Siebel Master Data Applications Reference for Industry Applications

Version 8.0, Rev. B
May 2007

ORACLE®

Copyright © 2006, 2007, Oracle. All rights reserved.

The Programs (which include both the software and documentation) contain proprietary information; they are provided under a license agreement containing restrictions on use and disclosure and are also protected by copyright, patent, and other intellectual and industrial property laws. Reverse engineering, disassembly, or decompilation of the Programs, except to the extent required to obtain interoperability with other independently created software or as specified by law, is prohibited.

The information contained in this document is subject to change without notice. If you find any problems in the documentation, please report them to us in writing. This document is not warranted to be error-free. Except as may be expressly permitted in your license agreement for these Programs, no part of these Programs may be reproduced or transmitted in any form or by any means, electronic or mechanical, for any purpose.

PRODUCT MODULES AND OPTIONS. This guide contains descriptions of modules that are optional and for which you may not have purchased a license. Siebel's Sample Database also includes data related to these optional modules. As a result, your software implementation may differ from descriptions in this guide. To find out more about the modules your organization has purchased, see your corporate purchasing agent or your Siebel sales representative.

If the Programs are delivered to the United States Government or anyone licensing or using the Programs on behalf of the United States Government, the following notice is applicable:

U.S. GOVERNMENT RIGHTS. Programs, software, databases, and related documentation and technical data delivered to U.S. Government customers are "commercial computer software" or "commercial technical data" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, use, duplication, disclosure, modification, and adaptation of the Programs, including documentation and technical data, shall be subject to the licensing restrictions set forth in the applicable Oracle license agreement, and, to the extent applicable, the additional rights set forth in FAR 52.227-19, Commercial Computer Software--Restricted Rights (June 1987). Oracle USA, Inc., 500 Oracle Parkway, Redwood City, CA 94065.

The Programs are not intended for use in any nuclear, aviation, mass transit, medical, or other inherently dangerous applications. It shall be the licensee's responsibility to take all appropriate fail-safe, backup, redundancy and other measures to ensure the safe use of such applications if the Programs are used for such purposes, and we disclaim liability for any damages caused by such use of the Programs.

Oracle, JD Edwards, PeopleSoft, and Siebel are registered trademarks of Oracle Corporation and/or its affiliates. Other names may be trademarks of their respective owners.

The Programs may provide links to Web sites and access to content, products, and services from third parties. Oracle is not responsible for the availability of, or any content provided on, third-party Web sites. You bear all risks associated with the use of such content. If you choose to purchase any products or services from a third party, the relationship is directly between you and the third party. Oracle is not responsible for: (a) the quality of third-party products or services; or (b) fulfilling any of the terms of the agreement with the third party, including delivery of products or services and warranty obligations related to purchased products or services. Oracle is not responsible for any loss or damage of any sort that you may incur from dealing with any third party.

Contents

Chapter 1: What's New in This Release

Chapter 2: About Siebel Master Data Applications

About Siebel Master Data Applications Modules 11

Product Components Included with Siebel Master Data Applications 14

About Siebel Universal Customer Master Concepts 15

Chapter 3: Siebel Master Data Applications Scenario

Siebel Universal Customer Master Scenario 19

Chapter 4: Architecture and Framework

About Siebel Master Data Applications Framework and Interactions 22

About Siebel Master Data Applications Functions and Services 22

About Integration Services for Siebel Master Data Applications 24

About Siebel Master Data Applications Architecture 25

- Siebel Universal Customer Master User Interface Layer 26
- Siebel Universal Customer Master Business Objects Layer 26
- Siebel Universal Customer Master Data Objects Layer 30
- Siebel Connector and Integration of Siebel Master Data Applications 30

About Siebel Master Data Application Workflows 35

- UCM Account Batch Data Management Workflow 35
- UCM Async Realtime Publish Workflow 35
- UCM Batch Process 36
- UCM Batch Process - Single step 36
- UCM Contact Batch Data Management Workflow 36
- UCM Customer Profile Integration Process 36
- UCM Customer Profile Integration Process (File Transport Test Workflow) 36
- UCM Daily Publish 37

About Siebel Universal Customer Master Server Components 37

About the Universally Unique Identifier (UUID) 37

Chapter 5: Configuring Siebel Master Data Applications

Loading Data into Siebel Universal Customer Master and Updating External Systems 39

Activating Siebel Universal Customer Master External Systems 39

Configuring Siebel Master Data Application Features 40

About Siebel Master Data Applications Integration Services 40
Configuring UCM Transaction Manager for Data Management and Survivorship 49
Configuring UCM Auto Merge and Manual Review Threshold Values 49
Configuring UCM Survivorship Engine Properties 50
About UCM Data Management Services 50
Configuring UCM Data Quality Manager for Exact Match 51
Configuring UCM Unmerge Functionality 52
Configuring UCM Server Components for Data Management and Survivorship 53
Configuring Run-time Events for UI Interaction 54
Configuring Default System for UI Interaction 55
Configuring UCM Inbound Server Communication 55
About Siebel Universal Customer Master and CRM Integration 61
Extending Siebel Universal Customer Master and CRM Integration 61

Configuring Siebel Master Data Integration Objects 61

About Siebel Master Data Application Integration Objects 62
Modifying Siebel Master Data Application Integration Objects 63
Modifying UCM Envelope Integration Objects 64
Extending UCM Integration Objects 64
Modifying the Map Property of Application Integration Objects 65
Adding Status Key Property to UCM Integration Object 66

Chapter 6: Administering Siebel Master Data Applications

Registering Systems Connected to Siebel Master Data Applications 67

Setting System Privileges for Siebel Master Data Applications 69

About System Publish and Subscribe Modes 72

Configuring System Publish and Subscribe for Siebel Master Data Applications 73

About UCM Survivorship Rules 76

Process of Creating Survivorship Rules 77

Creating Attribute Groups for Survivorship Rules 78

Creating UCM Survivorship Rules 78

Activating UCM Survivorship Rules 80

Example of a Survivorship Rule 81

About Managing UCM Data 83
About Cross-Referencing Records with External Systems 84
Cross-Referencing Records with External Systems 84
About SDH Tables 85
About SDH Type Codes 86
Monitoring History of UCM Records 86
Monitoring Source Data of UCM Records 87
Purging Source UCM Data 87
Managing Pending Updates from External Systems 88
Managing Duplicate UCM Records 88
Unmerging Previously Merged UCM Records 89
Running Data Management in Batch Mode 90
About System Preferences for Siebel Universal Customer Master 91
Configuring Siebel Data Quality Cleansing for Siebel UCM 92
Configuring SSA Data Matching for Siebel UCM 94
About Siebel Universal Customer Master Data Hierarchy 96
Troubleshooting Siebel Master Data Applications 96

Chapter 7: Siebel Master Data Applications Messages

Siebel Master Data Applications Application Services Interfaces 109
About CRMML Messages 114
Siebel Master Data Application Messages 122
Contact Profile 122
Contact Address 126
Contact Activity 130
Contact Product 134
Contact Relationship 138
Contact 143
Household 154
Personal Address 159
FINCORP Account 161
Account Profile 172
Account Address 176
Account Activity 181
Account Product 188
Account Relationship 194

Account 201
Cross Reference 215

Chapter 8: Configuring and Administering Siebel Universal Customer Master Privacy Management

About Siebel Universal Customer Master Privacy Management 221
About Siebel Universal Customer Master Privacy Management Features and Capabilities 221
About Siebel Universal Customer Master Privacy Management Configuration 222
About Siebel Universal Customer Master Privacy Management Administration 222

Process of Configuring Siebel Universal Customer Master Privacy Management 222
Preliminary Requirements Before Configuring Siebel Universal Customer Master Privacy Management 223
Deploying Siebel Universal Customer Master Privacy Workflow Processes 225
Setting Privacy-Related User Properties 225

Administering Siebel Universal Customer Master Privacy Management 228
Managing Privacy Data 228
Manage Privacy Rule Modules 230
Managing Privacy Runtime Events 230

Chapter 9: Privacy Management Solution

About Privacy Management 237

Privacy Vocabulary 238

Sample Privacy Rules 241

Customization Methods 244
Customization Strategies 244
Required Customizations 245
Customization Scenarios 246

About Blank Privacy Source Data and History Records 248

Appendix A: Additional License Restrictions

Additional License Grants and Restrictions Regarding Siebel Master Data Applications 251

Index

1 What's New in This Release

What's New in Siebel Master Data Applications Reference for Industry Applications, Version 8.0, Rev. B

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

Table 1. What's new in Siebel Master Data Applications Reference for Industry Applications, Version 8.0, Rev. B

Topic	Description
"Privacy Management Policy Hub" on page 10	Added explanation of Privacy Management Policy Hub.
"Additional License Grants and Restrictions Regarding Siebel Master Data Applications" on page 251	Added Loyalty business components to the Siebel Universal Marketing Master module.

Table 2 lists changes described in this version of the documentation to support Release 8.0 of the software.

Table 2. What's new in Siebel Master Data Applications Reference for Industry Applications, Version 8.0, Rev. A

Topic	Description
"About Siebel Master Data Applications" on page 9	Added bundle and licensing information on Customer Hub Data Steward and Product Hub Data Steward.

NOTE: Before upgrading from V7.8.2 to V8.0, read "Universal Customer Master (UCM) Upgrade Steps After Repository Merge" in *Siebel Database Upgrade Guide*.

What's New in Siebel Master Data Applications Reference for Industry Applications, Version 8.0

Table 3 lists changes described in this version of the documentation to support Release 8.0 of the software.

Table 3. What's new in Siebel Master Data Applications Reference for Industry Applications, Version 8.0

Topic	Description
Configuration Read "Configuring Siebel Master Data Applications" on page 39.	Added configuration information.
Privacy Management Read "Privacy Management Solution" on page 237.	Added new chapter on privacy management.
Privacy Rules Read "Configuring and Administering Siebel Universal Customer Master Privacy Management" on page 221.	Added new chapter on privacy rules.
System Publish and Subscribe Modes Read "About System Publish and Subscribe Modes" on page 72	Added information on System Publish and Subscribe Modes
Troubleshooting Read "Troubleshooting Siebel Master Data Applications" on page 96.	Added information on troubleshooting.

NOTE: Before upgrading from V7.8.2 to V8.0, read "Universal Customer Master (UCM) Upgrade Steps After Repository Merge" in *Siebel Database Upgrade Guide*.

2 About Siebel Master Data Applications

Oracle's Siebel Master Data Applications provide a Siebel platform that functions as the master file for an organization's enterprise-wide customer information. They consist of the following master file modules:

- Siebel Universal Customer Master
- Siebel Universal Activity Master
- Siebel Universal Product Master
- Siebel Universal Marketing Master
- Siebel Universal Sales Master
- Siebel Universal Service Master
- Siebel Universal Field Service Master
- Siebel Universal Identification Master

The Siebel Universal Customer Master is the base module on which all other master file modules build. These modules add extended enterprise-wide information capabilities. The Siebel Universal Customer Master module consists of a set of Siebel data model tables that store customer data; the other modules extend this concept to hold additional marketing, sales, and service data. These modules are all part of Oracle's Master Data Management (MDM) family of applications.

For information about Siebel Master Data Applications, see "About Siebel Master Data Applications Modules" on page 11.

When Siebel Master Data Applications are used as the master database of an organization's data, one or more of the modules—through Oracle and third-party software—interact with back-office systems and Oracle's Siebel Business Application deployments to provide a unified data-set across an organization's multiple channels, lines of business, and applications.

Customer Hub B2B

The Customer Hub B2B is a bundle that includes two components, Siebel Universal Customer Master B2B and Oracle Customer Data Hub.

If you are licensed for Customer Hub B2B, your use of the bundled programs is as defined in your license order.

If you are licensed for Siebel Universal Customer Master B2B, your use is limited to the component that you are licensed for and you do not have rights to use the Oracle Customer Data Hub.

If you are licensed for Oracle Customer Data Hub, your use is limited to the component that you are licensed for and you do not have rights to use the Siebel Universal Customer Master B2B.

Customer Hub B2C

The Customer Hub B2C is a bundle that includes two components, Siebel Universal Customer Master B2C and Oracle Customer Data Hub.

If you are licensed for Customer Hub B2C, your use of the bundled programs is as defined in your license order.

If you are licensed for Siebel Universal Customer Master B2C, your use is limited to the component that you are licensed for and you do not have rights to use the Oracle Customer Data Hub.

If you are licensed for Oracle Customer Data Hub, your use is limited to the component that you are licensed for and you do not have rights to use the Siebel Universal Customer Master B2C.

Product Hub

The Product Hub is a bundle that includes two components, Siebel Universal Product Master and Oracle Product Information Management Data Hub.

If you are licensed for Product Hub then your use of the bundled programs is as defined in your license order.

If you are licensed for Siebel Universal Product Master then your use is limited to the component that you are licensed for and you do not have rights to use the Oracle Product Information Management Data Hub.

If you are licensed for Oracle Product Information Management Data Hub then your use is limited to the component that you are licensed for and you do not have rights to use the Siebel Universal Product Master.

Customer Hub Data Steward

The Customer Hub Data Steward is a bundle that includes two components, Siebel Customer Data Steward and Oracle Customer Data Librarian.

If you are licensed for Customer Hub Data Steward, each user has a right to use the Siebel Customer Data Steward and the Oracle Customer Data Librarian.

If you are licensed for Siebel Customer Data Steward, your use is limited to the component that you are licensed for and you do not have rights to use the Oracle Customer Data Librarian.

If you are licensed for Oracle Customer Data Librarian, your use is limited to the component that you are licensed for and you do not have rights to use the Siebel Customer Data Steward.

Privacy Management Policy Hub

The Privacy Management Policy Hub module is an application extension to the Siebel Universal Customer Master that creates a central storage location for privacy rules. For more information, see "Privacy Management Solution" on page 237.

Product Hub Data Steward

The Product Hub Data Steward is a bundle that includes two components, Siebel Product Data Steward and Oracle Product Information Management Data Librarian[1].

If you are licensed for Product Hub Data Steward, each user has a right to use the Siebel Product Data Steward and the Oracle Product Information Management Data Librarian[1].

If you are licensed for Siebel Product Data Steward, your use is limited to the component that you are licensed for and you do not have rights to use the Oracle Product Information Management Data Librarian[1].

If you are licensed for Oracle Product Information Management Data Librarian,[1] your use is limited to the component that you are licensed for and you do not have rights to use the Siebel Product Data Steward.

In addition to the data model mentioned previously, Siebel Master Data Applications include a data manager, an object manager, and a user interface for administrators included with the Data Steward licensing options.

For information on integrating Siebel Master Data Applications, see Chapter 5, "Configuring Siebel Master Data Applications."

