value.) In/Out — The task property is passed into the task and returned. (Binary types cannot be assigned this value.) None — The task property is used only within the task.
Integration Object	The name of the Integration Object. This field is required when the Data Type is Integration Object or Strongly Typed Integration Object	Account - Get Oracle Customer (Oracle).
Name **(R)**	The name of the task property.	

See Also

"Task" on page 295
"Task Branch" on page 297
"Task Branch Criteria" on page 298
"Task Event" on page 301
"Task Metric" on page 305
"Task Step" on page 307

Task Step

Siebel Object Types > Task > Task Step

Task Step represents a unit of work done by the task controller. A sequence of Task Steps forms a functioning task.

Properties

Property	Description / Usage	Valid Values / Examples
Business Component	The name of the business component within the business object of the task. This is a required field when the Type is set to Siebel Operation.	The dropdown list displays business components defined for the business object selected for the task.
Business Service Method	The name of the method to invoke on the service. This is a required field when the Type is set to Business Service.	The drop-down list displays methods defined for the selected business service.
Business Service Name	The name of the service to invoke. This is a required field when the Type is set to Business Service.	The drop-down list displays business services existing in Siebel Tools with the Hidden flag set to FALSE.
Chapter **(O)**	The chapter that the step belongs to.	The drop-down list displays chapters in the task.
Defer Write Record **(O)**	Defines whether data will be validated before they are written to the database in an insert Siebel operation. This field is applicable only when the Type is set to Siebel Operation.	TRUE — data will not be validated FALSE — data will be validated
Description **(O)**	A text narrative describing the purpose of the step.	
Disable Cancel	Defines whether the Cancel button is disabled at a task view. This is a required field when the Type is set to Task View.	TRUE — Cancel button disabled FALSE — Cancel button enabled
Disable Pause	Defines whether the Pause button is disabled at a task view. This is a required field when the Type is set to Task View.	TRUE — Pause button disabled FALSE — Pause button enabled
Disable Previous	Defines whether the Previous button is disabled at a task view. This is a required field when the Type is set to Task View.	TRUE — Previous Button disabled FALSE — Previous button enabled
Display Name **(O)**	Name that is displayed in the Current/ Context Task pane.	This value is selected from a drop-down list of available translatable strings.

Property	Description / Usage	Valid Values / Examples
Display Name Type **(O)**	Defines whether the display name of a task view can be repeated right next to each other in the Current/Context Task pane.	Normal — display name is repeated Unique — display name is displayed only once
Error Code **(O)**	The name of the error code.	This value is selected from a drop-down list.
Forward Button Type **(R)**	The label on the forward button.	Normal — "Next" is shown on the forward button. Submit — "Submit" is shown on the forward button. Finish — "Finish" is shown on the forward button
Name **(S)**	The name of the Object.	
Operation	The type of the Siebel Operation. This is a required field when the Type is set to Siebel Operation.	Delete, Insert, Next Record, Previous Record, Query, QueryBiDirectional, Update
Repeatable	Flag that indicates whether the step can be executed more than once during the task execution. This is a required field when the Type is set to Business Service or Siebel Operation.	TRUE — step can be executed more than once FALSE — step is executed only at the first time it is encountered.
Retain Applet Search Spec	Flag that indicates whether applet search specs that are defined and executed in the current step are retained when the current step exits. This is a required field when the Type is set to Task View.	TRUE — applet search specs are retained FALSE — applet search specs are cleared
Retain Task Search Spec	Flag that indicates whether task search specs that are defined and executed in the current step are retained when the current step exits. This is a required field when the Type is set to Siebel Operation or Task View.	TRUE — task search specs are retained FALSE — task search specs are cleared

Property	Description / Usage	Valid Values / Examples
Retain User Search Spec	Flag that indicates whether user search specs that are executed in the current step are retained when the current step exits. This is a required field when the Type is set to Task View.	TRUE — user search specs are retained FALSE — user search specs are cleared
SubTask Name	The name of the subtask. This is a required field when the Type is set to Subtask.	This value is selected from a drop-down list.
Task View	The name of the task view. This is a required field when the Type is set to Task View.	This value is selected from a drop-down list.
Type **(S)**	The type of the step.	Business Service — invokes a business service Commit — commits data to database Decision Point — performs decision branching End — marks the end of task execution Error — abort task execution with error Siebel Operation — performs Siebel operation Start — marks the start of task execution Subtask — invokes a subtask Task View — displays a task view

See Also

"Task" on page 295
"Task Chapter" on page 300
"Task Property" on page 306
"Task Step Context" on page 311
"Task Step IO Argument" on page 312
"Task Step Locale" on page 313

Task Step Context

Defines the context for the step by setting up search specifications for business components in the task business object. This object should only be created for steps of Siebel Operation and Task View types.

Properties

Property	Description / Usage	Valid Values / Examples
Expression Business Component **(O)**	The name of the business component referenced in the right operand of the expression. For example, the expression "[Account Id] = [Id]" has an operator "=", a left operand "[Account Id]", and a right operand "[Id]". If the Expression Business Component is set to "Account", the right operand will be effectively treated as [Account.Id].	The dropdown list includes business components defined for the business object selected for the task.
Filter Business Component **(O)**	The name of the business component referenced in the left operand of the expression. For the same example as above, "[Account Id] = [Id]", if the Filter Business Component is set to "Contact", the left operand will be effectively treated as [Contact.Account Id]. As a result, the expression itself becomes "[Contact.Account Id] = [Account.Id]".	The dropdown list includes business components defined for the business object selected for the task.
Is User Search Spec **(R)**	Flag that indicates whether the search spec defined is a user search spec or a task search spec.	TRUE — user search spec FALSE — task search spec
Search Specification **(O)**	The value of the search specification.	
Type **(R)**	The type of the context.	Literal — literal value Expression — expression

See Also

"Task" on page 295
"Task Property" on page 306
"Task Step" on page 307
"Task Step IO Argument" on page 312
"Task Step Locale" on page 313

Task Step IO Argument

Identifies the input and output arguments for a task step. Each task step accepts input from the task and passes output back to the task.

Properties

Property	Description/Usage	Valid Values/Examples
Argument **(R)**	The name of the argument.	The drop-down list includes input and output arguments of the selected business service.
Business Component	The name of the business component within the business object of the task. This is a required field when the Type is set to Business Component.	The drop-down list includes business components defined for the business object selected for the task.
Business Component Field	The name of a field within the business component. This is a required field when the Type is set to Business Component.	The drop-down list displays the fields defined for the selected business component.
Input/Output **(R)**	Flag that indicates whether the argument is an input argument or an output argument.	Input — input argument Output — output argument
Name **(S)**	The name of the object.	
Preferred Sequence **(O)**	Evaluation sequence of the argument. When the step is executed, an argument with a lower preferred sequence is evaluated before an argument with a higher preferred sequence. Currently, only the input argument type has sequencing support. Evaluation sequence of other step and argument types is ignored.	Non-negative integer values.
Property Name	The name of the task property that is referenced in the argument. This is a required field when the Type is set to Task Property.	The drop-down list includes all task properties of the task.
Type **(R)**	The type of the argument.	Business Component, Literal, Expression, Output Argument, Task Property
Value **(O)**	This property stores a literal value or expression when the Type is set to Literal or Expression.	

See Also
"Task" on page 295
"Task Property" on page 306
"Task Step" on page 307
"Task Step Context" on page 311
"Task Step Locale" on page 313

Task Step Locale

Siebel Object Types > Task > Task Step > Task Step Locale

Represents language-specific overrides used with the Task Step object type.

Properties

Property	Description / Usage	Valid Values / Examples
Display Name **(O)**	Name displayed for a task step for a specific language.	
Name **(O)**	The abbreviation of the language being used by the application.	Example: ENU.

See Also
"Task" on page 295
"Task Step" on page 307

Text Style (H)

Siebel Object Types > Text Style (H)

Defines a named object that specifies font-related information. Several user interface objects including controls, applets, views, and applications have a Text Style property that refers to one of the named text styles.

Properties

Property	Description/Usage	Valid Values/ Examples
Bold **(O)**	A TRUE or FALSE value.	TRUE — Bold text.
Font Face **(O)**	The font face name, such as Arial, Courier, and so forth. If this property is not specified, the default font for that object is used. For example, if a control uses a text style that does not specify a font face, the font face used by all controls in that applet is used.	
Font Size **(O)**	The integer point size of the font to be used. If this property is not specified, the default font size for that object is used. For example, if a control uses a text style that does not specify a font size, the font size used by all controls in that applet is used.	
Italic **(O)**	A TRUE or FALSE value.	TRUE — Italic text.
Name **(R)**	The name of the text style.	
Underlined **(O)**	A TRUE or FALSE value.	TRUE — Underlined text.

See Also

"Hidden Object Types and Properties" on page 21

Toolbar

Siebel Object Types > Toolbar

Provides a named toolbar that the user can activate or deactivate in Siebel applications and to which command icons (toolbar item object definitions) can be associated or removed in Siebel Tools or Siebel applications.

Typical toolbar functionality for most applications is implemented using HTML toolbars. In an HTML toolbar, the buttons are static images. The buttons may be dimmed to reflect unavailability.

In contrast, some applications, such as Call Center, require toolbar icons that can be altered in response to events, such as blinking when a call is incoming. In this case, Java toolbars are required.

Properties

Property	Description/Usage	Valid Values/ Examples
Class **(O)**	For a Java toolbar, specifies the name of the Java class that implements the toolbar. Left blank for an HTML toolbar.	
Client Restriction **(O)(H)**	Toolbar is restricted to a certain client.	Example: Handheld client.
Display Name **(O)**	Used for the History button. Also used to show and hide toolbars by name.	
Name **(R)**	The name of the toolbar. This name appears in Siebel applications in the list in the Customize dialog box (Toolbars tab).	
Popup Indicator Bitmap **(O)**	Name of the bitmap icon to display in the toolbar.	

See Also

"Command" on page 135
"Hidden Object Types and Properties" on page 21
"Toolbar Item" on page 315

Toolbar Item

Siebel Object Types > Toolbar > Toolbar Item

Associates a command object definition (identified by name as a property in the Command property) with a toolbar object definition (the parent of the toolbar item). This association places a toolbar icon, whose bitmap image, invoked method, and target are specified in the command object definition, on the specified toolbar in a given location (relative to the other toolbar icons on that toolbar).

Properties

Property	Description/Usage	Valid Values/Examples
Command **(R)**	Name of the command object definition that is to provide the bitmap, method, and target for the toolbar item. One or more hyphens can be specified instead of the name of a command object to tell the Siebel application to insert a separator there between icons.	
Display Name **(O)**	Text name displayed on a button on the toolbar.	Example: The text Advanced Search is displayed on the button instead of a bitmap image representing advanced searching.
Group **(O)**	The group name. Makes all buttons belonging to the same group work as radio buttons (that is, only one button can be pushed at a time).	
HTML Attributes **(O)**	Reserved for future use.	
HTML Type **(O)**	Identifies the type of control to be displayed in the toolbar in the Web browser.	
HTML Width **(O)**	The width of the toolbar in pixels in the HTML thin client.	
HTML Width - Language Override **(O)**	Provides the ability to change the width of the toolbar in pixels in the HTML thin client when a particular language/locale is active.	
Name **(R)**	Name of the toolbar item.	
Position **(O)**	Specifies the position of the toolbar item.	
Position - Language Override **(O)**	Provides the ability to change the position of the toolbar item when a particular language/locale is active.	
Target View Frame **(O)**	The HTML page loaded into the frame that is specified by this property.	
Type **(R)**	The toolbar item type.	Valid values are Button, Combo Box, Edit, Label, and Separator.

Property	Description/Usage	Valid Values/Examples
Width **(R)**	Width of the toolbar item.	
Width - Language Override **(O)**	Provides the ability to change the width of the toolbar item when a particular language/ locale is active.	

User Override

The placement of a toolbar icon on a toolbar and the order of that icon relative to the other icons are only established as defaults for Siebel applications by the specification of toolbar item properties in Siebel Tools. The user can choose to override these placements; the user's override settings are stored in a local preferences file and are manipulated through the Customize dialog box in Siebel applications.

See Also

"Command" on page 135
"Toolbar" on page 314

Toolbar Item Locale

Siebel Object Types > Toolbar > Toolbar Item > Toolbar Item Locale

Represents language-specific overrides used with the Toolbar Item object Type.

Properties

Property	Description/Usage	Valid Values/Examples
Display Name **(O)**	Text name displayed on a button on the toolbar.	Example: The text Advanced Search is displayed on the button instead of a bitmap image representing advanced searching.
HTML Width **(O)**	The width of the toolbar in pixels in the HTML thin client.	
Name **(O)**	The abbreviation of the language being used by the application.	Example: ENU.
Position **(O)**	Specifies the position of the toolbar item.	
Width **(O)**	Width of the toolbar item.	

See Also

"Toolbar" on page 314
"Toolbar Item" on page 315

Toolbar Locale

Siebel Object Types > Toolbar > Toolbar Locale

Represents language-specific overrides used with the Toolbar object type.

Properties

Property	Description/Usage	Valid Values/Examples
Display Name **(O)**	Text name displayed on a button on the toolbar.	Example: The text Advanced Search is displayed on the button instead of a bitmap image representing advanced searching.
Name **(O)**	The abbreviation of the language being used by the application.	Example: ENU.

See Also

"Toolbar" on page 314

Tree

Siebel Object Types > Applet > Tree

Defines a tree control in a tree applet. The tree control causes the applet to be a tree applet. A tree applet is used in Siebel applications to create a view where users can navigate hierarchically through a structured list of records. An example of a tree applet is found in the explorer view in the Service Requests screen. The tree object definition basically just provides a name; it is an object definition to which tree nodes can be attached and that itself can be attached to the applet object definition.

Properties

Property	Description/Usage	Valid Values/ Examples
HTML Hierarchy Bitmap **(O)**	Specify the name of the HTML Hierarchy Bitmap object to use for displaying icons on the tree.	
Name **(S)**	Name of the tree object definition. The name is always Tree.	

See Also

"Applet" on page 29
"Tree Node" on page 319
"View" on page 325

Tree Node

Corresponds to one folder that the user sees in a tree applet. It has a label and an applet. The label determines the source of the text displayed next to the folder. The applet specifies the name of the applet that is displayed in the right half of the view when the user opens the folder. The tree node also has a position identifier, which is a set of integers separated by periods, such as 1.1.2, identifying the node's level in the hierarchy and its sequence on its level.

Trees and tree nodes are usually created and edited in the Applet Designer.

Properties

Property	Description / Usage	Valid Values / Examples
Applet **(O)**	When the user clicks on the corresponding node in the tree, the applet opens on the right side. This is typically a list applet, although that is not required. If the Applet property is left blank, nothing opens in the right-hand side of the view, which may be desirable in some situations.	
Bitmap Category **(O)(H)**	Not supported.	Not applicable.
Bitmap Index **(O)(H)**	Integer used to select a bitmap for the document symbol as a subset of a multiimage bitmap. Set this to a value of 5 = document symbol. Used internally.	
Bitmap Index Field **(O)(H)**	Not used.	Not applicable.
Business Component **(O)**	The business component that populates values in this tree node. This is the same as the business component specified in the applet invoked by this tree node.	
Display Name **(O)**	The text that appears next to the folder corresponding to the tree node.	
HTML Closed Bitmap **(O)**	Bitmap object that defines the attributes of a closed folder image. The Bitmap object must belong to the HTML Tree Icons Bitmap Category. This property, when not empty, overrides the Close Bitmap property of the HTML Hierarchy Bitmap object. Used by the Siebel Web Engine.	