The chapter includes the following topics:

- "About Siebel Master Data Applications Modules" on page 11
- "Product Components Included with Siebel Master Data Applications" on page 14

About Siebel Master Data Applications Modules

Siebel Master Data Applications include the base module (Siebel Universal Customer Master) and the following additional modules comprising various subsets of enterprise-wide customer related data:

- "Siebel Universal Customer Master"
- "Siebel Universal Activity Master"
- "Siebel Universal Product Master"
- "Siebel Universal Marketing Master" on page 12
- "Siebel Universal Sales Master" on page 12
- "Siebel Universal Service Master" on page 13
- "Siebel Universal Field Service Master" on page 13
- "Siebel Universal Identification Master" on page 13

1. (Includes: Engineering, Bills of Material, Restricted Use: Inventory Management)

Siebel Universal Customer Master

The Siebel Universal Customer Master is a Siebel platform configured to store a clean and unified profile for enterprise customers, partners, and prospects. Traditional customer data such as Accounts, Contacts, Households, Partner, and Agent data is included as well as customer relationship information, address information, and asset information. The Siebel Universal Customer Master interacts within an enterprise architecture by integrating with key back-office systems to act as the master record for the customer-specific subset of an organization's data.

The Siebel Universal Customer Master is based on the Siebel party data model. This model uses a single-party entity to represent organizations, positions, user lists, contacts, and employees and is implemented with Siebel extension tables. Each record of the party data model table can have complex hierarchical relationships with other parties.

For further information on concepts and functionality of the Siebel Universal Customer Master, see "About Siebel Universal Customer Master Concepts" on page 15.

Siebel Universal Activity Master

The Siebel Universal Activity Master module is an application extension to the Siebel Universal Customer Master that stores additional customer information. This extension creates an extended profiling view of an organization's customers. The key entities included with this module are activities, notes, agreements, entitlements, billing accounts, demographic information, business profiles, customer preferences, credit profiles, activities history, and analytical and segmentation information.

Siebel Universal Product Master

The Siebel Universal Product Master module is an application extension to the Siebel Universal Customer Master that stores an organization's product information. This extension creates a common and consistent definition of an organization's products and catalog. The key entities included with this module are product line management, product configuration, complex products, and price list.

Siebel Universal Marketing Master

The Siebel Universal Marketing Master module is an application extension to the Siebel Universal Customer Master that stores an organization's marketing information. This extension creates a central storage location for marketing efforts across an organization's multiple channels and disparate systems. The key entities included with this module are campaigns, offers, and responses.

Siebel Universal Sales Master

The Siebel Universal Sales Master module is an application extension to the Siebel Universal Customer Master that stores an organization's sales information. This extension creates a central storage location for sales efforts across an organization's multiple channels and disparate systems. The key entities included with this module are opportunities, opportunity products, competitors, decision issues, quotes, and orders.

Siebel Universal Service Master

The Siebel Universal Service Master module is an application extension to the Siebel Universal Customer Master that stores an organization's service information. This extension creates a central storage location for service efforts across an organization's multiple channels and disparate systems. The key entities included with this module are service requests, solutions, and account and policy details.

Siebel Universal Field Service Master

The Siebel Universal Field Service Master module is an application extension to the Siebel Universal Customer Master that stores an organization's field service information. This extension creates a central storage location for field service efforts across an organization's multiple channels and disparate systems. The key entities included with this module are field service activities, service agreements, asset management, invoices, inventory, repair, scheduling, and preventive maintenance.

Siebel Universal Identification Master

The Siebel Universal Identification Master module is an application extension to the Siebel Universal Customer Master that provides a secure repository for customers' social security numbers and generates a new Unique Universal ID cross-referenced with the customer's social security number. Siebel Universal Identification Master allows companies to comply with new privacy legislation that limits the use of the social security number as a unique customer identifier. The legislation requires companies to safeguard the social security numbers of their customers and severely limits the use of social security numbers in customer interactions.

Product Hub

See "About Siebel Master Data Applications" on page 9.

Customer Hub B2B

See "About Siebel Master Data Applications" on page 9.

Customer Hub B2C

See "About Siebel Master Data Applications" on page 9.

Customer Hub Data Steward

See "About Siebel Master Data Applications" on page 9.

Product Hub Data Steward

See "About Siebel Master Data Applications" on page 9.

Product Components Included with Siebel Master Data Applications

Certain product components of Siebel Business Applications are included with Siebel Master Data Applications and provide, or add, functionality to the master files. Only the Siebel products required to run Siebel Master Data Applications are included with the Siebel Master Data Applications base product. The following Business Application products are included with Siebel Master Data Applications:

- Siebel Application Object Manager

  Siebel Application Object Manager hosts the Business Objects layer and the Data Objects layer of the Siebel Business Application Architecture. See *Siebel System Administration Guide* for further information.

- Siebel Enterprise Application Integration (Siebel EAI)

  Siebel EAI provides components for integrating Siebel Business Applications with external applications and technologies. See *Overview: Siebel Enterprise Application Integration* for further information.

- Siebel Enterprise Integration Manager (Siebel EIM)

  Siebel EIM manages the exchange of data between Siebel database tables and other back-office databases. Siebel EIM is used for bulk data imports, exports, merges, and deletes. See *Siebel Enterprise Integration Manager Administration Guide* for further information.

- Siebel Business Process Designer

  Siebel Business Process Designer is a customizable business application that allows you to design, manage, and enforce your business processes. It allows you to design complex workflow processes and automate the enforcement of business policies and procedures. See *Siebel Business Process Framework: Workflow Guide* for further details about this product.

- Siebel Entity Relationship Designer

  The Entity Relationship Designer is a visual design tool that allows you to create entity relationship diagrams (ERDs) to represent your business and then map the entities and relationships depicted in the diagram to objects in the Siebel repository, such as business components, links, and joins. See *Configuring Siebel Business Applications* for further information.

Based on the software purchased, one or more of the following Siebel products and functionality may also be relevant to Siebel Master Data Applications implementation:

- Siebel Replication Manager

  Siebel Replication Manager distributes full schema and data replicas in order to support subordinate Siebel deployments. See *Siebel Remote and Replication Manager Administration Guide* for further information.

- Siebel Business Rules Developer (HaleyAuthority)

  Siebel Business Rules Developer provides a platform for users to develop new or customize existing business rules for company-specific privacy management requirements. It is seamlessly integrated into the UCM Privacy Management Solution.

- Siebel Assignment Manager

  Siebel Assignment Manager distributes and assigns entities such as opportunities or service requests to individuals, positions, or organizations based on defined assignment rules. See *Siebel Assignment Manager Administration Guide* for further information.

- Siebel Tools

  Siebel Tools is an integrated development environment for configuring/customizing various aspects of Siebel Business Applications, including Data Layer, Object Layer, User Interface Layer, and publish/subscribe services. See *Configuring Siebel Business Applications* for further information.

- Siebel Data Quality

  Siebel Data Quality assists enterprises by standardizing their contact, account, and prospect data by data matching and data cleansing. See *Siebel Data Quality Administration Guide* for further information.

- Siebel Audit Trail Functionality

  Siebel Audit Trail functionality creates a history of the changes that have been made to various types of information in Siebel Business Applications. See *Siebel Applications Administration Guide* for further information.

- Siebel D&B Integration

  D&B's information and technology solutions help businesses find profitable customers, reduce credit risk, manage receivables, and manage vendors. D&B's database of commercial information consists of over 75 million records world wide. Siebel D&B integration allows Siebel Business Applications to access and use D&B data and reports. See *Siebel Applications Administration Guide*.

About Siebel Universal Customer Master Concepts

Siebel Universal Customer Master (Siebel UCM) is a Siebel platform configured to store a clean and unified profile for enterprise customer data. Siebel UCM is the primary Master Data Applications product and serves as the example for installation, configuration, and administration tasks that are further documented in the following chapters.

In addition to storing the master set of customer data for an enterprise, Siebel UCM also includes many features to cleanse, evaluate, publish, store, and manage this customer data. The basic Siebel UCM concepts and functionality are briefly described in Table 4.

Table 4. Siebel UCM Features

Feature	Description
UCM Survivorship	The UCM Survivorship feature provides a rules-based means to automate the quality of the master customer data. Data is compared to its source and age to determine whether to maintain or update customer data. For further information on UCM Survivorship, see "About UCM Survivorship Rules" on page 76.
UCM Cross-referencing	UCM cross-referencing allows the identification of customer data in external systems to be saved in the Siebel UCM allowing a one-to-many mapping of this data. For further information on UCM cross-referencing, see "About Cross-Referencing Records with External Systems" on page 84.
UCM Best Version Records	UCM Best Version records describe the current best state of the customer data stored in the Siebel UCM. These records may be updated in the future based on survivorship rules or other data management processes. Historical best versions of customer data are also stored in the UCM Source Data History table.
UCM Source Data History (SDH) Table	The UCM Source Data History (SDH) tables maintain a record of data transactions between Siebel UCM and registered external systems. For further information on SDH tables, see "About SDH Tables" on page 85.
UCM Publish and Subscribe	UCM publish and subscribe functionality determines the details on how external systems receive customer data updates from the Siebel UCM. For further information on UCM publish and subscribe, see"About System Publish and Subscribe Modes" on page 72 and "Configuring System Publish and Subscribe for Siebel Master Data Applications" on page 73.
UCM Data Cleansing and Matching	Siebel UCM supports data cleansing using the Siebel Data Quality module and data matching using SSA-NAME3 (SSA) third-party software. Siebel UCM also supports other third-party cleansing and matching technologies. For further information on configuring UCM data cleansing and matching, see "Configuring Siebel Data Quality Cleansing for Siebel UCM" on page 92 and "Configuring SSA Data Matching for Siebel UCM" on page 94.

Table 4. Siebel UCM Features

Feature	Description
Siebel Connector for Master Data Applications	Siebel Connector for Master Data applications describes a set of preconfigured business services used to administer Siebel UCM processes. For further information on the Siebel Connector for Master Data Applications, see "Siebel Connector and Integration of Siebel Master Data Applications" on page 30.
UCM CRMML Messages	Siebel UCM CRMML messages are a Siebel-specific XML messaging format used for UCM data exchange. For further information on CRMML messages, see "About CRMML Messages" on page 114.

3 Siebel Master Data Applications Scenario

This chapter examines a fictional financial institution, its business requirements, and the Siebel Master Data Applications product solution based on the following scenario: a stand-alone UCM deployment. For background information about Siebel Master Data Applications, see Chapter 2, "About Siebel Master Data Applications."

Topics in this chapter include the following:

- "Siebel Universal Customer Master Scenario" on page 19

Siebel Universal Customer Master Scenario

The following business scenario is designed to provide guidance for a potential Siebel Master Data Applications deployment. The business institution and its existing implementation are described, followed by the benefits of the Siebel Master Data Applications and Siebel Universal Application Network solution. A review of the integration process with back-office systems is also included.

Business Client

A large, successful financial institution has evolved into a national enterprise with offices and business units in several locations across the country. Its core banking systems include client deposits, loans, and mortgages. It also offers credit card services and an investment banking service. Through mergers and acquisitions, the financial institution is also involved with the insurance industry, providing life, home, and car policies for its clients.

Existing Implementation

With this large number of business ventures and offices, the financial institution found itself using a myriad of applications for individual business units. Even in the same business units, different departments (HR and Accounting, for example) functioned on different back-office systems, which included multiple front office applications and multiple middle-tier systems, and various applications that stored subsets of customer information. Client, partner, and product information is spread across multiple IT systems. A large percentage of the IT budget for this institution is spent on maintenance and integration of these applications. However, removing or expanding the functionality of these legacy systems is difficult and costly.

The solution is to continue to use the Siebel Master Data Applications as a key component in the synchronization of customer profile information across customer-related applications, including back-office applications.

Siebel Master Data Applications Deployment

In this scenario, Siebel Master Data Applications operate as a stand-alone application in an application network. Siebel Master Data Applications reside outside any operational application and serve to store, synchronize, and reconcile customer data across the financial organization's enterprise. See Figure 1 for a model of this deployment.

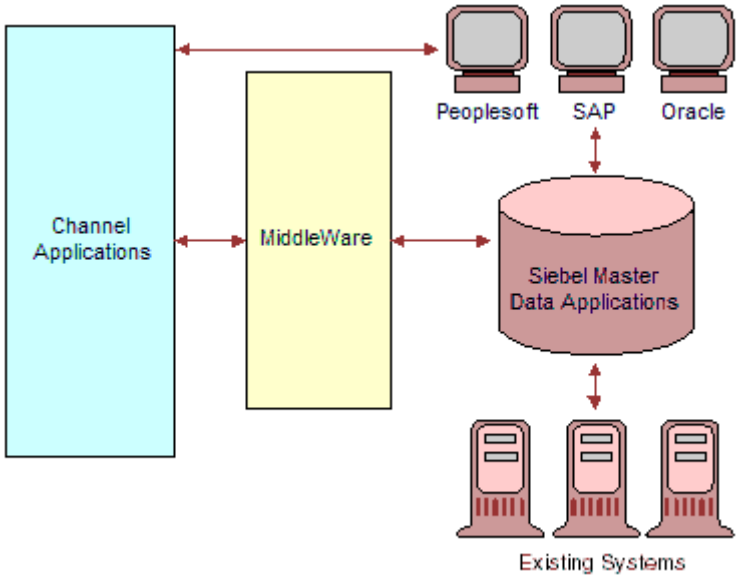

Figure 1. Siebel Master Data Applications Stand-Alone Deployment Scenario

Siebel Master Data Applications Integration

The Siebel Master Data Applications deployment uses several prebuilt connectors along with the full Siebel EAI module functionality to integrate and consolidate customer information across back-office systems. See "Siebel Connector and Integration of Siebel Master Data Applications" on page 30 for details about this process.

4 Architecture and Framework

This chapter provides a background for the framework, architecture, integration, and interactions of Siebel Master Data Applications. It also briefly describes the application's functionality and administration.

The chapter includes the following topics:

- "About Siebel Master Data Applications Framework and Interactions" on page 22
- "About Siebel Master Data Applications Functions and Services" on page 22
- "About Integration Services for Siebel Master Data Applications" on page 24
- "About Siebel Master Data Applications Architecture" on page 25
- "About Siebel Master Data Application Workflows" on page 35
- "About Siebel Universal Customer Master Server Components" on page 37
- "About the Universally Unique Identifier (UUID)" on page 37

About Siebel Master Data Applications Framework and Interactions

Siebel Master Data Applications form the master application and database of an organization's data. It interacts with back-office systems and Siebel deployments to provide different organizational business units with consistent and timely data. A variety of functionality is available between Siebel Master Data Applications and Siebel Business Applications or back-office applications using enterprise integration and the regular links to Siebel Business Applications. See Figure 2 for this relationship.

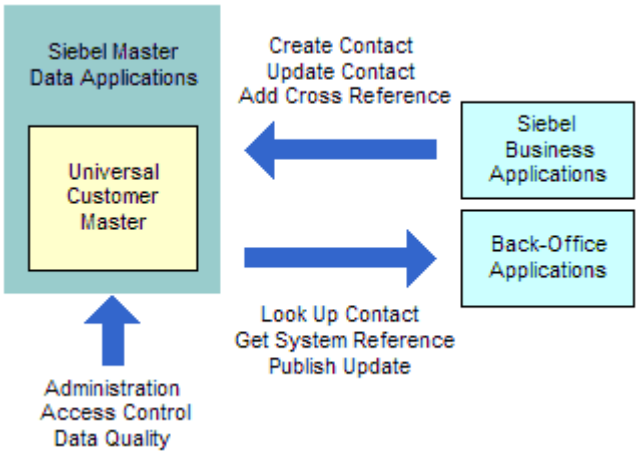

Figure 2. Siebel Master Data Applications Framework

- For information about the functionality included with Siebel Master Data Applications, see "About Siebel Master Data Applications Functions and Services" on page 22.
- For information about the administrative functionality of Siebel Master Data Applications, see "Administering Siebel Master Data Applications" on page 23.
- For information about the technical integration aspects used with Siebel Master Data Applications, see "Siebel Connector and Integration of Siebel Master Data Applications" on page 30.

About Siebel Master Data Applications Functions and Services

The database tables and Siebel Connector for Master Data Applications support insert, delete, update, and query functionality on master data, such as customers, products, and so on. When these functions are available in Siebel Business Applications and back-office systems, they can be used to make modifications to an organization's data, which is then stored and reconciled in Siebel Master Data Applications. For further information on Siebel Connector for Master Data Applications, see "About Integration Services for Siebel Master Data Applications" on page 24 and "Siebel Connector and Integration of Siebel Master Data Applications" on page 30.

For details about prebuilt Siebel Master Data Application services interfaces for accessing these functions, see Chapter 7, "Siebel Master Data Applications Messages."

Siebel Master Data Applications also provide the following services through configuration:

- **Publish.** Publish data records from Siebel Master Data Applications to subscribing applications. For further information on administering this functionality, see "Setting System Privileges for Siebel Master Data Applications" on page 69.
- **Request and Reply.** Other applications send a request for data from Siebel Master Data Applications. For further information on configuring request and reply, see "Setting System Privileges for Siebel Master Data Applications" on page 69.

Services can be configured to operate in real-time or batch mode.

Siebel Master Data Applications use a variety of integration techniques to allow the operation of functions and services between other applications and themselves. See "About Siebel Master Data Applications Architecture" on page 25 for further information.

Administering Siebel Master Data Applications

Siebel Master Data Applications are administered through Siebel Customer Data Steward user interface screens. However, only those screens necessary for viewing and administering the information contained within Siebel Master Data Applications are included. See Chapter 6, "Administering Siebel Master Data Applications" for further information on administration. Only administrative users have access to these screens. An example of one of the administration screens appears in Figure 3.

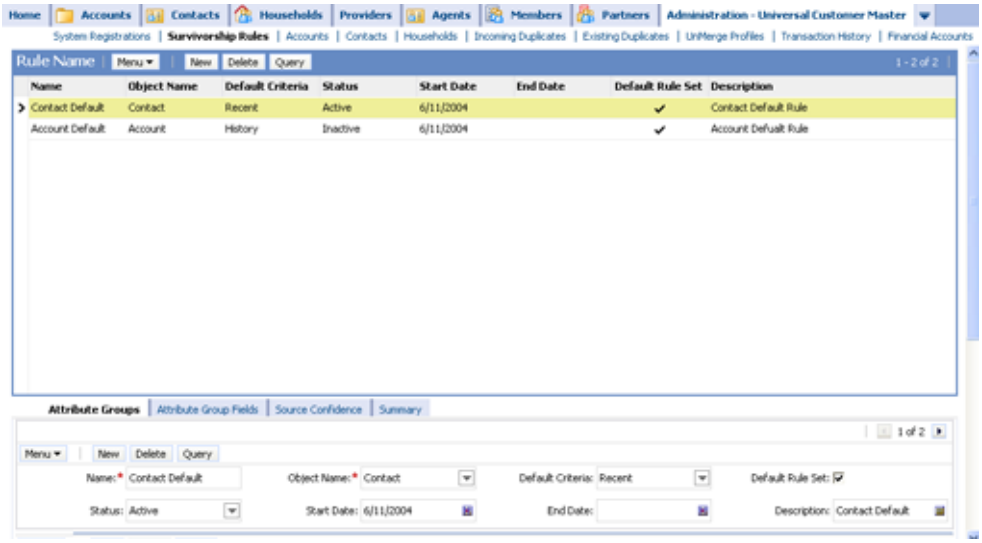

Figure 3. Siebel Business Applications Administration—Universal Customer Master screen.

The administration screens are used to make manual changes to the data, make decisions on suspect information, and set-up rules and policies for UCM data management.

Access Control is enforced at the user interface level for Siebel Master Data Applications administration screens. Siebel Universal Customer Master Application messages have full visibility of the data, that is, no access control mechanism is applied to the messages but they are subject to system privilege checks when accessing the Universal Customer Master Application. For further information and procedures on configuring system privileges, see "Setting System Privileges for Siebel Master Data Applications" on page 69.

For general information and procedures on access control, see *Configuring Siebel Business Applications* and *Siebel Security Guide*.

About Integration Services for Siebel Master Data Applications

The Siebel Connector for Master Data Applications provides the integration services for Siebel Master Data Applications, and is a configurable set of components, which you can use to exchange data between Siebel Master Data Applications and external applications and databases.

The Siebel Connector for Master Data Applications is responsible for receiving, parsing, and executing the business processes specified in the XML message that conforms to the specification detailed in the Siebel Universal Customer Master Application business process specification document. This solution allows you to manage information between Siebel Master Data applications and other applications. The Siebel Connector for Master Data Applications extends Siebel applications to integrate with back-office data and business processes by serving as a master application for all other applications.

The Siebel Connector for Master Data Applications supports both synchronous and asynchronous transactions across application boundaries. The resulting data consistency and the ability to share information allows for efficient coordination among enterprise operations. See Chapter 3, "Siebel Master Data Applications Scenario" for scenarios using the Siebel Master Data Applications.

In addition, the Siebel Connector for Master Data Applications also offers the following capabilities:

- Registration of systems accessing the Universal Customer Master Application
- Enforcement of execution privileges for systems accessing Universal Customer Master Application
- Broadcasting of customer information changes to enterprise systems
- Retrieval of External IDs from the Key Map. For more information see "Cross-Referencing Records with External Systems" on page 84.
- Running of intelligent merge and update
- Running of point-in-time recovery, including unmerge functionality
- Batch data management
- Enabling survivorship rules in UCM

Within the Siebel Master Data Applications, the Siebel Connector offers the following:

- Generation of Universal Unique IDs (UUID)

- Creation of source data and history records

These capabilities are not possible without a predefined message structure to relay required information in exchanged XML messages known as the CRMML Message.

About Siebel Master Data Applications Architecture

Siebel Master Data Applications are based on the Siebel Business Application n-tier object architecture in which the user interface, business logic, and data are separated and layered, see Figure 4. Each tier contains a set of objects and components, which allows a high degree of reuse. The consolidated object layer makes for uniform communication across channels and interfaces. This architecture also allows flexible deployment scenarios and integration with new business processes and systems.

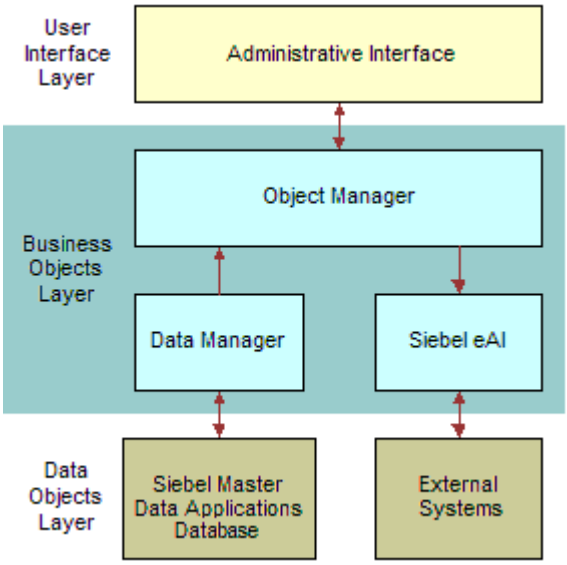

Figure 4. Siebel Master Data Applications n-tier Architecture

The Siebel Master Data Applications platform is designed for extreme fault tolerance with zero downtime. The database platform includes:

- High-availability upgrades and online upgrades of the production environment.
- Enhanced cluster support that allows automatic and transparent fail-over between servers and eliminates single points of failure.
- Database auto-reconnect, which allows servers to continue operation after a temporary outage without any intervention.
- Server component reconfiguration without stopping current tasks or client sessions.
- Supports multithreaded and interactive components, except background mode components.

Siebel Universal Customer Master User Interface Layer

The user interface layer for Siebel Master Data Applications is interactive and customizable. It consists of two parts: the physical user interface layer, which includes templates and tags that render the user interface, and the logical user interface objects layer, which presents the data in the user interface. The physical rendering includes applets, charts, and reports. There are multiple presentation formats, including HTML, XML, and WML. The user interface layer contains no business logic. For further information about the user interface layer, see *Configuring Siebel Business Applications*. The user interface layer is used for the administration of the Siebel Business Information suite; see "Administering Siebel Master Data Applications" on page 23 for further information.

Siebel Universal Customer Master Business Objects Layer

The business logic layer for Siebel Master Data Applications contains object abstractions of entities and represents internal and external data. The logic layer also allows for complex entity relationships. For further information about the business objects layer and how to customize it, see *Configuring Siebel Business Applications*.

The business object manager consists of the Object Manager, Data Manager, and Siebel Master Data Applications Integration.

Object Manager and Data Manager

The object manager and data manager layer use:

- Business Objects
- Business Components
- Virtual Business Components
- External Business Components

These business objects are customizable. For further information about the business objects and data objects layer, see *Configuring Siebel Business Applications*.

Siebel Industry EAI Architecture

Siebel Industry EAI architecture is built to work with the Siebel Enterprise Application Integration (EAI) architecture and to support XML-based messaging communication infrastructure. Different applications require integration using messaging mechanisms. Connectors must be built to support various industry standards.

Siebel Business Applications allow you to build and deploy multiple connectors. For example, the Siebel Financial Services Application has built three connectors—Siebel Connector for Master Data Applications, Siebel Connector for ACORD XML, and Siebel Connector for IFX XML—based on the Siebel Industry EAI framework.

Figure 5 illustrates the high-level architecture of the Siebel Industry EAI and the standard connectors.

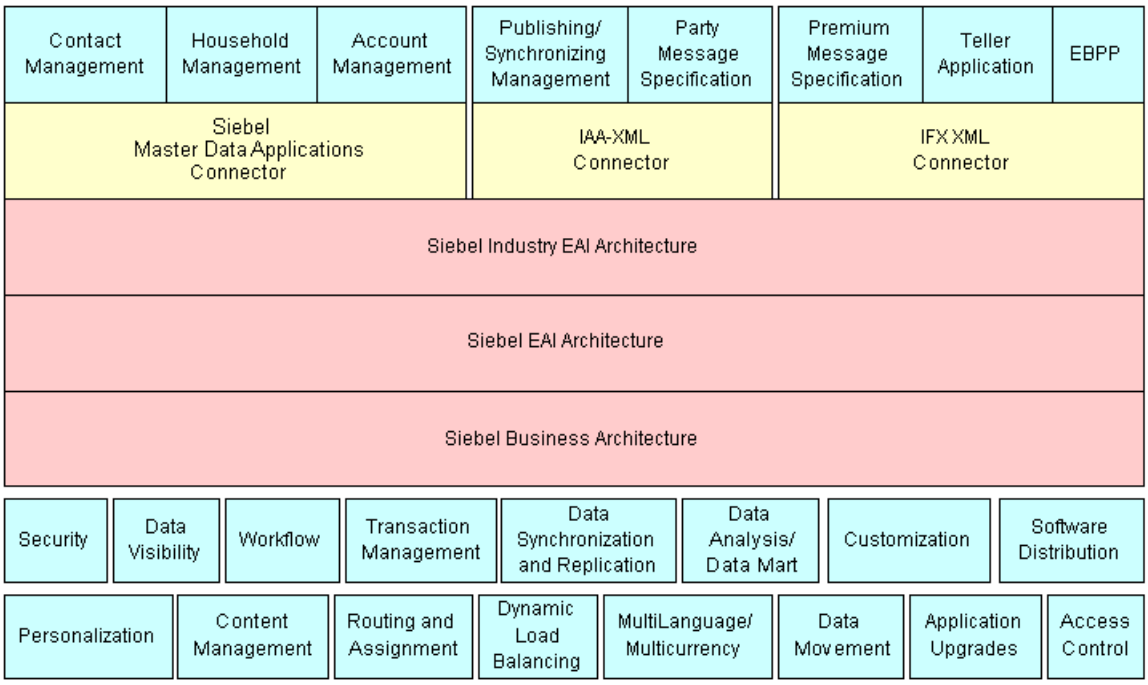

Figure 5. High-Level Architecture of Siebel Industry EAI and Standard Connectors

Siebel Connector for Master Data Applications is based on Siebel XML standards in addition to a predefined Universal Customer Master Application envelope, headers for identifying the sending system, and associated privileges information. It is designed to address the real-time requirement by defining business processes that include both a request and a response message. The Siebel Connector for Master Data Applications provides functions such as:

- Handling message header
- Handling heterogeneous objects in the body section of an XML message
- Security checking for system registration
- Privilege checking for customer business process management
- Publishing customer information changes in both real and batch time processing

The Siebel Connector for Master Data Applications includes several business services. For further information on these services, see "Siebel Connector and Integration of Siebel Master Data Applications" on page 30.

Business Process Flow

Each standard integration or custom integration is based on the creation of business process flows. A business process flow controls the entire business transaction instance. Figure 6 illustrates inbound (Receive-Send) business process flow. Some of the business flows might constitute messages published by Siebel Master Data Applications such as <InsertContactProduct>, <DeleteAccountAddress>, <UpdateContactProduct>, <UpdateHousehold>, <DeleteAccountAddressResponse>, <LookUpAccount>, <LookUpPersonalAddressByState>, and so on. These messages are included in Siebel Master Data Applications for Industry Applications.