Property	Description/Usage	Valid Values/ Examples
HTML Open Bitmap **(O)**	Bitmap object that defines the attributes of an open folder image. The Bitmap object must belong to the HTML Tree Icons Bitmap Category. This property, when not empty, overrides the Open Bitmap property of the HTML Hierarchy Bitmap object. Used by the Siebel Web Engine.	
Label Field **(O)**	The field name in the business component that the Siebel application uses to populate the names of the document object definitions under this tree node.	
Max Child Items **(O)**	An integer that sets a limit on the number of child items that appear under the tree node. An arrow appears to let the user scroll further if desired.	
Name **(R)**	Name of the tree node object definition. Must be unique within the tree object definition.	
Position **(R)**	Pair, triplet, and so on of integer values separated by periods indicating the hierarchical level of the tree node in the tree structure and the sequence of the tree node at its level. The top node is position 1. The first (sequentially) of the first-level subordinates is 1.1. The number of integers provides the level, and the value gives the sequence at that level.	
Recursive **(S)**	A TRUE or FALSE value indicating whether or not the tree is recursive, that is, having only one object type (node), whose children are of the same object type, down to some unspecified number of levels. For example, accounts can have subaccounts, which themselves have subaccounts.	TRUE — The tree is recursive.
Recursive Link **(O)**	The name of the link object definition that describes the relationship within the business component's record structure. Used for recursive trees.	
Root Search Spec **(O)**	A string value that qualifies the top level of the tree, primarily for recursive trees. The Root Search Spec setting identifies how to come up with the list of top-level records. Creates a starting point for the tree.	Example: [Parent Account Id] is NULL.

Property	Description/Usage	Valid Values/ Examples
Selected Bitmap Index **(O)(H)**	An integer indicating which bitmap, in a multiple bitmap file, to display when the record is selected. Use the value 5.	
Selected Bitmap Index Field **(O)(H)**	Not used.	Not applicable.

See Also

"Applet" on page 29
"Hidden Object Types and Properties" on page 21
"Tree" on page 318

Tree Node Locale

Siebel Object Types > Applet > Tree > Tree Node > Tree Node Locale

Represents language-specific overrides used with the Tree Node object type.

Properties

Property	Description/Usage	Valid Values/ Examples
Display Name **(O)**	The text that appears next to the folder corresponding to the tree node.	
Name **(R)**	The abbreviation of the language being used by the application.	Example: ENU.

See Also

"Tree Node" on page 319

Type

Siebel Object Types > Type

A collection of object types in a repository.

CAUTION: Do not modify this object type. Any modification can adversely affect performance and operation.

Properties

Property	Description / Usage
Bitmap Category **(S)**	Not used.
Business Component **(S)**	Name of the business component object definition.
Language Sub Type **(S)**	The name of the Locale-associated object type. For example, Applet Locale object type is associated with the Applet object property. Information is reflected in the Applet object definition.
Name **(S)**	Name of the object type.
No Compile **(S)**	A TRUE or FALSE value. If TRUE, the object definition is not compiled when Compile is selected from the Tools menu in Siebel Tools.
Table Name **(S)**	Name of the table.

User Key

Siebel Object Types > Table > User Key

Object type used by the Siebel Enterprise Integration Manager (EIM). For more information, read *Siebel Developer's Reference*.

Each user key object definition provides a set of attribute columns and related information that specifies how the table's rows can be accessed in a particular EIM scenario.

CAUTION: Do not modify this object type. Any modification can adversely affect performance and operation.

Properties

Property	Description / Usage
Inaccessible **(S)**	A TRUE or FALSE value. If TRUE, the user key is accessible.
Index Name **(S)**	Name of the index.
Name **(S)**	Name of the user key.
Source Interface Table Name **(S)**	Name of the source interface table for the user key.
User Key Type **(S)**	The type of user key of the object definition.

See Also
"Table" on page 292
"Column" on page 130
"Data Source (H)" on page 155
"Index" on page 200
"User Key Attribute" on page 323
"User Key Attribute Tree Node" on page 324
"User Key Column" on page 324

User Key Attribute

Siebel Object Types > Table > User Key > User Key Attribute

Object type used by the Siebel Enterprise Integration Manager (EIM). For more information, read *Siebel Developer's Reference*.

Each user key attribute object definition in the parent user key specifies one in the set of attribute columns that collectively identify rows in the grandparent table.

CAUTION: Do not modify this object type. Any modification can adversely affect performance and operation.

Properties

Property	Description/Usage
Attribute Sequence **(S)**	Position of the attribute column in user key.
Name **(S)**	Name of the attribute column.

User Key Attribute Join

Siebel Object Types > Table > User Key > User Key Attribute > User Key Attribute Join

Object type used by the Siebel Enterprise Integration Manager (EIM). For more information, read *Siebel Developer's Reference*.

The user key attribute join specifies a join operation that can be used to convert a user key attribute that is itself a foreign key to another table into attribute column values in that table.

CAUTION: Do not modify this object type. Any modification can adversely affect performance and operation.

Properties

Property	Description/Usage
Join Column Name **(S)**	Name of the column used for the join.
Join Sequence **(S)**	Integer value signifying in what order the join is used.
Join Type **(S)**	Signifies the type of join being used.
Name **(S)**	Name of the columns being used in the join.

See Also

"Table" on page 292
"User Key" on page 322
"User Key Attribute" on page 323

User Key Attribute Tree Node

Siebel Object Types > Table > User Key > User Key Attribute Tree Node

Object type used by the Siebel Enterprise Integration Manager (EIM). For more information, read *Siebel Developer's Reference*.

CAUTION: Do not modify this object type. Any modification can adversely affect performance and operation.

Properties

Property	Description/Usage
Join Column Name **(S)**	Name of the join column.
Join Sequence **(S)**	Integer value signifying in what order the join is used.
Name **(S)**	Name of the user key.
Previous Tree Node Name **(S)**	Name of the previous tree node.
Sub Node Table Name **(S)**	Name of the subnode table.
User Key Attribute Name **(S)**	Name of the user key attribute.

User Key Column

Siebel Object Types > Table > User Key > User Key Column

Object type used by the Siebel Enterprise Integration Manager (EIM). For more information, read *Siebel Developer's Reference*.

CAUTION: Do not modify this object type. Any modification can adversely affect performance and operation.

Properties

Property	Description / Usage
Column Name **(S)**	Name of the column from the user key index.
Column Sequence **(S)**	Integer value signifying the order of the column in the user key.
Name **(S)**	Same as column name property.

View

Siebel Object Types > View

Defines a visual representation of a business object's data.

Properties

Property	Description/Usage	Valid Values/Examples
Add To History **(O)**	Defines whether or not a view is included in the historical list of views to which a user has navigated to during a session. This history is accessed by clicking the browser's Back button.	TRUE or FALSE.
Admin Mode Flag **(O)**	A TRUE or FALSE value. Admin mode disables all visibility rules. Setting this to TRUE ignores any insert update visibility restrictions on the view. This should only be set for server level administrative views. If the Admin Mode visibility property is set to TRUE, and if pop-up visibility is also set for the business component, then pop-up visibility still takes effect. Also if view is configured in Admin Mode, the business component visibility type is ignored unless it is set to Organization. See also "Admin Mode Flag" on page 329	TRUE — Designates the view as one that operates in Admin mode.
Background Bitmap **(O)(H)**	A bitmap to be used as the background of the entire view area.	
Bitmap Category **(O)(H)**	Not used.	Not applicable.
Business Object **(R)**	The name of a business object that forms the basis of the visual representation.	
Container Web Page **(O)(H)**	The container page to place this view into. Overrides the container web page setting in the Application object.	
Drop Sectors **(R)(H)**	The pair of sectors to drop in six-sector mode for resolutions less than 1024 by 768.	Valid values are 0and4, 1and5, 2and6, and 3and7.
Explicit Logon **(O)**	A TRUE or FALSE value.	TRUE — Requires that the user has explicitly logged on through the logon Web page to see this view.

Property	Description/Usage	Valid Values/Examples
Help Identifier **(O)**	A topic name for context-sensitive help.	The name is a constant character string, all uppercase, with no blanks, as in `ID_VIEW_OPPORTUNITY_LIST`.
HTML Bitmap **(O)**	An icon that can be used as a link to the view.	
HTML Popup Dimension **(O)**	Dimension, in pixels, of the pop-up window.	An example is 640x480 (specified with a small x and without blank spaces).
Name **(R)**	The name of the view. All references to the view are made through its name.	
No Borders **(O)(H)**	A TRUE or FALSE value that controls borders around applets. This can be used in conjunction with a view background bitmap to provide a contiguous canvas look.	TRUE — Causes the user interface to be rendered with no borders around the applets and no spacing between the applets.
Sector0 Applet **(R)(H)**	The applet that appears in sector 0 of the view.	
Sector1 Applet **(R)(H)**	The applet that appears in sector 1 of the view.	
Sector2 Applet **(R)(H)**	The applet that appears in sector 2 of the view.	
Sector3 Applet **(R)(H)**	The applet that appears in sector 3 of the view.	
Sector4 Applet **(R)(H)**	The applet that appears in sector 4 of the view.	
Sector5 Applet **(R)(H)**	The applet that appears in sector 5 of the view.	
Sector6 Applet **(R)(H)**	The applet that appears in sector 6 of the view.	
Sector7 Applet **(R)(H)**	The applet that appears in sector 7 of the view.	

Property	Description/Usage	Valid Values/Examples
Secure **(O)**	A TRUE or FALSE value. If TRUE, the Siebel Web Engine verifies that the current request used the HTTPS protocol. This prevents a client from obtaining access to a secure view by typing HTTP instead of HTTPS into their browser. See also "Admin Mode Flag" on page 329.	The default is FALSE. TRUE — All URLs to the view generated by the Siebel Web Engine specify the HTTPS protocol.
Status Text **(O)(H)**	Not used in this release.	
Text Style **(O)(H)**	Defines the named text style to be used by default by all controls in the applets in the view, if the control and applet do not explicitly specify a text style.	
Thread Applet **(O)**	The name of the applet that contains the field whose data value is used as part of the thread title. The source field is identified by the Thread Field property.	
Thread Field **(O)**	The name of the field of the thread applet whose data value is used as part of the thread title. The source applet is identified by the Thread Applet property.	
Thread Title **(O)**	The text used as the thread title label.	
Title **(O)**	Not used.	Not applicable.
Upgrade Behavior **(R)**	This property is set by Oracle and cannot be modified. Indicates if the object should be ignored if the repository merge is run with the Incorporate Custom Layout option.	Admin or Null Admin indicates that the object should be ignored. Null indicates that the object is to be processed by the Incorporate Custom Layout option.

Property	Description/Usage	Valid Values/Examples
Vertical Line Position **(O)(H)**	Determines the relative position of the dividing line between applets in sectors 0 to 3 versus those in sectors 4 to 7. Primarily applicable for tree applets. This property is especially useful for views that contain tree applets; it is desirable for the tree applet to occupy only about one-third of the width of the application. Therefore, this setting should be in the 4 to 8 range; otherwise, one applet or the other is likely to be distorted by stretching or compression.	Valid value range is between 0 and 12. A value of 0 is all the way to the left, 12 is all the way to the right, and 6 is in the middle. The default is 6.
Visibility Applet **(O)**	Identifies which of the applets in the view is the one where visibility is controlled. Usually this is the list (upper) applet in a list-form view or the master applet in a master-detail view. The Visibility Applet property must be set to the master applet of the given view. This property is used in conjunction with Visibility Applet Type. For more information, read "Visibility Settings" on page 330.	
Visibility Applet Type **(O)**	The visibility mode that is applied to the applet indicated in the Visibility Applet property. This property can be overridden by the View Web Template's Applet Visibility Type property. For more information, read "Visibility Settings" on page 330.	Valid values are All, Catalog, Group, Manager, Organization, Personal, Sales Rep, and Sub Organization.

Admin Mode Flag

If the view is in Admin mode, then the following restrictions for the business components used by the applets of the view are ignored: no insert, no merge, and no update. However, if these restrictions are set at the applet level rather than the business component level, or if a field has a Read Only value of TRUE, the Admin mode does not circumvent them.

In Admin Mode the user can display every record, even those with no primary team member designated. This is distinct from All visibility, which shows all records that have a primary team member designated. The Account Administration view, Opportunity Administration view, and Product Administration view are examples of Admin mode views.

A conflict may arise on a view with the Admin Mode property set to True if the underlying business component has visibility configured on a Multi Value Link (MVL) and Multi Value Field (MVF) and either Force Active set at the business component level or Force Active set on the visibility MVF.

Secure

The implementation of HTTPS is external to the Siebel Web Engine. HTTPS is negotiated by the browser and the Web Server. The Siebel Web Engine only specifies that HTTPS be used for a particular view. Therefore, any server that is expected to provide secure views must have HTTPS enabled.

Visibility Settings

A view not set up with visibility settings (the properties Visibility Applet and Visibility Applet Type are blank) means the following:

When navigating from another view to this view without visibility settings using the Show drop-down list, the queries context and record focus are kept from the previous view, assuming that the same business components and search specifications are relevant.

However, when navigating to a view without visibility settings using the Site Map or the Screen Tabs, the view behaves as if the most restrictive visibility would apply to this view. (Normally this means Personal or Sales Rep visibility, depending on the corresponding visibility settings of the business component of the view's Thread Applet property.)

See Also

“Screen View” on page 259
“View Report” on page 331
“Hidden Object Types and Properties” on page 21

View Locale

Siebel Object Types > View > View Locale

Represents language-specific overrides used with the View object type.

Properties

Property	Description/Usage	Valid Values/Examples
Name **(R)**	The abbreviation of the language being used by the application.	Example: ENU.
Status Text **(O)**	Not used in this release.	
Thread Title **(O)**	The text used as the thread title label.	
Title **(O)**	The text string used in the window title, following the prefix text specified in the CFG file. The prefix text for the view name in the title bar is obtained from the CaptionPrefix argument in the CFG file, if present. If CaptionPrefix is not specified but ApplicationTitle is, the value in ApplicationTitle is used. If neither value is specified, “Siebel applications” is used as the prefix text.	Example: If the text in this property is My Accounts, and the prefix text is Siebel Sales, the title bar of the Siebel application window reads Siebel Sales - My Accounts when this view is active.

See Also

“Hidden Object Types and Properties” on page 21
“View” on page 325

View Report

Siebel Object Types > View > View Report

Indicates which reports are available when using a particular view.

Properties

Property	Description/Usage	Valid Values/ Examples
Name **(R)**	The name within this view.	
Report **(R)**	The name of the report to include for the view.	
Sequence **(O)**	The sequence number. This number determines the order of the report in the Reports menu.	
Status Text **(O)**	Not used.	Not applicable.

See Also
"Report" on page 250
"View" on page 325

View Report Locale

Siebel Object Types > View > View Report > View Report Locale

Represents language-specific overrides used with the View Report object type.

Properties

Property	Description/Usage	Valid Values/ Examples
Name **(R)**	The abbreviation of the language being used by the application.	Example: ENU.
Status Text **(O)**	Not used.	Not applicable.

See Also
"View Report" on page 331

View Task Group

Siebel Object Types > View > View Task Group

Lists which task groups and their children tasks to display in the Task Pane, when the current view is shown. For more information about configuring the task groups for the Task Pane, see the *Siebel Business Process Framework: Task UI Guide*.

Properties

Property	Description / Usage	Valid Values / Examples
Task Group	The task group to display in the Task Pane for this view. When a task group is associated to a view, the following are shown: ■ The display name of the task group is shown. ■ Under the display name, the tasks that belong to this task group (which the user can run) are shown as a list.	
Application	The application for which this task group is shown. If left blank, this task group is shown for any application.	
Sequence	Order in which the task groups are shown in the Task Pane.	The value of this property is numerical.