The processing flow for each of these business process flows is largely contained within a Siebel workflow process. The workflow process is instantiated by the Business Integration Manager after receiving the inbound request from enterprise applications.

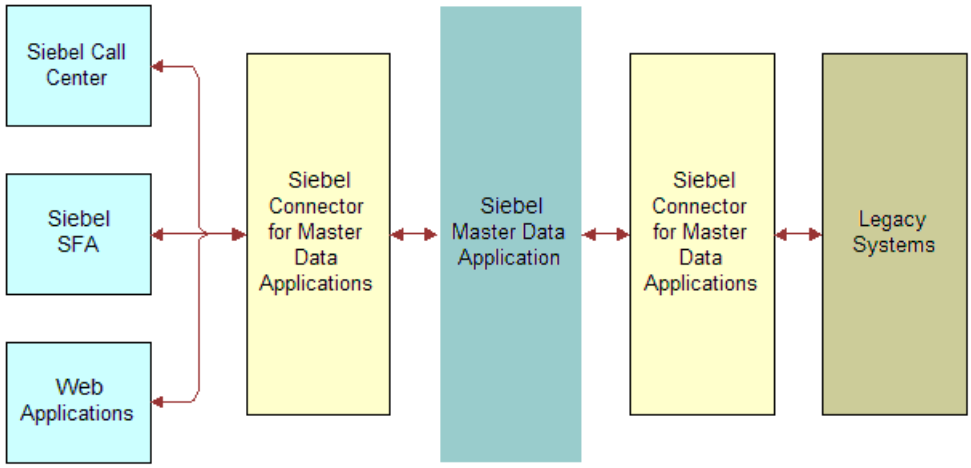

Figure 6. Business Process Flow

Inbound Data Flow

An inbound business process flow starts with a Receiver Server Component, such as the MQSeries or HTTP Receiver. The Receiver runs in the background, continuously waiting for messages to arrive from external applications. After receiving a CRMML message, the receiver then invokes the workflow process configured to handle and process the data.

The workflow dictates the business logic behind the Siebel Connector for Master Data Applications as follows:

1. The raw XML text string passes through XML Hierarchy Converter and is converted into an XML integration object instance.
2. The UCM Dispatcher traverses the XML instance and identifies the messages received according to the rule sets of the UCM Dispatcher Map. It then identifies the envelope, header, and body sections among the hierarchy nodes and sends it to the UCM Security Manager.

   NOTE: The Dispatcher Map is shipped as a part of Siebel UCM Applications. For details, see "UCM Dispatcher Map Integration Objects" on page 63.

3. The UCM Security Manager takes the XML instance, verifies the System for registration with Siebel Master Data Applications, checks the operation identified in each body section for privileges, and attaches any fail privilege bodies to the FailSecurityHierarchy before sending it to the UCM Converter.
4. The UCM Converter takes the XML instance, and processes individual sections of the instance while converting each sub-tree into external integration object instances before sending it to the UCM Transaction Manager.
5. The UCM Transaction Manager performs operations specified in the instance by invoking the services configured in its user properties. Source data is captured for insert, update, and delete operations, and history data is captured for update and delete operations.
   - If the EnableRealtimeInsert user property is set to false, insert requests are queued in the Source Data History table.
   - Update operations invoke UCM Survivorship Engine or UCM Data Quality Manager, if enabled.
6. If UCM Publish/Subscribe Service is configured for real-time publishing, then messages are sent to systems registered for the particular business objects executed.

Workflow Integration

Siebel Business Process Designer is the center of the business data flow. Workflow processes control the flow and transformation of data into and out of the Siebel Business Applications. You create the workflow processes using a graphical user interface provided within Siebel Business Applications, called the Siebel Business Process Designer.

For details on workflow and the Siebel Business Process Designer, see *Siebel Business Process Framework: Workflow Guide.*

Integration Objects

Integration objects are data containers used within the workflow environment. They represent the data structure of a Siebel Business Object or an external application's data object. You create integration objects with the Integration Object Builder provided in Siebel Tools. The Integration Object Builder creates Siebel Integration Objects from Siebel Business Objects, which are then used by components within the Siebel Master Data Applications Integration. Supporting integration objects for Siebel Master Data Applications business processes are shipped as part of the Siebel Master Data Applications.

For more information on Integration Objects, see *Overview: Siebel Enterprise Application Integration*.

Business Services

Business services execute predefined or custom actions in a workflow process. Examples of business services include the UCM Transaction Manager, the EAI Siebel Adapter, the UCM Converter, and so on. These business services act on property sets passed to them. They perform business logic operations such as interfacing with the database, interfacing to external systems, or transforming one integration object into another. Many business services are provided, but you can create your own. Although you can use business services to perform many different functions, they all have a standard interface. Business services have object-like qualities, such as methods, method arguments, and user properties. These elements define how a business service can be used. Business services are defined in Siebel Tools. This guide describes those business services used to interface to external systems within an organization's application network.

For more information on business services in general, *Integration Platform Technologies: Siebel Enterprise Application Integration*.

CRMML Message

Customer Relationship Management Markup Language (CRMML) is a Siebel messaging system targeted for CRM data exchange and remote business process invocation. CRMML allows different applications with different data models to share information, or replicate data through document exchange and cross reference objects. It also allows multiple applications to share business logic and business processes through remote procedure calls.

CRMML is an extensible XML vocabulary. CRMML specification defines a set of syntax and rules to describe how to interpret messages and how to extend the messages beyond the current offer. CRMML messages contain a simple envelope, a header section, and a body section.

For further information on CRMML, see "About CRMML Messages" on page 114.

Siebel Universal Customer Master Data Objects Layer

The Data Objects layer for Siebel Master Data Applications separates other layers from the database and is used only for data storage. It has native SQL and fully dynamic SQL generation. The data layer allows run-time switching between databases, for example, from local to server or testing to production. It auto-adapts to object manager and user interface requirements, and requires zero downtime for major release schema upgrades.

Siebel Master Data Applications support leading databases (DB2 UDB for Windows and UNIX, DB2 UDB for OS/390 and z/OS; Oracle; Microsoft SQL Server) on various platforms. The data layer can scale vertically (with size of server) and horizontally with database parallelization. For further information about the Data Objects layer, see *Configuring Siebel Business Applications*.

Siebel Connector and Integration of Siebel Master Data Applications

Siebel Connector for Master Data Applications describes a set of configurable components for integration among enterprise applications and Siebel Master Data Applications.

The Siebel Connector for Master Data Applications comprises the following prebuilt business services:

- UCM Account Source Data and History Service on page 31
- UCM Batch Manager on page 31
- UCM Contact Source Data and History Service on page 31
- UCM Converter on page 31
- UCM Data Quality Manager on page 32
- UCM Dispatcher on page 32
- UCM FINCORP Account Source Data and History Service on page 32
- UCM Household Source Data and History Service on page 32
- UCM Publish/Subscribe Service on page 32
- UCM Security Manager on page 33
- UCM Survivorship Engine on page 33
- UCM Transaction Manager on page 33
- UCM UI Interaction Service on page 33

Use Siebel Tools to review specific details on business services, including user properties, methods, and method arguments. For information on using Siebel Tools, see *Using Siebel Tools*. See also "About Siebel Master Data Applications Integration Services" on page 40.

UCM Account Source Data and History Service

UCM Account Source Data and History service synchronizes the Account record with the UCM Account Source Data and History record.

UCM Batch Manager

UCM Batch Manager processes insert requests queued in the source data history table in batch mode. It also processes batch data during initial load through EIM.

UCM Contact Source Data and History Service

UCM Contact Source Data and History service synchronizes the Contact record with the UCM Contact Source Data and History record.

UCM Converter

The UCM Converter converts Siebel Master Data Applications messages received to a property set for inbound communication. The converter iterates through the requests and responses in the message to construct error information (if any), and also constructs the envelope and header section of the message.

UCM Data Quality Manager

UCM Data Quality Manager is responsible for cleansing and matching records before processing a transaction.

UCM Dispatcher

The UCM Dispatcher is responsible for receiving and dispatching inbound messages. It receives the inbound message and scans the message for any operations specified in its rule sets dispatcher map, and associates the integration objects for the connector components. It then parses the envelope of the message, converts it to the header property set, and passes the integration objects for the specified action, the envelope layer property set, and the XML message to the connector components for further processing.

UCM FINCORP Account Source Data and History Service

UCM FINCORP Account Source Data and History service synchronizes the FINCORP Account record with the UCM FINCORP Account Source Data and History record.

UCM Household Source Data and History Service

UCM Household Source Data and History service synchronizes the Household record with the UCM Household Source Data and History record.

UCM Publish/Subscribe Service

The UCM Publish/Subscribe Service supports both near real-time and daily-batch publish types.

The near real-time publish module takes the Siebel Connector for Master Data Applications output message, which could include heterogeneous body segments with different business object types, as an input message. It dynamically constructs a unique output message for each system based on the Business Object type that the system registered in the Publish/Subscribe administration view. The near real-time publish module sends the constructed messages to the registered systems through a registered protocol type.

Similarly, the daily-batch publish module publishes the synchronization information of registered Business Object types. The module constructs messages according to Siebel Master Data Applications message specification and sends them to the registered system through registered protocol types.

Both publish modules then update the Last Publish time to the system in the Publish/Subscribe table.

The UCM Publish/Subscribe Service has the following default UserProperties:

- EnableCrossReference[FALSE]
- ReloadSystemsCounter[10]
- WorkflowName[UCM Async Realtime Publish Workflow]

UCM Security Manager

The UCM Security Manager acts as the gatekeeper of the Siebel Universal Customer Master Application. It verifies the System ID in the message header against the System Registrations in the Siebel Universal Customer Master Application before the system can perform its desired operations. After a system passes the registration verification process, the UCM Security Manager cycles through the heterogeneous commands in the body. The UCM Security Manager checks the privilege of the individual operations in the body sections against the privileges in the Universal Customer Master for the particular object and operation. Any fail operation instances in the body are removed from the XMLHierarchy and attached to the FailSecurityHierarchy.

UCM Survivorship Engine

UCM Survivorship Engine is a prebuilt business service to execute survivorship logic defined by data administrators. For background information on administering survivorship, see "About UCM Survivorship Rules" on page 76.

UCM Transaction Manager

The UCM Transaction Manager executes operations specified in Universal Customer Master Application message instances as Siebel database transactions. The UCM Transaction Manager also:

- Evaluates heterogeneous commands and executes the transactions.
- Invokes business services, including Siebel EAI, that are configured in its user properties. These business services can be invoked multiple times.
- Translates Universal Customer Master Application command elements to Siebel Adapter actions and combines return results as a single property set.

UCM UI Interaction Service

UCM UI Interaction Service creates a UUID during insert process. It captures attribute group data for survivorship during the user interface insert or upsert. It also captures the source data history records during the update or delete process.

Transport Adapter

Transport Adapter is a prebuilt business service providing an interface between Siebel Business Applications and external applications. Transports allow Siebel Master Data Applications to exchange data with external applications using standard technologies for both synchronous and asynchronous communication protocols. Transports provide connectivity to virtually any communication protocol that can represent data as text or binary messages, including MQSeries from IBM, and HTTP.

For details on transport adapters, see *Transports and Interfaces: Siebel Enterprise Application Integration*.

Siebel EIM and Integration of Siebel Master Data Applications

Siebel EIM is an important component of Siebel Master Data Applications. It bulk imports and exports data residing in other back-office systems into Siebel Master Data Applications and consolidates and replicates information from these sources. Unlike Siebel EAI, Siebel EIM does not go through a application object manager, but the data moves from the EIM_UCM_ORG interface table directly into the SDH (S_UCM_ORG) tables. Siebel EIM also provides a schema abstraction and has high throughput. Some features of Siebel EIM include:

- Denormalized interface tables
- Declarative mapping defined in Siebel Tools
- Automated SQL generation
- Support for parallelized import

For batch integration with external applications, Siebel UCM provides the following UCM-specific EIM tables to facilitate the loading of selected master objects into UCM's Source Data and History tables. These records can then be processed by UCM data management tasks at a later date:

- EIM_PTYUCM_DTL
- EIM_UCM_ADRPER
- EIM_UCM_ASSET
- EIM_UCM_ASTCON
- EIM_UCM_CON
- EIM_UCM_CONCHLD
- EIM_UCM_OGPCHD
- EIM_UCM_ORG
- EIM_UCM_ORGCHLD
- EIM_UCM_ORGGRP
- EIM_UCM_PRIVCY

For further information about Siebel EIM, see *Siebel Enterprise Integration Manager Administration Guide*.

Generation of Party_UID

Siebel EIM and Siebel Master Data Applications workflow processes generate the Party_UID field based on the value in the UCM_EXT_ID field. The process for generating this field occurs by either of the following means:

- **Loading data to UCM SDH table using UCM Customer Profile Integration Process**. In this scenario, the user must populate the <ID> integration object component field value in the CRMML message, which maps to UCM_EXT_ID field in the UCM SDH table. Additionally, the <PartyUId> field, which maps to PARTY_UID, must be empty or not sent with the CRMML message. The UCM Customer Profile Integration Process then generates the PARTY_UID based on UCM_EXT_ID.

- **Loading data to UCM SDH table using Siebel EIM**. In this scenario, users must populate the UCM_EXT_ID column value in the EIM interface table and the UCM_PARTY_UID field value must remain empty. The EIM interface table value populates the UCM_EXT_ID field in the SDH table. The UCM Batch Data Management workflow then generates the PARTY_UID based on the UCM_EXT_ID.

For further information on the Party_UID, see *Siebel Enterprise Integration Manager Administration Guide*.

About Siebel Master Data Application Workflows

Workflows are a feature of Siebel Master Data Applications. They automate the business processes associated with managing data stored in the master data application.

You can modify these workflows to suit your own business model using Siebel Business Process Designer. For more information, see *Siebel Business Process Framework: Workflow Guide*.

To view the diagrams of the following workflows, use the Workflow Process Designer view in Administration—Business Process screen.

The following workflows describe the business processes for the Universal Customer Master (UCM):

- "UCM Account Batch Data Management Workflow" on page 35
- "UCM Async Realtime Publish Workflow" on page 35
- "UCM Batch Process" on page 36
- "UCM Batch Process - Single step" on page 36
- "UCM Contact Batch Data Management Workflow" on page 36
- "UCM Customer Profile Integration Process" on page 36
- "UCM Customer Profile Integration Process (File Transport Test Workflow)" on page 36
- "UCM Daily Publish" on page 37

UCM Account Batch Data Management Workflow

This workflow cleanses and matches account data. Based on the quality of the data, the workflow marks the record as requiring a review or inserts the record into the base tables for publishing.