See Also

"View" on page 325
"View Task Group Locale" on page 333

View Task Group Locale

Siebel Object Types > View > View Task Group > View Task Group Locale

Represents language-specific overrides used with the View Task Group object type.

Properties

Property	Description / Usage	Valid Values / Examples
Name **(R)**	The abbreviation of the language being used by the application.	Example: ENU.
Sequence **(O)**	Name displayed for a View Task Group for a specific language.	

See Also

"View" on page 325
"View Task Group" on page 332

View User Prop (H)

Siebel Object Types > View > View User Prop

Properties

Property	Description/Usage	Valid Values/Examples
Name	Information not available.	
Value	Information not available.	

See Also

"View" on page 325
"View Locale" on page 330
"View Report" on page 331
"View Web Template" on page 334

View Web Template

Siebel Object Types > View > View Web Template

Defines the layout of the applets in the view. The View Web Template properties are used by the Siebel Web Engine.

Properties

Property	Description/Usage	Valid Values/Examples
Name **(R)**	The name of the view Web template.	
User Layout **(O)**	A TRUE or FALSE value. Specifies whether a view can be personalized by the end user.	TRUE — Users can change the layout in the view. FALSE — Layout can not be personalized. This is the DEFAULT.
Web Template **(R)**	Web template to be used to render the view.	

See Also

"View" on page 325
"View Web Template Item" on page 335

View Web Template Item

Provides a mapping of applets to IDs of placeholders in the template (the tag is a placeholder for an applet) and selects the applet for the mapping. This can be done manually using the Object List Editor or visually using the Web View Editor.

Used by the Siebel Web Engine to map <swe:applet> tag to applets.

Properties

Property	Description/Usage	Valid Values/Examples
Applet **(R)**	Name of the applet mapped to the ID.	
Applet Mode **(O)**	Default mode of the applet when mapped to the view.	Valid values are Base, Edit, Edit List, and Query.
Applet Visibility Type **(O)**	The visibility of the applet referred to. This property is only applicable on the home page view. This setting takes precedence over the View's Visibility Applet Type property.	Valid values are All, Contact, Manager, Organization, Personal, and Sales Rep.
Display Size **(O)**	Specifies whether the applet is minimized or maximized. Always Maximized indicates that the applet cannot be minimized by the end-user. Generally applies to applets that appear on a Home Page.	Valid values are Always Maximized and Minimized.
Display Visibility **(O)**	Specifies whether the applet is visible or hidden. Always Show indicates that the applet cannot be hidden by the end-user. Generally applies to applets that appear on a Home Page.	Valid values are Always Show, Hide, and Show.
Item Identifier **(R)**	ID of the swe:applet tag in the Web Template.	
Item Identifier - Language Override **(O)**	Provides the ability to change the ID of the swe:applet tag in the Web Template when a particular language/locale is active.	

Property	Description/Usage	Valid Values/Examples
Move Range **(O)**	Defines a range in which the applet may be moved. For example, on an application home page with two columns, applets would specify a move range of either Column1 or Column2. Any applet in move range Column1 is movable only within the first (left) column. Any applet in move range Column2 is movable only within the second (right) column. If this property is not defined, the applet cannot be moved by the end-user. In other words, the applet location is fixed within the view. For example, the salutation applet on the home page would most likely not specify a move range.	
Name **(R)**	The name of the view Web template item.	
Namespace **(O)**	Qualifier for the item identifier. If the swe:applet tag for which this mapping is made is inside a swe:idgroup tag, this property is the ID of the idgroup tag.	
Namespace - Language Override **(O)**	Provides the ability to change the qualifier for the item identifier when a particular language/locale is active.	
Position **(O)**	Used to define the hierarchical relationship between the applets in the view. This property is used only when rendering the view with applets within applets as in Catalog layouts.	

See Also
"View" on page 325
"View Web Template" on page 334

View Web Template Item Locale

Siebel Object Types > View > View Web Template > View Web Template Item > View Web Template Item Locale

Represents the language-specific override used with the View Web Template Item object type.

Properties

Property	Description/Usage	Valid Values/ Examples
Item Identifier **(O)**	ID of the <swe:applet> tag in the Web Template.	
Name **(R)**	The abbreviation of the language being used by the application.	
Namespace **(O)**	Qualifier for the item identifier. If the <swe:applet tag> for which this mapping is made is inside a <swe:idgroup> tag, then this property is the ID of the idgroup tag.	

See Also

"View Web Template Item" on page 335

Web Page

Siebel Object Types > Web Page

A Web page is used as a container of views.

Properties

Property	Description/Usage	Valid Values/Examples
Do Not Use Container **(O)**	The value is TRUE or FALSE.	TRUE — The template file for this page is shown with the applications container Web page. FALSE — Shown without the container page.
Name **(R)**	The name of the Web page.	
Title	Information not available.	
Web Template **(R)**	The name of the Web template associated with this Web page.	

See Also

"Application" on page 45

Web Page Item

Item that can be shown on a Web page. Similar to a control in an applet, but for use outside the context of an applet on a Web page. These are mapped to swe:pageitem tags in the Web page.

Properties

Property	Description/Usage	Valid Values/Examples
Caption **(O)**	The display name of this item.	
HTML Attributes **(O)**	Can be used to add HTML tag attributes to the HTML tags that the Standard Interactivity client creates to render the Web page item.	
Item Identifier **(R)**	ID of the <swe:pageitem> tag.	
Item Identifier - Language Override **(O)**	Provides the ability to change the ID of the <swe:pageitem> tag when a particular language/locale is active.	
Method Invoked **(O)**	The method to be invoked.	
Name **(R)**	The name of the Web page item.	
Namespace **(O)**	Qualifier for the item identifier. If the swe:pageitem tag for which this mapping is made is inside a swe:idgroup tag, this property is the ID of the idgroup tag.	
Namespace - Language Override **(O)**	Provides the ability to change the qualifier for the item identifier when a particular language/locale is active.	
Show Popup **(O)**	A TRUE or FALSE value.	TRUE — Specifies that a new browser window is opened before invoking the method. FALSE — Specifies that the method is invoked in the current browser window.
Target View Frame **(O)**	The HTML page loaded into the frame that is specified by this property.	
Type **(R)**	Indicates how this item is displayed on the generated HTML page.	Valid values are Button, CheckBox, ComboBox, File, Label, Link, Mailto, Password, Text, TextArea, and URL.

See Also
"Control" on page 139
"Web Page" on page 337

Web Page Item Locale

Siebel Object Types > Web Page > Web Page Item > Web Page Item Locale

Represents language-specific overrides used with the Web Page Item object type.

Properties

Property	Description/Usage	Valid Values/ Examples
Caption **(O)**	The display name of this item.	
Item Identifier **(O)**	ID of the <swe:pageitem> tag.	
Name **(R)**	The abbreviation of the language being used by the application.	Example: ENU.
Namespace **(O)**	Qualifier for the item identifier. If the <swe:pageitem> tag for which this mapping is made is inside a <swe:idgroup> tag, then this property is the ID of the idgroup tag.	

See Also
"Web Page Item" on page 338

Web Page Item Parameter

Siebel Object Types > Web Page > Web Page Item > Web Page Item Parameter

Parameter of the page item. The definition varies depending on the item. For example, for a page item that has Invoked Method set to GoToView, an optional parameter is view and its value is the name of the view to go to.

Properties

Property	Description/Usage	Valid Values/ Examples
Name **(R)**	The name of the Web page item parameter.	
Value **(R)**	The value of the parameter.	

See Also
"Web Page" on page 337
"Web Page Item" on page 338

Web Page Locale (H)

Siebel Object Types > Web Page > Web Page Locale

Represents language-specific overrides used with the Web Page object type.

Properties

Property	Description/Usage	Valid Values/ Examples
Title **(O)**	The text used for the title of the Web page for a specific language.	

Web Template

Siebel Object Types > Web Template

Provides a repository representation to a (SWT) template file. Other repository objects that use templates (for example, applet, view, Web page) refer to a Web Template object by its name.

Properties

Property	Description/Usage	Valid Values/Examples
Name **(R)**	The name of the Web template.	
Type **(R)**	The Web template type.	Valid values are Applet Template, View Template, or Web Page Template.

See Also
"Applet Web Template" on page 42
"View Web Template" on page 334

Web Template File

Siebel Object Types > Web Template > Web Template File

Defines the physical file used for the Web template.

Properties

Property	Description/Usage	Valid Values/Examples
Bitmap **(O)**	Name of the bitmap file.	
File Name **(O)**	Name of the file.	
Name **(O)**	Name of the Web template file.	

See Also

"Web Template" on page 340

WF Branch Connector (H)

Siebel Object Types > Workflow Process > WF Step > WF Step Branch (H) > WF Branch Connector

Branch is a link that connects two steps. Branches can be of many types. Branch Connector is the default type of a branch. It connects two steps.

Properties

Property	Description/Usage	Valid Values/ Examples
From Step Name	The name of the step from which the branch begins.	Defaults to the value of the first step.
Layout	Identifies the coordinates of the branch on the design canvas.	Based on the position on the canvas.
Name	Name of the branch connector. Use a descriptive name that is consistent with your overall naming strategy and meaningful to the process designer.	
To Step Name	Name of the step where the branch ends.	Defaults to the name of the end step.

See Also

"WF Branch Criteria (H)" on page 341
"WF Step" on page 346
"WF Step Branch (H)" on page 349
"Workflow Process" on page 360

WF Branch Criteria (H)

Siebel Object Types > Workflow Process > WF Step > WF Step Branch (H) > WF Branch Criteria

Branch Criteria is a list of conditions. At runtime, the workflow engine evaluates the conditions and when the outcome is true, the workflow takes the path defined by the branch.

Properties

Property	Description/Usage	Valid Values/Examples
Applet Field Name (see Description)	The name of the field within the named applet. This is a required field when the Compare To value is set to Applet.	The drop-down list displays all fields defined for the selected applet.
Applet Name (see Description)	Name of the applet to compare. This is a required field when the Compare To value is set to Applet.	
Business Component (see Description)	The name of the business component within the business object of the business process. This is a required field when the Compare to value is set to Business Component or Expression.	The picklist includes business components defined for the business object selected for the process.
Business Component Field (see Description)	The name of a field within the business component. This is a required field when the Compare To value is set to Business Component.	The picklist displays all fields defined for the selected business component.
Compare To **(R)**	Indicates where the comparison value is coming from.	Applet, Business Component, Expression, and Process Property.
Edit	Indicates if this record is editable. Users cannot modify this setting.	
Name	Name of this object.	

Property	Description/Usage	Valid Values/Examples
Operation	Identifies the comparison operation.	All Must Match — All of the values must match exactly, including case. All Must Match (ignore case) — All of the values must match without regard to case. Between — Value must be between a range of values. Greater Than — Value must be greater than the comparison value. Is Not Null — Value cannot be null. Is Null — Value must be null. Less Than — Value must be less than the comparison value. None Can Match — None of the values can match exactly, including case. None Can Match (ignore case) — None of the values can match without regard to case. Not Between — Value cannot be between a range of values. One Must Match — One or more values must match exactly, including case. One Must Match (ignore case) — One or more values must match without regard to case. This Must Match — The current value must match exactly, including case. This Must Match (ignore case) — The current value must match without regard to case.

Property	Description/Usage	Valid Values/Examples
Property Name **(R)**	Identifies the specific Business Process property on which to base the condition.	
Sequence **(S)**	Sequence of the criteria (for upgrade purposes).	

See Also

"WF Branch Connector (H)" on page 341
"WF Branch Criteria Value (H)" on page 344
"WF Step" on page 346
"WF Step Branch (H)" on page 349
"Workflow Process" on page 360

WF Branch Criteria Value (H)

Siebel Object Types > Workflow Process > WF Step > WF Step Branch (H) > WF Branch Criteria (H) > WF Branch Criteria Value

This object is for Oracle use only.

WF Process Metric

Siebel Object Types > Workflow Process > WF Process Metric

Defines a metric to be collected during workflow execution.

Properties

Property	Description/Usage	Valid Values/Examples
Metric Name **(R)**	The name of the metric.	The drop-down list displays a list of pre-defined metrics.
Name **(R)**	The name of the object, pre-defaulted to the Metric Name by system.	
Parent Name **(S)**	The name of the parent Workflow Process.	
Property Name **(R)**	The name of the process property that the metric is mapped to.	The drop-down list displays a list of process properties in the task.

See Also

"Task Metric" on page 305
"WF Process Prop" on page 345
"Workflow Process" on page 360

WF Process Prop

Process Properties are fields for storing values that you can use in workflow steps, either as input and output arguments, or for performing evaluations.

Properties

Property	Description/Usage	Valid Values/Examples
Access Mode	Reserved for future use.	
Business Component **(O)**	The name of the business component containing the virtual field specified in the Virtual Field property.	This value is selected from a drop-down list of business components belonging to the workflow process business object.
Business Object **(O)**	The name of the associated business object.	Chosen from a drop-down list of business objects. Only business objects with a defined primary component appear in this drop-down list.
Correlator Flag	Determines if the process property acts as the correlator.	TRUE and FALSE
Data Type	The type of data that can be stored.	Binary — For variant or binary information. Binary types must be assigned the None value in the In/Out field. Date — For dates. Hierarchy — Data type used by Enterprise Application Integration (EAI) to store data from a property set. Integration Object — For variant or binary information. Number — For numeric data. String — For alphanumeric data.
Default Date	Initial value if the process property is a date type.	
Default Number	Initial value if the process property is a numeric type.	

Property	Description/Usage	Valid Values/Examples
Default String	Initial value if the process property is a string type.	Free-form text. If you enter <Value>, the process property is initialized with the value in the Value field of the workflow input property set.
Display Name	The name as seen on the view.	
In/Out	Describes whether or not the process property is passed in or out of the process, passed into the process and returned, or used only within the process.	In — The process property is passed into the process. (Binary types cannot be assigned this value.) Out — The process property is passed out of the process. (Binary types cannot be assigned this value.) In/Out — The process property is passed into the process and returned. (Binary types cannot be assigned this value.) None — The process property is used only within the process.
Integration Object	Data type used by Enterprise Application Integration (EAI) to store data from an integration object.	Account - Get Oracle Customer (Oracle).
Name	The name of the process property.	
Virtual Field **(O)**	The name of the business component field mapped to the workflow process property.	This value is selected from a drop-down list of fields belonging to the business component. Only calculated fields with no calculated values appear in this drop-down list.

See Also

"Workflow Process" on page 360
"WF Step" on page 346

WF Step

Siebel Object Types > Workflow Process > WF Step

WF Step represents a unit of work done by the workflow engine. Each step is a visual representation of work done by the workflow engine. A sequence of WF Steps forms a functions workflow process.

Properties

Property	Description/Usage	Valid Values/Examples
Allow Retry Flag	Information not available.	
Business Component **(O)(R)** (see Description)	Identifies the specific field within the business component on which to base the condition. This is a required field when the Compare To value is set to Business Component.	The drop-down list displays all fields defined for the selected business component.
Business Service Method	The name of the method to invoke on the service.	The drop-down list displays methods defined for the selected business service.
Business Service Name	The name of the service to invoke.	The drop-down list displays business services existing in Siebel Tools with the Hidden flag set to FALSE.
Description	Text narrative describing the purpose of the task.	
Error Code	The name of the error code entered in the log.	This value is selected from a drop-down list.
Error Message	Text of the error message.	Automatically populated when an error code is selected.
Maximum Iterations	The maximum number of times you can execute this step within a process instance. Note that when the maximum number of iterations is reached, an Object Manager error is generated and the workflow process returns an In Error status. If you want the process to run to completion, you need to use a Workflow exception mechanism (such as an error process or exception branch) to catch and handle the error.	
Name	Text narrative describing the purpose of the task.	