UCM Async Realtime Publish Workflow

This workflow is invoked as part of a server request to perform asynchronous real-time publishing. It is called from either the connector workflow (for example, UCM Customer Profile Integration Process) or Batch Data Management workflow (UCM Batch Process) when the real-time Publish/Subscribe flag is true.

In this workflow, the second step, RealTimePublish, calls the UCM Publish/Subscribe service with the method RealTimePublishMethod. This method of UCM Publish/Subscribe Business Service handles the actual publishing of the message to the registered systems. Real-time, asynchronous, publishing increases the performance for both the Connector workflow and the batch data management workflow.

UCM Batch Process

This workflow mainly calls the UCM Batch Manager component (alias UCMBatchProcess) to process the queued Insert requests. It looks for queued records in the Source Data History table, then calls UCM Data Quality Manager to cleanse and match the records. Depending on the match results, it then calls Transaction Manager to execute the insert or update operation, or store the cleansed version for later deduplication processing.

This workflow performs the same functionality as "UCM Batch Process - Single step", and it exposes each service call as individual workflow steps to allow easier debugging.

UCM Batch Process - Single step

This workflow mainly calls the UCM Batch Manager component (alias UCMBatchProcess) to process the queued Insert requests. It looks for queued records in the Source Data History table, then calls UCM Data Quality Manager to cleanse and match the records. Depending on the match results, it then calls Transaction Manager to execute the insert or update operation, or store the cleansed version for later deduplication processing.

UCM Contact Batch Data Management Workflow

This workflow cleanses and matches contact data. Based on the quality of the data, the workflow marks the record as requiring a review or inserts the record into the base tables for publishing.

UCM Customer Profile Integration Process

This workflow receives data as a CRMML message, which differs the UCM Batch Process workflow in that it does not go through the SDH tables. It also handles errors.

UCM Customer Profile Integration Process (File Transport Test Workflow)

This workflow receives data as a CRMML message, which differs the UCM Batch Process workflow in that it does not go through the SDH tables. It also handles reading the request message from a file and writing the response message to a file.

UCM Daily Publish

This workflow calls the UCM Publish Subscribe business service's Publish Method to work in Batch mode. It picks up the updates and new records that each external system (whose "Publish Frequency" = "Daily Batch") is configured to receive within a given (start and end) date time interval after the last publish date time.

The user property ReloadSystemsCounter-10 (default value) is used by Pub sub in batch mode so as to reload the system configuration information after the set number of iterations.

About Siebel Universal Customer Master Server Components

The Siebel Universal Customer Master (UCM) has one server component group and three server components to administer the functionality of the Siebel UCM. The component group's name is Siebel Universal Customer Master (alias UCM). The UCM component group contains the following three server components:

Server Component	Alias	Description
UCM Object Manager	UCMObjMgr	An interactive-mode server component, the UCM Object Manager is the application object manager that administers all requests and data management tasks to the UCM.
UCM Batch Manager	UCMBatchProcess	A batch-mode server component that manages batch invocation of Customer Data Management for account and contact insert requests that have been queued in source data history table; this server component utilizes the UCM Batch Process workflow.
UCM Batch Publish Subscribe	UCMBatchPubSub	A batch-mode server component that manages the batch publishing of new and updated records to subscribing external systems.

For background information on configuring and managing server components, see *Siebel System Administration Guide*.

About the Universally Unique Identifier (UUID)

The Common Key Infrastructure provides functionality to create a universally unique identifier (UUID) for each new record created in the Siebel Universal Customer Master Application. The key is unique across multiple Siebel Applications as well as other applications connected to the Siebel Universal Customer Master Application.

5 Configuring Siebel Master Data Applications

This chapter describes background information and configuration procedures for Siebel Master Data Applications.

The following topics describe procedures for configuring the Siebel application with Siebel Tools. Review background information on configuring Siebel applications, using Siebel Tools, and configuring integration objects in the following documentation:

- *Configuring Siebel Business Applications*
- *Using Siebel Tools*
- *Integration Platform Technologies: Siebel Enterprise Application Integration*

This chapter includes configuration topics in the following areas:

- Loading Data into Siebel Universal Customer Master and Updating External Systems on page 39
- Activating Siebel Universal Customer Master External Systems on page 39
- Configuring Siebel Master Data Application Features on page 40
- Configuring Siebel Master Data Integration Objects on page 61

Loading Data into Siebel Universal Customer Master and Updating External Systems

After installation of Siebel Universal Customer Master (Siebel UCM), the initial customer data must be loaded into the database. In preparation for the this process, it is recommended to:

- cleansing the data in each external system prior to loading data
- loading best or highest quality data first

Use Siebel Enterprise Integration Manager (Siebel EIM) to load bulk data records into the Siebel UCM. For further information on Siebel EIM, see "Siebel EIM and Integration of Siebel Master Data Applications" on page 34 and *Siebel Enterprise Integration Manager Administration Guide*.

Activating Siebel Universal Customer Master External Systems

After loading the customer data into the Siebel Universal Customer Master (Siebel UCM), you must register and activate external systems that access the customer master data.

For information on these procedures, see:

- "Registering Systems Connected to Siebel Master Data Applications" on page 67
- "Setting System Privileges for Siebel Master Data Applications" on page 69

Additionally, other Siebel UCM application features may require configuration or activation.

For further information, see:

- "Configuring Siebel Master Data Application Features" on page 40
- Chapter 6, "Administering Siebel Master Data Applications"

To further customize the Siebel MDA, see "Configuring Siebel Master Data Integration Objects" on page 61.

Configuring Siebel Master Data Application Features

This topic contains a list of general application feature configurations you can perform after installing and initially configuring Siebel Master Data Applications.

- "About Siebel Master Data Applications Integration Services" on page 40
- "Configuring UCM Transaction Manager for Data Management and Survivorship" on page 49
- "Configuring UCM Auto Merge and Manual Review Threshold Values" on page 49
- "Configuring UCM Survivorship Engine Properties" on page 50
- "About UCM Data Management Services" on page 50
- "Configuring UCM Data Quality Manager for Exact Match" on page 51
- "Configuring UCM Server Components for Data Management and Survivorship" on page 53
- "Configuring Run-time Events for UI Interaction" on page 54

See the following related topics for further information on other areas of application configuration and the installation of Siebel MDA.

Related Topics

"Configuring Siebel Master Data Integration Objects" on page 61

About Siebel Master Data Applications Integration Services

Siebel Master Data Applications (Siebel MDA) provide an object called a business service, which you can reuse in multiple applications. These business services together perform the insert, update, query, and delete operations on Siebel MDA data. You can use these business services together in the prebuilt workflow processes or build your own workflow processes or business services.

The Siebel Connector for the Master Data Applications provides the following prebuilt business services that you can configure to meet your business requirements:

- UCM Transaction Manager
- UCM Converter
- UCM Dispatcher
- UCM Security Manager
- UCM Publish/Subscribe Service
- UCM Batch Manager
- UCM Data Quality Manager
- UCM Survivorship Engine
- UCM Account Source Data and History Service
- UCM Contact Source Data and History Service
- UCM UI Interaction Service
- UCM FINCORP Account Source Data and History Service
- UCM Household Source Data and History Service

You can configure business services by manipulating their user properties, or you can create your own business service in Siebel Tools. For information on using Siebel Tools to configure your application, see *Configuring Siebel Business Applications*.

NOTE: After you have configured your business services to accomplish the tasks required for your business scenario, you need to compile the business service to include the new information in your Siebel repository file (`.srf`). Then copy the `.srf` file to the directory where your Siebel Servers can access it.

For further information on integration services and the Siebel MDA architecture, see "About Integration Services for Siebel Master Data Applications" on page 24 and other topics in Chapter 4, "Architecture and Framework."

UCM Transaction Manager

You can extend this business service by manipulating its Operation user property. The basic format for value entry for each operation is as follows:

CIFOperation_Query

Service/Method/Argument;Argument; or /Method/Argument;Argument;

- Service, Method, and Argument are separated by a forward slash (/).
- Each Argument ends with a semi-colon (;).

CIFOperation_XMLQuery

EAI Siebel Adapter/Query/#XMLHierarchy;

- The default Service name is EAI Siebel Adapter and the default argument name is SiebelMessage.

- SiebelMessage indicates turning off the SiebelMessage.
- XMLHierarchy indicates replacing SiebelMessage with XMLHierarchy.

CIFOperation_GetValue

FINS Industry/BC Facility Service/HierarchySearchSpec/!SiebelMessage;A=>B;

- A=>B means getting argument value of A from argument value of B where argument B is an argument of the Connector Integration Object Instance.

Table 5 lists the user properties examples for the UCM Transaction Manager.

Table 5. User Properties for UCM Transaction Manager

Operation Name	Value
Account Address Field 1	Street Address;Street Address
Account Address Field 2	City;City
Account Address Field 3	State;State
Account Address Field 4	Postal Code;Postal Code
Account Address Field 5	Country;Country
Account Address Primary CUT Address	Primary Address Id;Account_Business Address
Contact Address Field 1	INS Personal Street Address;Street Address
Contact Address Field 2	INS Personal City;City
Contact Address Field 3	INS Personal State;State
Contact Address Field 4	INS Personal Postal Code;Postal Code
Contact Address Field 5	INS Personal Country;Country
Contact Address Primary Personal Address	Primary Personal Id;Contact_INS Personal Address
DispatcherMapName	CIFDispMap
Enable Updating SDH Type on Error	TRUE
EnableAgentLock	TRUE
Household Address Field 1	Street Address;Street Address
Household Address Field 2	City;City
Household Address Field 3	State;State
Household Address Field 4	Postal Code;Postal Code
Household Address Field 5	Country;Country
Household Address Primary CUT Address	Primary Household Address Id;Household_Household Address

Table 5. User Properties for UCM Transaction Manager

Operation Name	Value
Insert_IOandOp_Account	CIFAccountInsert;CIFAccountInsertRs;IXMLOperation_ADD
Insert_IOandOp_Contact	CIFContactInsert;CIFContactInsertRs;IXMLOperation_ADD
Insert_IOandOp_FINCORP Account	CIFFINCORPAccountInsert;CIFFINCORPAccountInsertRs;IXMLOperation_ADD
Insert_IOandOp_Household	CIFHouseholdInsert;CIFHouseholdInsertRs;IXMLOperation_ADD
IONameForQueryBV_Account	CIFAccoun
IONameForQueryBV_Contact	CIFContact
IONameForQueryBV_FINCORP Account	CIFFINCORPAccount
IONameForQueryBV_Household	CIFHousehold
IXMLOperation_ADD	EAI Siebel Adapter/Insert/
IXMLOperation_DELETE	EAI Siebel Adapter/Delete/
IXMLOperation_QUERY	EAI Siebel Adapter/Query/
IXMLOperation_QUERY_BY_ID	EAI Siebel Adapter/Query/ PrimaryRowId;!SiebelMessage;
IXMLOperation_QUERYPAGE	EAI Siebel Adapter/QueryPage/ NewQuery=>IXML_NewQuery;SearchSpec=>SearchSpec;StartRowNum=>IXML_StartRowNum;PageSize=>IXML_PageSize;SERVICE_VALUE_RETURN;$LastPage=>IXML_LastPage;$NumOutputObjects=>IXML_NumOutputObjects;
IXMLOperation_SYNCH	EAI Siebel Adapter/Synchronize/
IXMLOperation_UPDATE	EAI Siebel Adapter/Update
IXMLOperation_UPSERT	EAI Siebel Adapter/Upsert/
Operation Delete	IXMLOperation_DELETE
Operation Insert	IXMLOperation_ADD
Operation Update	IXMLOperation_UPDATE
Operation Upsert	IXMLOperation_UPSERT
Primary Object 1	Address/CIFPersonalAddress
RealtimeInsertAutoMatch_Account	CIFAccountUpdate;CIFAccountUpdateRs;IXMLOperation_UPDATE
RealtimeInsertAutoMatch_Contact	CIFContactUpdate;CIFContactUpdateRs;IXMLOperation_UPDATE

Table 5. User Properties for UCM Transaction Manager

Operation Name	Value
RealtimeInsertAutoMatch_Household	CIFHouseholdUpdate;CIFHouseholdUpdateRs;IXMLOperation_UPDATE
SDHCommitSize	5
Update_IOandOp_Account	CIFAccountUpdate;CIFAccountUpdateRs;IXMLOperation_UPDATE
Update_IOandOp_Contact	CIFContactUpdate;CIFContactUpdateRs;IXMLOperation_UPDATE
Update_IOandOp_FINCORP Account	CIFFINCORPAccountUpdate;CIFFINCORPAccountUpdateRs;IXMLOperation_UPDATE
Update_IOandOp_Household	CIFHouseholdUpdate;CIFHouseholdUpdateRs;IXMLOperation_UPDATE
Upsert_IOandOp_Account	CIFAccountUpsert;CIFAccountUpsertRs;IXMLOperation_UPSERT
Upsert_IOandOp_Contact	CIFContactUpsert;CIFContactUpsertRs;IXMLOperation_UPSERT
Upsert_IOandOp_FINCORP Account	CIFFINCORPAccountUpsert;CIFFINCORPAccountUpsertRs;IXMLOperation_UPSERT
Upsert_IOandOp_Household	CIFHouseholdUpsert;CIFHouseholdUpsertRs;IXMLOperation_UPSERT

UCM Converter

Table 6 displays the only user property you can configure for this business service. This value appears in the preheader section of your CRMML message.

Table 6. User Property for UCM Converter

Name	Value
XMLEnvIntObjectName	Name of the envelope integration object shipped or customized by you.

The UCM Converter uses the hierarchy represented in the UCM integration object to guide the message through the converting process. If the integration object instance receives elements that do not have a definition defined in the integration object definition, the converter errors out. If you expect such a situation, you can set the Ignore Undefined XML Tag parameter on the user property of the corresponding integration object.

NOTE: This user property is created by the wizard and is set to Y. You can turn it off if you want the converter to error out.

UCM Dispatcher

You can modify both user properties for this business service as shown in Table 7.

Table 7. User Properties for UCM Dispatcher

Name	Value
DispatcherMapName	Name of the dispatcher map shipped or customized by you.
XMLEnvIntObjectName	Name of the envelope integration object shipped or customized by you.

UCM Security Manager

You can modify or extend this business service by manipulating its user properties. The Name column stands for the UCM Transaction Manager Operation Name, while the Value column represents the Field Name for different types of privileges in the System Privileges View. Table 8 displays these user properties.

Table 8. User Properties for UCM Security Manager

Name	Value
IXMLOperation_ADD	Insert
IXMLOperation_DELETE	Delete
IXMLOperation_QUERY	Query
IXMLOperation_QUERYPAGE	Query
IXMLOperation_UPDATE	Update
IXMLOperation_UPSERT	Upsert

UCM Data Quality Manager

The user properties for this business service appear in Table 9.