Property	Description/Usage	Valid Values/Examples
Operation	Identifies the comparison operation.	All Must Match — All of the values must match exactly, including case. All Must Match (ignore case) — All of the values must match without regard to case. Between — Value must be between a range of values. Greater Than — Value must be greater than the comparison value. Is Not Null — Value cannot be null. Is Null — Value must be null. Less Than — Value must be less than the comparison value. None Can Match — None of the values can match exactly, including case. None Can Match (ignore case) — None of the values can match without regard to case. Not Between — Value cannot be between a range of values. One Must Match — One or more values must match exactly, including case. One Must Match (ignore case) — One or more values must match without regard to case. This Must Match — The current value must match exactly, including case. This Must Match (ignore case) — The current value must match without regard to case.

Property	Description/Usage	Valid Values/Examples
Processing Mode **(O)**	The mode in which the process will be run when triggered by runtime events.	Local Synchronous — Executes the process in the application object manager. This is the default. Remote Asynchronous — Submits an asynchronous request to the Workflow Process Manager server component to execute the process. Remote Synchronous — Submits a synchronous request to the Workflow Process Manager server component to execute the process.
Subprocess Name	Name of the subprocess.	Chosen from a drop-down list of defined workflow processes.
Type	The type of step.	
User Interact View	Name of the Siebel view to be displayed.	Chosen from a drop-down list of defined views for that Business Object.

See Also
"WF Process Prop" on page 345
"Workflow Process" on page 360

WF Step Branch (H)

Siebel Object Types > Workflow Process > WF Step > WF Step Branch

Connects two workflow steps. It controls the data and process flow of a workflow process.

Properties

Property	Description/Usage	Valid Values/Examples
Event **(O) (R)** (see Description)	The specific event the object supports. Required if Event Object Type is specified.	The set of events is different for different object types.
Event Cancel Flag **(O)**	Abort the runtime event after executing the process.	This flag only applies to events that can be canceled. It functions like CancelOperation in scripting.

Property	Description/Usage	Valid Values/Examples
Event Object **(O) (R)** (see Description)	The name of the object. Required if Event Object Type is specified. This is the name as defined in Siebel Tools.	The set of objects is different for different object types.
Event Object Type **(O)**	The type of the object the event refers to.	Applet, Application, BusComp
Event Visibility	Controls whether waiting instances need to be resumed based on instance information from local session or from the database.	Enterprise and Local
Expression	Uses an expression to evaluate a specific value. Evaluates the expression that is entered in the Value column and uses the result. Expression as defined in Siebel Query Language. If the expression refers to any business component fields, you must enter the business component name in the Business Component column.	
Name	The name of the step branch. To import or export the workflow process, the name must be unique.	
Subevent **(O)**	An options parameter for the event.	For InvokeMethod, the name of the method being invoked. For SetFieldValue, the name of the field being set.

Property	Description/Usage	Valid Values/Examples
Type	The type of branch.	Condition — Use to indicate that a condition is defined for the branch. Default — Use to indicate that if nothing else is satisfied, this branch will be followed. Additionally, if this value is used, any conditions defined for the branch are ignored. Connector — Use whenever there is no condition branching involved. Error Exception — Use to define exception handling. This connector type captures system errors, such as an error noting that the Assignment Manager server component is not available. User Defined Exception — Use to define exception handling. This connector type captures user-defined errors, such as an error noting that an order being submitted is incomplete.
User Event Name	Unique name to describe the user event.	
User Event Storage	Receives the set of process properties.	
User Event Timeout (Days)	A timeout defined in days for the workflow that is waiting for a user event to resume operation.	

See Also

"WF Branch Connector (H)" on page 341
"WF Branch Criteria (H)" on page 341
"WF Step" on page 346
"WF Step I/O Argument (H)" on page 352
"WF Step Recipient (H)" on page 353
"Workflow Process" on page 360

WF Step I/O Argument (H)

Siebel Object Types > Workflow Process > WF Step > WF Step I/O Argument

Identifies the input and output arguments to process step. Each step accepts input from the workflow process and passes back output to the workflow process.

Properties

Property	Description/Usage	Valid Values/Examples
Business Component	The name of the business component within the business object of the business process.	
Business Component Field	The name of a field within the business component.	
Field Name	Field referred to by its name.	
Input Flag	Indicates if the field is an input argument, output argument, or a search specification. Read only.	
Method Arg	Used as storage for different calculated fields depending upon input flag.	
Name	Text narrative describing the purpose of the Step I/O Argument.	
Output Arg	The name of the output argument from the business service.	This is a drop-down list of output arguments for the selected method. An argument appears in this drop-down list if it has been defined as a business service method argument, the Hidden flag is set to FALSE, and the type is Output or Input/Output.
Property Name **(R)**	The name of the Process Property to store the results.	This is a drop-down list of properties that have been defined for the process.

Property	Description/Usage	Valid Values/Examples
Sequence	Sequence of the arguments (for upgrade purposes).	
SubProcess Input Arg	List of arguments passed to a subprocess from the parent workflow process.	The drop-down list contains process properties for the selected subprocess.
SubProcess Output Arg	The name of the output argument from the subprocess.	For Output Argument type.
Type **(R)**	The type of argument.	Business Component, Expression, Literal, and Output Argument
Value/Search Specification	An optional search specification.	If you entered Expression in the Type field, enter an expression such as [Status] LIKE '*Open*'. The expression is evaluated by the Expression business component you specify.

See Also

"WF Step" on page 346
"WF Step Branch (H)" on page 349
"WF Step I/O Argument (H)" on page 352
"WF Step Recipient (H)" on page 353
"Workflow Process" on page 360

WF Step Recipient (H)

Siebel Object Types > Workflow Process > WF Step > WF Step Recipient

Configurators use Step Recipient to assign a workflow process to an end user. Assignment occurs based on login name. This login name may be a literal, held in a process property, or the result of an expression.

Properties

Property	Description/Usage	Valid Values/ Examples
Business Component Name	The name of the business component within the business object of the business process.	
Business Object **(O)**	The name of the associated business object.	Chosen from a drop-down list of business objects. Only business objects with a defined primary component appear in this drop-down list.
Expression	Uses an expression to evaluate a specific value. Evaluates the expression that is entered in the Value column and uses the result. Use Oracle's Siebel Query Language rules when writing expressions. If the expression refers to any business component fields, you must enter the business component name in the Business Component column.	
Name **(R)**	Name of the step recipient field. Use a descriptive name that is consistent with your overall naming strategy and meaningful to the process designer	
Process Property Name **(R)**	Name of the step recipient field. Use a descriptive name that is consistent with your overall naming strategy and meaningful to the process designer	
Recipient Name	The login name of the user.	
Recipient Type Code	Type of recipient (User, Position, and so on) this is hardcoded to User.	User
Value Type Code	Determines the type of the recipient name. The recipient name can be a entered manually or populated from a business component or process property or based on expressions.	An LOV of Name, process property, business component, expression

See Also

"WF Step" on page 346
"WF Step Branch (H)" on page 349
"WF Step I/O Argument (H)" on page 352
"Workflow Process" on page 360

Workflow Policy Column (H)

Siebel Object Types > Workflow Policy Column

Identifies a column to monitor from the Siebel database or extension columns. Also specifies what is used for that column's picklist. Workflow Policy Column object definitions can be used by one or more workflow object definitions.

Properties

Property	Description / Usage	Valid Values / Examples
Applet **(O)**	The pick applet used to display the picklist.	
Column Name **(R)**	The name of the column. This is a column from the picklist of columns from the table specified in the Table Name property.	
Name **(R)**	The name of the condition column. The name should be meaningful to the policy maker and descriptive of how the column is used.	
PickList **(O)**	The picklist that appears in the Workflow Policy Detail view for this column and is used by the policy maker to select the comparison value of the column.	
Source Field **(O)**	The field in the business component of the picklist that is the source of the comparison value. This is a field name from the picklist specified in the PickList property.	
Table Name **(R)**	The name of the table that contains the column. Chosen from a picklist.	