Table 9. User Properties for UCM Data Quality Manager

Name	Value
Account Account Cleansing Field 1	Name:Name;Location:Location;
Account Account Matching Field 1	Name:Name;DUNS Number:DUNS Number;
Account Account_Business Address Cleansing Field 1	Street Address:Street Address;City:City;State:State;Postal Code:Postal Code;Country:Country;

Table 9. User Properties for UCM Data Quality Manager

Name	Value
Account Account_Business Address Matching Field 1	Street Address:Primary Account Street Address;City:Primary Account City;State:Primary Account State;Postal Code:Primary Account Postal Code;Country:Primary Account Country;
Account Auto Threshold	90
Account Cleansing Component 1	Account:Account;Account_Business Address:CUT Address;
Account Manual Threshold	70
Account Matching Component 1	Account:Account;Primary Account_Business Address;
Contact Auto Threshold	90
Contact Cleansing Component 1	Contact:Contact;Contact_INS Personal Address:Personal Address;Contact_Account:Account;
Contact Contact Cleansing Field 1	Last Name:Last Name;First Name:First Name;Middle Name:Middle Name;Job Title:Job Title;
Contact Contact Matching Field 1	Last Name:Last Name;First Name:First Name;Middle Name:Middle Name;Work Phone #:Work Phone #;Cellular Phone #:Cellular Phone #;Home Phone #:Home Phone #;
Contact Contact Matching Field 2	Email Address:Email Address;Social Security Number:Social Security Number;Birth Date:Birth Date;
Contact Contact_Account Cleansing Field 1	Account:Name;Account Location:Location;
Contact Contact_Account Matching Field 1	Account:Account;
Contact Contact_INS Personal Address Cleansing Field 1	INS Personal Street Address:Street Address;INS Personal City:City;INS Personal State:State;INS Personal Postal Code:Postal Code;INS Personal Country:Country;
Contact Contact_INS Personal Address Matching Field 1	INS Personal Street Address:Primary Personal Street Address;INS Personal City:Primary Personal City;INS Personal State:Primary Personal State;INS Personal Postal Code:Primary Personal Postal Code;INS Personal Country:Primary Personal Country;
Contact Manual Threshold	70

Table 9. User Properties for UCM Data Quality Manager

Name	Value
Contact Matching Component 1	Contact:Contact;Primary Contact_INS Personal Address;Contact_Account;
Enable XRef Match	false
ExactMatch Object 1	Account:CIFAccount;Contact:CIFContact;

UCM Batch Manager

The user properties for this business service appear in Table 10.

Table 10. User Properties for UCM Batch Manager

Name	Value
Account AutoMatch	CIFAccountUpdate;CIFAccountUpdateRs;IXMLOperation_UPDATE
Account NoMatch	CIFAccountInsert;CIFAccountInsertRs;IXMLOperation_ADD
Contact AutoMatch	CIFContactUpdate;CIFContactUpdateRs;IXMLOperation_UPDATE
Contact NoMatch	CIFContactInsert;CIFContactInsertRs;IXMLOperation_ADD
FINCORP Account NoMatch	CIFFINCORPAccountInsert;CIFFINCORPAccountInsertRs;IXMLOperation_ADD
Household AutoMatch	CIFHouseholdUpdate;CIFHouseholdUpdateRs;IXMLOperation_UPDATE
Household NoMatch	CIFHouseholdInsert;CIFHouseholdInsertRs;IXMLOperation_ADD

UCM Survivorship Engine

The user properties for this business service appear in Table 11.

Table 11. User Properties for UCM Survivorship Engine

Name	Value
DefaultAttrGrpName_Account	Default Account Attribute Group
DefaultAttrGrpName_Contact	Default Contact Attribute Group
ExemptFields_Account	Party UId;Id
ExemptFields_Contact	Party UId;Person UId;Id

UCM Account Source Data and History Service

The user properties for this business service appear in Table 12.

Table 12. User Properties for UCM Account Source Data and History Service

Name	Value
Application Services Interface	Y
Internal Integration Object	UCMAccountSourceDataAndHistory

UCM Contact Source Data and History Service

The user properties for this business service appear in Table 13.

Table 13. User Properties for UCM Contact Data and History Service

Name	Value
Application Services Interface	Y
Internal Integration Object	UCMContactSourceDataAndHistory

UCM FINCORP Account Source Data and History Service

The user properties for this business service appear in Table 14.

Table 14. User Properties for UCM FINCORP Account Source Data and History Service

Name	Value
Application Services Interface	Y
Internal Integration Object	UCMFINCORPAccountSourceDataAndHistory

UCM Household Source Data and History Service

The user properties for this business service appear in Table 15.

Table 15. User Properties for UCM Household Source Data and History Service

Name	Value
Application Services Interface	Y
Internal Integration Object	UCMHouseholdSourceDataAndHistory

Configuring UCM Transaction Manager for Data Management and Survivorship

Enable customer data management or the survivorship engine for UCM Transaction Manager business service by configuring its user properties:

For details on configuring business service user properties using Siebel Tools, see *Integration Platform Technologies: Siebel Enterprise Application Integration*.

For background information on UCM Transaction Manager business service, see "UCM Transaction Manager" on page 41.

To configure UCM Transaction Manager to Enable Data Management and Survivorship Engine

- Configure the following input arguments for the UCM Transaction Manager:

Input Arguments	Default Value
TurnOnCDMCleanse	FALSE
TurnOnCDMExactMatch	FALSE
TurnOnCDMMatch	FALSE

Configuring UCM Auto Merge and Manual Review Threshold Values

Configure Universal Customer Master (UCM) auto merge and manual review threshold functionality by configuring user properties for the UCM Data Quality Manager business service.

The auto threshold value determines which incoming records are merged into existing matching best version records. The manual threshold determines the value (between auto merge and manual review value) at which incoming records are shown to administrators in Data Management screens for review. For information on manually reviewing records in the data management screen, see "Managing Pending Updates from External Systems" on page 88.

For details on configuring business service user properties using Siebel Tools, see *Integration Platform Technologies: Siebel Enterprise Application Integration*.

For background information on UCM Data Quality Manager business service, see "UCM Data Quality Manager" on page 45.

To configure auto merge and manual review thresholds

- Configure the following user properties for the UCM Data Quality Manager business service:

User Property	Value
Account Auto Threshold	90
Account Manual Threshold	70
Contact Auto Threshold	90
Contact Manual Threshold	70

Configuring UCM Survivorship Engine Properties

Configure the following Universal Customer Master (UCM) survivorship functionality by configuring user properties in UCM Survivorship Engine business service:

- Default attribute group names
- Fields exempt by survivorship rules engine

For details on configuring business service user properties using Siebel Tools, see *Integration Platform Technologies: Siebel Enterprise Application Integration*.

For background information on UCM Survivorship Engine business service, see "UCM Survivorship Engine" on page 47.

To configure survivorship properties

- Configure the following user properties for the UCM Survivorship Engine business service:

User Property	Default Value
DefaultAttrGrpName_Account	Default Account Attribute Group
DefaultAttrGrpName_Contact	Default Contact Attribute Group
ExemptFields_Account	Party UId; Id
ExemptFields_Contact	Party UId; Person UId; Id

About UCM Data Management Services

Siebel Universal Customer Master (Siebel UCM) provides the following data management services to cleanse, identify, and link incoming source data to the master data records. These services are invoked through the Siebel Enterprise Application Integration (Siebel EAI) interface and can process single or multiple records. Records updated directly within Siebel UCM application also trigger similar data management services but do not trigger survivorship rules.

- Account Data Management Service

- Contact Data Management Service

These services take an incoming Account or Contact record and:

- Check the system privilege of the application submitting the record
- Cross-reference the record if external account ID is provided
- Create the UUID for the record if the record is new
- Call the data cleansing engine to standardize and validate account or contact name and address
- Perform an exact match process based on configured parameter (default is external account ID for account records; external contact ID for contact records).
- If no exact match found, call data matching engine to perform fuzzy matching (default is Account name for account records; default is Contact first name and last name for contact records) to further identify possible duplication.

NOTE: Siebel UCM provides embedded data matching and cleansing third-party software (SSA). The Siebel Data Quality module is licensed separately from Siebel UCM. See *Siebel Data Quality Administration Guide* and your third-party data quality provider documentation for further information. See also,"Configuring Siebel Data Quality Cleansing for Siebel UCM" on page 92.

Depending on the matching score returned from the data matching engine, UCM can create a new record, link to an existing record, or store the incoming record for further investigation by the data steward. By default the data matching engine returns two threshold scores, an auto threshold score (upper) and manual threshold scores (lower), numbered on a scale of 0-100. These threshold numbers are configurable using Siebel Tools. The matching score results in the following three scenarios:

- When the incoming record is matched above the upper threshold, Siebel UCM applies survivorship rules to merge the incoming record with an existing record in Siebel UCM. For information on survivorship and details on setting up the rules, see "Process of Creating Survivorship Rules" on page 77. When the incoming record is matched below the lower threshold, Siebel UCM creates a new record and publishes a new-record message to other external systems. For information on publishing and subscribing, see "Configuring System Publish and Subscribe for Siebel Master Data Applications" on page 73.
- When the incoming record is matched in between the upper and the lower threshold numbers, Siebel UCM stores the record in the deduplication table (S_UCM_DEDUP) for further investigation. Data stewards can view the pending records in the Incoming Duplicates-Account or Contact screen.
- Configure your matching threshold and the data engine matching rules based on your organization's data quality standards.

Configuring UCM Data Quality Manager for Exact Match

Configure the Universal Customer Master (UCM) exact match functionality by configuring user properties in the UCM Data Quality Manager business service.

For details on configuring business service user properties using Siebel Tools, see *Integration Platform Technologies: Siebel Enterprise Application Integration*.

To configure Siebel UCM exact match

- Configure the following user properties for the UCM Data Quality Manager business service:

User Property	Value
ExactMatch Object	{Name + ":" + IntObjName + ";"}
Name ExactMatch Field	{ICFieldName + ":" + BCFieldName + ";"}

where:

Name—Must be unique; could be integration object name, root component name, or supported objects (for example, Account, Contact).

IntObjName—Integration object name (use for query by user key).

ICFieldName—Integration component field name.

BCFieldName—Business component field name (use for query by example).

Configuring UCM Unmerge Functionality

Configure child objects for Universal Customer Master (UCM) unmerge functionality by configuring user properties for the UCM DeDuplication Results (Account) or UCM DeDuplication Results (Contact) business components. This configuration determines the list of child objects that are reparented to the appropriate record during an unmerge execution.

For background information on configuring business components in Siebel Tools, see *Configuring Siebel Business Applications*.

To configure child objects for UCM unmerge functionality

- Configure the default values for the following business component user properties:

Business Component	User Property	Default Value
UCM DeDuplication Results (Account)	MERGE_SUPPORT_CHILD0	CUT Address;FINCORP Loan Account;FINCORP Deposit Account;FIN Business Phone;UCM Account Source Data and History;CIF Account Reference
UCM DeDuplication Results (Contact)	MERGE_SUPPORT_CHILD0	Personal Address;FINCORP Account;FIN Contact Phone;UCM Contact Source Data and History;CIF Contact Reference

Configuring UCM Server Components for Data Management and Survivorship

Configure the UCM server components UCM Batch Manager and UCM Batch Publish Subscribe for data management and survivorship functionality by setting parameter values for these components. Parameter values can be set in the Server Manager GUI or at the command-line interface. For full details on setting server component parameters, see *Siebel System Administration Guide*.

For background information on these UCM server components, see "About Siebel Universal Customer Master Server Components" on page 37.

To configure UCM Server Components for Data Management and Survivorship

- Configure the following parameters for the appropriate UCM server component:

Parameter	Alias	Default Value	Description
UCM Batch Object Type	UCMBatchObjectType	Contact	Object Type for UCM Batch Pub Sub or UCM Batch Manager
UCM Batch Size	UCMBatchSize	10	Page Size for UCM Batch Pub Sub or UCM Batch Manager
UCM Data Management Cleanse Flag	UCMCDMCleanseFlag	FALSE	To Enable UCM Data Management Cleanse Capability
UCM Data Management Exact Match Flag	UCMCDMExactMatchFlag	FALSE	To Enable the UCM Data Management Exact Match Capability
UCM Data Management Match Flag	UCMCDMMatchFlag	FALSE	To Enable the UCM Data Management Match Capability
UCM Publish/Subscribe Flag	UCMPubSubFlag	FALSE	To Enable the UCM Publish/ Subscribe Capability
UCM Search Specification	UCMSearchSpec		Search Specification
UCM Sleep Time	UCMSleepTime	60	Sleep Time (seconds)
UCM Sort Specification	UCMSortSpec		Sort Specification
UCM Survivorship Engine Flag	UCMSurvivorshipEngineFlag	FALSE	To Enable the UCM Survivorship Engine Capability

Configuring Run-time Events for UI Interaction

Creating UCM run-time events in the Administration—Runtime Events screen is necessary to configure enhanced UI interactions. For background information on run-time events and for details on creating these events, see *Siebel Personalization Administration Guide*.

To configure run-time events for UI interaction

1 Navigate to Administration—Runtime Events > Action Sets view, and create the following action sets:

Action Set Name	Activate	Enable Export
UCM WriteRecord	Y	Y
UCM PreWriteRecord	Y	Y
UCM PreDeleteRecord	Y	Y
UCM RecordDeleted	Y	Y

2 For each action set defined, create one new record in the Action Set's more info view with the following information that corresponds with the action set name (for example, the WriteRecord business service method is the record for the UCM WriteRecord action set):

Name	Business Service Name	Business Service Method	Action Type	Sequence
UCM WriteRecord	UCM UI Interaction Service	WriteRecord	BusService	1
UCM PreWriteRecord	UCM UI Interaction Service	PreWriteRecord	BusService	1
UCM PreDeleteRecord	UCM UI Interaction Service	PreDeleteRecord	BusService	1
UCM RecordDeleted	UCM UI Interaction Service	RecordDeleted	BusService	1

3 Navigate to Administration—Runtime Events > Events, and create four events with the following field information:

Object Type	Object Name	Event	Subevent	Action Set Name	Sequence
Applet	Contact List Applet	InvokeMethod	WriteRecord	UCM WriteRecord	1
Applet	Contact List Applet	PreInvokeMethod	WriteRecord	UCM PreWriteRecord	1

Object Type	Object Name	Event	Subevent	Action Set Name	Sequence
Applet	Contact List Applet	InvokeMethod	DeleteRecord	UCM DeleteRecord	1
Applet	Contact List Applet	PreInvokeMethod	DeleteRecord	UCM PreDeleteRecord	1

4 Create the same four events for the following additional screen views (objects):

- SIS Account Entry Applet
- SIS Account List Applet
- Contact Form Applet

Configuring Default System for UI Interaction

A default system must be registered in the System Registration view to enable UI interaction. The Default System is necessary to capture the registered system that last modified the data for future Survivorship Engine evaluation.

To configure a default system for UI interaction

- Register the system in the System Registration view with the following data:
  - System ID = Default System
  - System Name = Default System

  For further information on this task, see "Registering Systems Connected to Siebel Master Data Applications" on page 67.

Configuring UCM Inbound Server Communication

Configuring inbound UCM Siebel Server communications depends on the registered system protocol type for the Siebel Connector for Master Data Applications, either the Siebel EAI MQSeries Transport, the Siebel EAI JMS Transport, or the Siebel EAI HTTP Transport. (For further details on the Siebel Connector for Master Data Applications, see "Siebel Connector and Integration of Siebel Master Data Applications" on page 30.)

The Siebel EAI MQSeries Transport allows you to integrate data between Siebel Business Applications and external applications that can interface with the IBM MQSeries. The EAI MQSeries Server Transport transports messages to and from IBM MQSeries queues. To configure Siebel EAI MQSeries Transport, see "Configuring for Siebel EAI MQSeries Transport" on page 56.

Siebel EAI HTTP Transport allows you to send XML messages over HTTP to a target URL. The Siebel Web Engine (SWE) serves as the transport to receive XML messages sent over the HTTP protocol to a Siebel application. To configure Siebel EAI HTTP Transport, see "Configuring Siebel EAI HTTP Transport" on page 60.

For background information and details on configuring Siebel EAI MQSeries Transport and Siebel EAI HTTP Transport, see *Transports and Interfaces: Siebel Enterprise Application Integration*.

Configuring for Siebel EAI MQSeries Transport

To configure UCM inbound communication using the Siebel EAI MQSeries Transport, you must:

- Create and configure two named subsystems.
- Configure the Siebel Server component MQSeries Server Receiver (alias: MqSeriesSrvRcvr)

NOTE: Make sure Siebel EAI MQSeries Transport is enabled prior to configuring for UCM. For further information on enabling Siebel EAI MQSeries Transport, see *Transports and Interfaces: Siebel Enterprise Application Integration*.

For background information on these Siebel Server administrative tasks, see *Siebel System Administration Guide*.

To configure named subsystems

1. Navigate to the Administration—Server Configuration screen.
2. From the link bar, click Enterprises.
3. Select the Profile Configuration view tab.
4. Create two new records in the Profile Configuration list and provide the following information.

	Record 1	Record 2
Name	Any name, CIFMQConnSubsy for example	Any name, CIFMQDataSubsys for example
Alias	This field is required for creation of this record.	This field is required for creation of this record.
Subsystem Type	MQSeriesServerSubsys	EAITransportDataHandlingSubsys

NOTE: The subsystem type that you select must have a check mark in the Is Named Enabled field.

5 For each record, modify the following parameters in the Profile Parameters list:

Parameter Name	Record 1	Record 2
MQSeries Physical Queue Name	Queue name to receive inbound request message from	N/A
MQSeries Queue Manager Name	Queue manager name who own the queues	N/A
MQSeries Response Physical Queue Name	Queue name to send response message to	N/A
MQSeries Sleep Time	100 (or longer if required)	N/A
Workflow Process to Execute	N/A	UCM Customer Profile Integration Process

6 Save both records.

For details on this procedure and the Named Subsystems, see *Siebel System Administration Guide*.

To configure MQSeries Server Receiver Siebel Server component

1 Navigate to the Administration—Server Configuration screen.

2 From the link bar, click Servers.

3 In the Siebel Servers list, select the Siebel Server of interest.

4 Click the Components view tab.

5 In the Components list, select MQSeries Server Receiver (alias: MqSeriesSrvRcvr).

6 Select the Parameters view tab under the Components list.

7 Set the following parameters as follows:

Parameter Name	Alias	Value
Receiver Connection Subsystem	ReceiverConnectionSubsystem	Subsystem name created in "To configure named subsystems" on page 56. For example, CIFMQConnSubsys or CIFMQDataSubsys.
Receiver Method Name	ReceiverMethodName	ReceiveDispatch or ReceiveDispatchSend.
Default Tasks	DfltTasks	1 or number of tasks desired.

8 Restart the Siebel Server and make sure the MQSeries Server Receiver server component is running.

Configuring for Siebel EAI JMS Transport

Use the Siebel EAI JMS Transport to integrate data between Siebel Business Applications and external applications that can interface using JMS Protocol. The EAI JMS Transport transports messages to and from JMS queues.

To configure UCM inbound communication using the Siebel EAI JMS Transport, you must:

- Create and configure two named subsystems.
- Configure the Siebel Server component JMS Receiver (alias: JMSReceiver)

NOTE: Make sure Siebel EAI JMS Transport is enabled prior to configuring for UCM. For more information on enabling Siebel EAI JMS Transport, see *Transports and Interfaces: Siebel Enterprise Application Integration*.

For background information on these Siebel Server administrative tasks, see *Siebel System Administration Guide*.

To configure named subsystems

1 Navigate to the Administration-Server Configuration screen.

2 From the link bar, click Enterprises.

3 Click the Profile Configuration view tab.

4 Create two new records in the Profile Configuration list and provide the following information.

	Record 1	Record 2
Name	Any name, such as: CIFJMSConnSubsy	CIFJMSDataSubsys
Alias	This field is required for creation of this record	This field is required for creation of this record
Subsystem Type	JMSSubsys	EAITransportDataHandlingSubsys

NOTE: The subsystem type that you select must have a check mark in the Is Named Enabled field.

5 For each record, modify the following parameters in the Profile Parameters list:

Parameter Name	Record 1	Record 2
ConnectionFactory name	Name of Connection Factory	N/A
ConnectionUsername	User Name for the connection	
ConnectionPassword	User Password for the connection	
ReceiveQueue name	Queue name to receive inbound request message from	N/A
SendQueue name	Queue name to send response message to	N/A
JVM Subsystem name	JAVA	N/A
DispatchWorkflowProcess	N/A	UCM Customer Profile Integration Process

NOTE: Make sure that you have JAVA named subsystem created for subsystem JVMSubSys with DLL, Classpath, and VMOPTIONS parameters defined. For information on how to create this subsystem, see *Business Processes and Rules: Siebel Enterprise Application Integration*.

6 Save both records.

For details on this procedure and the Named Subsystems, see *Siebel System Administration Guide*.

To configure JMS Receiver Siebel Server component

1 Navigate to the Administration-Server Configuration screen.

2 From the link bar, click Servers.

3 In the Siebel Servers list, choose a Siebel Server.

4 Click the Components view tab.

5 In the Components list, choose JMS Receiver (alias: JMSReceiver).

6 Click the Parameters view tab under the Components list.

7 Set the following parameters:

Parameter Name	Alias	Value
Receiver Connection Subsystem	ReceiverConnectionSubsystem	Subsystem name created previously. For example: CIFJMSConnSubsy
Receiver Method Name	ReceiverMethodName	ReceiveDispatch or ReceiveDispatchSend

8 Restart the Siebel Server and make sure the JMS Receiver server component is running.

Configuring Siebel EAI HTTP Transport

To configure UCM inbound communication using the Siebel EAI HTTP Transport, you must:

- Configure the Siebel Web Engine (SWE)
- Modify the Siebel Universal Customer Master Configuration file (`.cfg` file) to run the extension service.

NOTE: Make sure Siebel EAI HTTP Transport is enabled prior to configuring for UCM. For further information on enabling Siebel EAI HTTP Transport, see *Transports and Interfaces: Siebel Enterprise Application Integration*.

For background information on these Siebel Server administrative tasks, see *Siebel System Administration Guide*.

To configure the Siebel Web Engine to run the Inbound HTTP Transport

1. Open `eapps_sia.cfg` file in the \bin subdirectory in the install directory.
2. Locate the section `[/ucm_%language%]`, for example `[/ucm_enu]`.
3. Add the EnableExtServiceOnly configuration parameter or set it as follows, if it already exists:

   ```
   [/ucm_enu]

   ConnectString = <Connect String>

   EnableExtServiceOnly = TRUE
   ```

4. Save and close the configuration file.

After creating and configuring the SWE, configure the required Named Subsystem. For more information on this procedure, see *Transports and Interfaces: Siebel Enterprise Application Integration*.

To configure the Siebel Universal Customer Master Application to run the Extension Service

1. Open `ucm.cfg` file in the \bin subdirectory in the install directory.
2. Locate the section `[/HTTP Services]`, or add this section if it is not found.
3. Add the services and corresponding named subsystems configured. For example:

   ```
   [/HTTP Services]

   SiebelCIFContact = CIFInboundHTTPDispatch
   ```

   The name SiebelCIFContact in the queryString matches the name CIFInboundHTTPDispatch, which in turn looks up for the named subsystem list and dispatches accordingly.

4. Save and close the configuration file.

About Siebel Universal Customer Master and CRM Integration

Siebel Universal Customer Master (Siebel UCM) is preconfigured to support integration with a combined instance of Siebel CRM Business Applications for Industry Applications (SIA version 7.8.2 only) and Siebel UCM. When enabled, the integration allows Siebel UCM to apply history, cross-reference, and data quality processes to Siebel CRM contact and account data.

The integration of Siebel UCM and CRM uses the Siebel UCM CRMML interface to deliver and receive data messages, and includes the following preconfigured workflows to facilitate the integration:

- CRM-UCM-Account-WF
- CRM-UCM-Contact-WF
- UCM-CRM-Account-WF
- UCM-CRM-Contact-WF

For information on extending the Siebel UCM and CRM integration, see "Extending Siebel Universal Customer Master and CRM Integration" on page 61.

To enable the preconfigured Siebel UCM and CRM integration, make sure the system preference Enable UCM Processes is set to TRUE. For further information on UCM System Preferences, see "About System Preferences for Siebel Universal Customer Master" on page 91.

Extending Siebel Universal Customer Master and CRM Integration

Extending the preconfigured Siebel UCM and CRM integration requires the following high-level configurations:

- Update the appropriate Integration Object
- Generate a new XSD
- Update the Transformation XSLT
- Update the Validation XSLT if required

For further information on these tasks, see *XML Reference: Siebel Enterprise Application Integration*.

Configuring Siebel Master Data Integration Objects

This topic contains a list of general integration configurations you can perform after installing and initially configuring Siebel Master Data Applications.

- "About Siebel Master Data Application Integration Objects" on page 62
- "Modifying Siebel Master Data Application Integration Objects" on page 63

- "Modifying UCM Envelope Integration Objects" on page 64
- "Extending UCM Integration Objects" on page 64
- "Modifying the Map Property of Application Integration Objects" on page 65
- "Adding Status Key Property to UCM Integration Object" on page 66

See the following related topics for further information on other areas of application configuration and the installation of Siebel MDA.

Related Topics

"Configuring Siebel Master Data Application Features" on page 40

About Siebel Master Data Application Integration Objects

There are three types of Siebel Master Data Applications integration objects:

- "UCM Integration Objects"
- "UCM Envelope Integration Objects"
- "UCM Dispatcher Map Integration Objects"

For background information on integration objects, see *Integration Platform Technologies: Siebel Enterprise Application Integration*.

UCM Integration Objects

The UCM integration objects contain the data hierarchy that is a subset of the data in a Siebel application business object. The UCM Integration objects map to the different UCM business objects in the Siebel Business Application. Many preconfigured integration objects are provided for you, which you can extend and modify as necessary.

These integration objects can be found in the Siebel repository, and their names start with CIF. In addition, you can create your own integration objects for use using Integration Object Builder in Siebel Tools; see *Integration Platform Technologies: Siebel Enterprise Application Integration* for further information.

This internal integration object is required so that the UCM business services, such as the UCM Transaction Manager, can receive and package the data from XML (as a CRMML message) and into a format understood by the Siebel application object manager (as business component data).

UCM Envelope Integration Objects

The envelope integration object stores system information about where the data originated and where the destination of the data. It stores information about the message (such as its ID) rather than the message contents.

The envelope integration object is required for all integration business processes. If your integration process requires a slightly different envelope integration object, customize it for your needs. Otherwise, use the envelope integration object named UCM Envelope.

For information on customizing the envelope integration object, see "Modifying UCM Envelope Integration Objects" on page 64.

UCM Dispatcher Map Integration Objects

The UCM dispatcher map is an integration object that contains the rule sets used by the UCM Dispatcher. The default UCM dispatcher map is CIFDispMap. Siebel Master Data Applications use the following predefined dispatcher maps:

- CIFDispMap—Default dispatcher map used
- CIFDispMapAccount—Dispatcher map containing only CIFAccount messages
- CIFDispMapContact—Dispatcher map containing only CIFContact messages
- CIFDispMapFINCORPAccount—Dispatcher map containing only CIFFINCORPAccount messages
- CIFDispMapHHMisc—Dispatcher map containing the CIFHousehold, CIFFINCORP Account, and CIFPersonal Address messages
- CIFDispMapPerf—Dispatcher map containing a subset of CIFContact messages for improving performance

For performance reasons in a production environment, deactivate business processes that you are not supporting. You can do so by deactivating the user property name corresponding to the business processes that are not applicable.

Modifying Siebel Master Data Application Integration Objects

This topic describes the task of modifying Siebel Master Data Application integration objects.

For background information on Siebel Master Data Application integration objects, see "About Siebel Master Data Application Integration Objects" on page 62.

To modify UCM Integration Objects

1. Determine whether or not you need to customize any integration objects listed in Table 23 on page 110, and use Siebel Tools to select the Integration Object tab.
2. Query for name of the integration objects that require modification.

3 Select the components to modify in your integration object and enter the proper information for each of the applicable columns.

NOTE: Only modify the UCM integration objects by activating or deactivating integration component fields and integration components. Large amounts of customization are difficult to upgrade. For a lists of shipped integration objects, see Chapter 7, "Siebel Master Data Applications Messages."

Modifying UCM Envelope Integration Objects

This topic describes the task of modifying Siebel UCM envelope integration objects.

For background information on Siebel Master Data Application integration objects, see "About Siebel Master Data Application Integration Objects" on page 62.

To modify UCM envelope integration objects

1 Determine whether or not you need a customized envelope integration object and use Siebel Tools to select the Integration Object tab.

2 Query for name=CIF Envelope.

3 Select the components to modify in your integration object and enter the proper information for each of the applicable columns.

4 After configuring the envelope integration object, compile it to the Siebel Repository File (`.srf`) file.

NOTE: If you have created a new envelope integration object, make sure to reflect the new configuration when configuring the Siebel Connector components for the Siebel Universal Customer Master Application. Only configure a new envelope integration object if additional integration component fields are required. For details, see "CRMML Message" on page 30.