Adding and Changing Workflow Policy Columns

You can add new Workflow Policy Column object definitions, but it is not recommended that you change any of the properties in a predefined workflow policy column. If you need to modify a predefined column, you should add a new column instead.

See Also

"Workflow Policy Component" on page 355
"Workflow Policy Component Col" on page 357
"Workflow Policy Object (H)" on page 358

Workflow Policy Component

Siebel Object Types > Workflow Policy Object (H) > Workflow Policy Component

Identifies a database table and its relationship with another table in the workflow object. A primary workflow policy component is a workflow policy component that all other workflow policy components are directly or indirectly related to. From these workflow policy components, the workflow conditions (workflow policy columns) that are available for monitoring in the workflow policy can be defined.

Properties

Property	Description/Usage	Valid Values/Examples
Join Spec **(O)**	Specifies an additional join between this workflow policy component and the target workflow policy component, when the existing join is to an intersection table and would therefore return multiple rows. The second join restricts the relationship so that only one row is returned at a time. The expression entered in the property is equivalent to an additional WHERE clause in SQL, and it takes the form: `workflow_component.foreign_key = business_component.primary_key`	Example: The following join specification occurs in the Primary Opportunity/Position workflow policy component in the Opportunity workflow object: `[Primary Opportunity/ Position].OPTY_ID = [Opportunity].ROW_ID` The component named must be in brackets.
Name **(R)**	The name of the workflow policy component.	
Primary **(O)**	A TRUE or FALSE value indicating whether this workflow policy component is the primary one for the parent workflow policy object definition. Each workflow policy object definition must have one (and only one) primary workflow policy component.	TRUE — The workflow policy component is the primary one.
Source Column Name **(O)**	The column in the source table that relates to another workflow policy component. This is from a drop-down list of columns from the table specified in the Source Table Name property.	
Source Table Name **(R)**	The table that the workflow policy component is based on. This is a table name from the drop-down list.	

Property	Description/Usage	Valid Values/Examples
Target Column Name **(O)**	The column in the target workflow policy component that the source column in this workflow policy component is joined to. This is selected from a drop-down list of columns from the table specified in the target workflow policy component.	
Target Component Name **(O)**	The target workflow policy component that this workflow policy component is related to. This is a component name from the list of workflow policy components for the current workflow policy object.	

See Also

"Workflow Policy Column (H)" on page 355
"Workflow Policy Component Col" on page 357
"Workflow Policy Object (H)" on page 358

Workflow Policy Component Col

Siebel Object Types > Workflow Policy Object (H) > Workflow Policy Component > Workflow Policy Component Col

Identifies a column that can be monitored from the workflow policy component that is its parent.

Properties

Property	Description/Usage	Valid Values/ Examples
Name **(R)**	The name of the column as it appears in the Conditions applet in the Workflow Policy Detail view. This property defaults to the same value as the Workflow Column Name property.	
Workflow Column Name **(R)**	The name of the column monitored in the selected workflow policy component. This name must be an existing workflow policy column object definition.	
Workflow Object Name **(R)**	The name of the workflow policy object associated with this workflow policy component column.	

See Also
“Workflow Policy Column (H)” on page 355
“Workflow Policy Component” on page 355
“Workflow Policy Object (H)” on page 358

Workflow Policy Object (H)

Siebel Object Types > Workflow Policy Object

Groups workflow policy component objects and workflow policy component columns that can be monitored in the same workflow policy. Each workflow policy object definition contains exactly one primary workflow policy component object, any number of additional workflow policy components, and any number of workflow policy component columns from those workflow policy components.

Properties

Property	Description/Usage	Valid Values/ Examples
Name **(R)**	The name of the workflow policy object. This name should be meaningful to the policy maker, and it is generally the same as the corresponding business object.	

See Also
“Assignment Object (H)” on page 60
“Workflow Policy Column (H)” on page 355
“Workflow Policy Component” on page 355
“Workflow Policy Component Col” on page 357

Workflow Policy Program (H)

Siebel Object Types > Workflow Policy Program

Provides the name and type of a workflow policy program or action that can be initiated in response to the conditions of policies being met. The names of the defined Workflow Policy Program object definitions appear in the drop-down list of the Actions Applet field in the Workflow Policy Detail view. There are four types of responses available: Send Email, Send Page, Run External Program, and Database Operation. There are also predefined workflow policy programs that have all the variables set except for the message text.

Properties

Property	Description/Usage	Valid Values/Examples
Name **(R)**	The descriptive name of the action to perform.	
Type **(R)**	The type of action.	Values are selected from a drop-down list, and are: ■ DB Operation — Insert or update a database table based on arguments. ■ External Program — Execute an external program in Windows. ■ Generic Request Server — Invoke a Server Component. ■ Send Broadcast Message — Broadcast a message to users. ■ Send Message — Compose and send an automatic email message. ■ Send Page — Send a page to a pager.
Workflow Object **(O)**	The name of the workflow policy object with which the workflow policy program is associated.	

Variables in Messages

Variables can be used in the messages to make the content more specific. For example, a message can state “The [SR] from [Account] has a severity level [Value] and has been open for 2 hours.” Using an SQL statement, you can fill in each of the variables with specific data to make the message more meaningful. This method also allows you to use the same message for more than one policy.

See Also

“Workflow Policy Program Arg” on page 359
“Workflow Policy Object (H)” on page 358

Workflow Policy Program Arg

Siebel Object Types > Workflow Policy Program (H) > Workflow Policy Program Arg

Defines a message, sender, recipient, database action, program to execute, or similar kind of parameter value for the parent Workflow Policy Program object definition. One Workflow Policy Program object definition typically has several Workflow Policy Program Arg object definitions.

CAUTION: The Workflow Policy Program Arg object type should be created and modified with caution. Refer to *Siebel Business Process Framework: Workflow Guide* for details.

Properties

Property	Description/Usage	Valid Values/Examples
Applet **(O)**	The picklist applet.	
Default Value **(O)**	The text value of a type that depends on the name of the workflow policy program argument; for example, an SQL statement, the text of a message, or the email address of a recipient.	
Name **(R)**	The argument, identified from a predefined list.	
Picklist **(O)**	The picklist object definition.	
Required **(O)**	A TRUE or FALSE value that indicates whether or not data entry is required.	TRUE — Data entry is required.
Source Field **(O)**	The picklist source field.	
Visible **(O)**	A TRUE or FALSE value that indicates whether the data supplied by this argument is displayed.	TRUE — The data is displayed.

See Also

"Workflow Policy Object (H)" on page 358

Workflow Process

Siebel Object Types > Workflow Process

Workflow process is a executable version of an organization's business process. Use it to define your company's business processes using a familiar flowcharting interface. A workflow process consists of one or more process steps such as start steps, subprocesses, decision points, and tasks.

Properties

Property	Description/Usage	Valid Values/Examples
Auto Persist	Flag that indicates if the workflow instance information should be saved to the database.	TRUE or FALSE
Business Object	The name of the associated business object.	Optional. This value is chosen from a drop-down list of business objects. Only business objects with a defined primary component appear in this drop-down list.

Property	Description/Usage	Valid Values/Examples
Description	A text narrative describing the purpose of the process.	
Error Process Name	The name of the error process to run when this process encounters an error.	Optional. This value is selected from a drop-down list of workflow processes.
Group	The name of the associated group for the business process.	Optional. Chosen from a drop-down list. Use this value to segregate business processes by group. You cannot type a value for the group. The value must be chosen from the drop-down list or left null. Groups for business processes are defined in the List of Values view in Application Administration.
Module	Information not available.	
Process Name	Descriptive name that describes the process text.	
Status	The current status of the process.	Read-only. The default status is Inactive.
Version	The version number of this process definition.	Read-only. The default version is 0.
Workflow Mode	A set of modes that defines the runtime behavior of a workflow.	

See Also

"WF Process Prop" on page 345
"WF Step" on page 346

Index

B

C

D

E

F

G

H

I

M

N

O

P

Q

R

S

T

U

V

W

X