Extending UCM Integration Objects

This topic describes the task of extending the integration object to include other data, such as Assets, Service Requests, and so on. Extending the UCM integration objects to include other data requires the creation of a new integration object based on either the Account, Contact, or Household business object. Only objects that are based on the Party data model are supported by UCM specific services.

For background information on Siebel Master Data Application integration objects, see "About Siebel Master Data Application Integration Objects" on page 62.

To modify UCM Integration Objects to include other data

1 From Siebel Tools, choose Object Explorer > Integration Object.

2 Create an integration object with integration components based on either the Account, Contact, or Household business object. For example, create an integration object with integration components Account and Assets based on the Account business object.

For background information on integration objects and how to create them, see *Integration Platform Technologies: Siebel Enterprise Application Integration*.

Modifying the Map Property of Application Integration Objects

This topic describes the task of modifying the map property of Siebel Master Data Application dispatcher map integration objects.

For background information on Siebel Master Data Application integration objects, see "About Siebel Master Data Application Integration Objects" on page 62.

To modify the dispatcher map user properties

1. From Siebel Tools, choose Object Explorer > Integration Object.
2. Query for the dispatcher map name. For example, CIFDispMap.
3. Navigate to the user properties of the dispatcher map to see its user properties; modify the properties as appropriate.

The name of the user property represents the rule the dispatcher tries to match. An example name follows:

```
CRMML/PartyPackage/DeleteAccount
```

The dispatcher tries to locate the user property name in the message received. If it finds the match, it then uses the information in the value column to determine the values it needs to insert. The value for the previous example name follows:

```
CRMML/PartyPackage/
DeleteAccount;a;b;CIFAccountDelete;CIFAccountDeleteRs;IXMLOperation_DELETE
```

Each value is made up of six tokens that are separated by a semi-colon (;), and each token represents a specific information as outlined in Table 16.

Table 16. Dispatcher Map Value User Property

Token	Description	Example
1st	The location to insert the remaining five tokens at runtime **NOTE:** The message needs to start with CRMML/PartyPackage because the UCM Dispatcher uses it to inspect the incoming XML data; that is, the first token is an XPath (XML Path Language) search expression.	`CRMML/PartyPackage/DeleteAccount`
2nd	System reserved token	a
3rd	System reserved token	b

Table 16. Dispatcher Map Value User Property

Token	Description	Example
4th	The request integration object	`CIFAccountDelete`
5th	The response integration object	`CIFAccountDeleteRs`
6th	The operation corresponding to <DeleteContact> business process, which is configured in the user property of the UCM Transaction Manager as key to operation	`IXMLOperation_DELETE`

The name column of the map user property must be unique and represents the business process name you want to support. You need to modify the dispatcher map entries to reflect the new business process name. The same principle applies to all the tokens.

NOTE: Compile the integration objects and the dispatcher map created by the wizard into your Siebel repository file (`.srf`). Make sure you migrate your new integration objects to the same database used by your client. You also need to copy your newly compiled `.srf` to the correct object directory of the same server used by your servers.

Adding Status Key Property to UCM Integration Object

This topic describes the task of adding the Status Key property to Universal Customer Master (UCM) integration objects.

For background information on Siebel Master Data Application integration objects, see "About Siebel Master Data Application Integration Objects" on page 62.

To add status key property to UCM integration object

1. Make sure that there is no active StatusObject user property in the UCM Transaction manager business service.

   For background information on UCM Transaction Manager, see "UCM Transaction Manager" on page 33.

2. Modify the input integration object to reflect the status key you require at each integration component level. For further information on this task, see *Integration Platform Technologies: Siebel Enterprise Application Integration*.

3. Set the Siebel Adapter user property StatusObject = True using Siebel Tools, and compile the Siebel repository file (srf).

6 Administering Siebel Master Data Applications

This chapter describes the administrative capabilities of Siebel Master Data Applications. Only users with administration privileges have access to these screens and procedures.

This chapter includes the following topics:

- "Registering Systems Connected to Siebel Master Data Applications" on page 67
- "Setting System Privileges for Siebel Master Data Applications" on page 69
- "About System Publish and Subscribe Modes" on page 72
- "Configuring System Publish and Subscribe for Siebel Master Data Applications" on page 73
- "Process of Creating Survivorship Rules" on page 77
- "Cross-Referencing Records with External Systems" on page 84
- "Monitoring Source Data of UCM Records" on page 87
- "Purging Source UCM Data" on page 87
- "Managing Pending Updates from External Systems" on page 88
- "Managing Duplicate UCM Records" on page 88
- "Unmerging Previously Merged UCM Records" on page 89
- "Running Data Management in Batch Mode" on page 90
- "About System Preferences for Siebel Universal Customer Master" on page 91
- "Configuring Siebel Data Quality Cleansing for Siebel UCM" on page 92
- "Configuring SSA Data Matching for Siebel UCM" on page 94
- "About Siebel Universal Customer Master Data Hierarchy" on page 96
- "Troubleshooting Siebel Master Data Applications" on page 96

Registering Systems Connected to Siebel Master Data Applications

Every application or system that connects to Siebel Master Data Applications must register through the System Registrations view of the Administration—Universal Customer Master screen. Once registered, the system's privileges and accessibility to the Siebel Master Data Applications is administered on other tabs within this view. For information on further defining the registered system, see:

- "Setting System Privileges for Siebel Master Data Applications" on page 69
- "Configuring System Publish and Subscribe for Siebel Master Data Applications" on page 73

To create a new system

1. From the application-level menu, choose Navigate > Site Map > Administration—Universal Customer Master screen.
2. From the link bar, select System Registrations.
3. In the System Registrations view, click New.
4. Enter the appropriate information in the available fields to define the system.

   Use Table 17 for information on each of the fields.

Table 17. System Fields

Field	Description
System ID	The System Number represents an identifier for an application that accesses Siebel Master Data Applications. This number can be an IP address or some other means to identify a system.
System Name	The System Name is the name of the application that accesses Siebel Master Data Applications.
Protocol Type	The protocol that an application uses to access Siebel Master Data Applications. Currently, this value is HTTP, MQSeries, or JMS.
Queue Manager Name	Name of the queue manager that receives and sends messages from Siebel Master Data Applications. Only applicable when protocol type selected is MQSeries.
Queue Receiver Channel	Name of the queue configured on the Queue Manager to receive requests and send responses. Only applicable when protocol type selected is MQSeries.
URL	URL destination for posting responses to messages. Only applicable when protocol type is HTTP.
Connection Factory	The name of the connection factory; that is, the connection means to the JMS provider. Only applicable when protocol type selected is JMS.
Send Queue	Specifies the JMS queue that messages get sent to. Only applicable when protocol type selected is JMS.
Description	Description of the application accessing Siebel Master Data Applications.
Comment	Comments regarding the application accessing Siebel Master Data Applications.

Figure 7 shows the creation of an example system.

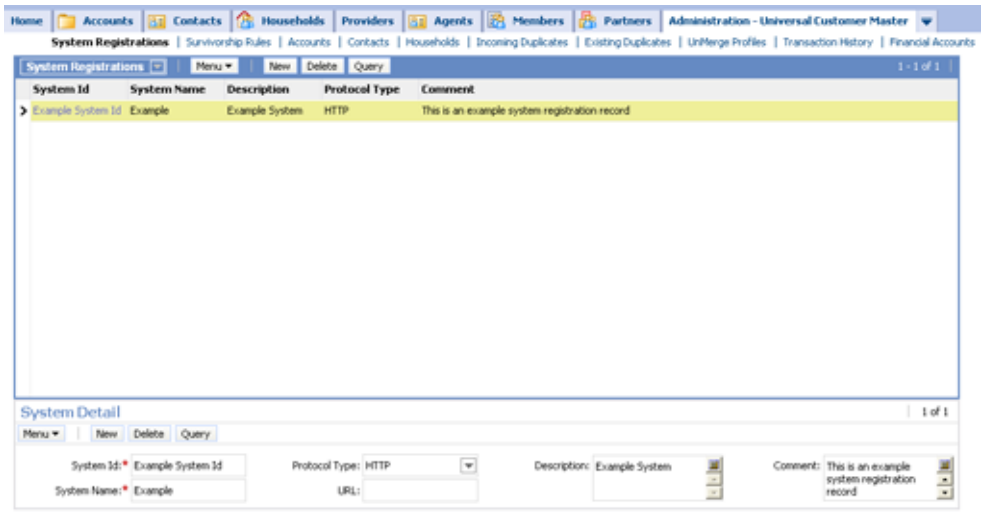

Figure 7. Creating a New System

To delete a system

1. From the application-level menu, choose Navigate > Site Map > Administration—Universal Customer Master screen.
2. From the link bar, select System Registrations.
3. In the System Registrations list, select the system of interest.

   The System Registrations form for this system, which appears under the list, provides detailed information on this system.
4. In the System Registrations form, click Delete.

   A confirmation dialog appears.

   NOTE: Deletion of registered systems can create foreign key integrity violations due to referencing of external systems in cross reference, survivorship definition, source data, history records, and so on.
5. Select OK to delete the system.

Setting System Privileges for Siebel Master Data Applications

After a system is registered, administrators can set privileges for this system, which allows grant access to Siebel Master Data Applications. For example, a system may allow insert, update, and query privileges on Contact records, but not allow the privilege of deleting the contact. Privileges are granted on the Business Objects layer.

For information on registering a system, see "Registering Systems Connected to Siebel Master Data Applications" on page 67.

To set a system's privileges

1. From the application-level menu, choose Navigate > Site Map > Administration—Universal Customer Master screen.
2. From the link bar, select System Registrations.
3. In the Systems Registrations list, select the system of interest, and drill down on the hyper link in the System Id field.
4. From the System Detail view, select the System Privileges tab.
5. In the System Privileges view, click New.
6. Enter the appropriate information in the available fields to define the privileges for each object in the system.

   Use Table 18 for information on each of the fields.

Table 18. Object Fields

Field	Description
Object Name	An object stored in Siebel Master Data Applications like Contact, Account, Household, and so on.
Query	Query privileges for the object selected in the object name field.
Insert	Insert privileges for the object selected in the object name field.
Update	Update privileges for the object selected in the object name field.
Delete	Delete privileges for the object selected in the object name field.
Comment	Comments on the privileges chosen for the selected object of the selected system.

Figure 8 shows the definition of privileges for the example system.

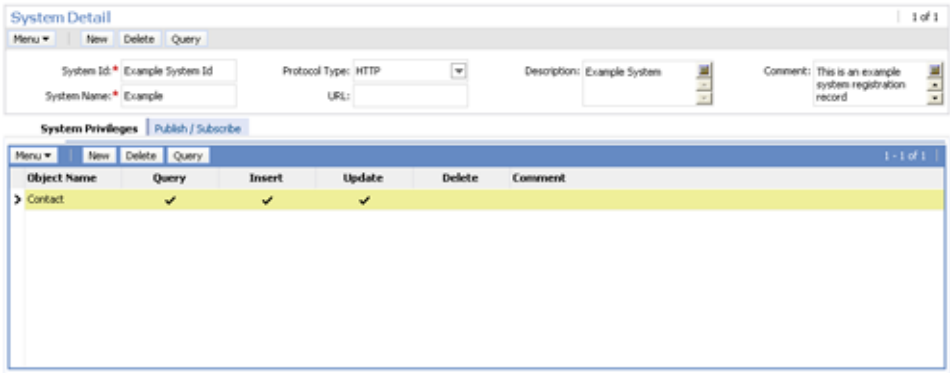

Figure 8. Defining System Privileges

The configuration in Figure 8 indicates that the system registered with the name Example has query, insert, and update privileges for the Contact business object. However, it is not allowed to delete any of the Contact records in the Siebel Universal Customer Master Application. Furthermore, it is set to subscribe to record changes for the Contact business object.

About System Publish and Subscribe Modes

Siebel Master Data Applications uses one of three modes to publish or update data: real-time mode, batch mode, and event mode. Details on how these modes function within the application are as follow.

For details on configuring these modes, see "Configuring System Publish and Subscribe for Siebel Master Data Applications" on page 73.

Real-Time Mode

When configured, the real-time mode of the publish and subscribe functionality runs when another process or business service explicitly invokes the UCM Publish/Subscribe Service business service. The calling process sends this business service a record to publish. The following workflow processes provide examples of real-time publishing: The UCM Customer Profile Integration Process, UCM Account Batch Data Management Workflow, and UCM Contact Batch Data Management Workflow.

Batch Mode

During a batch mode operation, the UCMBatchPubSub server component calls the UCM Daily Publish workflow process. This workflow process calls the UCM Publish/Subscribe Service business service, and this business service checks if it is being run on a server component. If the service is being run on a server component, it retrieves the server parameters for UCM Batch Object Type (alias UCMBatchObjectType), UCM Batch Size (alias UCMBatchSize), and UCM Sleep Time (alias UCMSleepTime). Then the business service iteratively runs through the following steps until the server component stops:

1. Creates an internal search specification, SearchSpec, for the business component based on the following format:

   `'(([Last Update - SDQ] <= 'mm/dd/yyyy hh:mm:ss') AND ([Last Update - SDQ] > 'mm/dd/yyyy hh:mm:ss'))'`

   where:

   `mm/dd/yyyy hh:mm:ss` arguments = the current time and the current time less the UCMSleepTime parameter value (default value of 60 seconds).

   `[Last Update - SDQ]` = the business component field name of the integration component field name specified by the SearchFieldName user property configured in the UCMBatchObjectType integration object. By default, the IO CIFContact has the IC property SearchFieldName = Last Update, which maps to the BC Contact's field Last Update - SDQ component field value [Last Update - SDQ]

2. The search specification is passed to the EAI Siebel Adapter, and it uses the method QueryPage with the SearchSpec against the UCMBatchObjectType to find the records that have changed or are new. The UCMBatchSize parameter value is used to determine how many records are returned in the query page's result set.

3 For each record returned in the query page's result set, it is published according to the registered UCM systems that have the object defined in the Publish/Subscribe view, that have a publish frequency set to Batch, and that are not expired.

Event Mode

Any system configured for Real-time or Daily-Batch mode can also subscribe to Event mode if that system requires the message to be published when an event occurs. Events refer to Siebel UCM operations such as Merge or Unmerge or Link and Update or Create New and so on. Though preconfigured, these events do not get published; Siebel UCM can be customized to prepare the outbound message (resulting from such operations) by using UCM Publish/Subscribe Service—EventPublishMethod.

Configuring System Publish and Subscribe for Siebel Master Data Applications

The Publish and Subscribe infrastructure provides workflow policies, workflow processes, and Siebel Universal Customer Master Application messages to publish data records that have been inserted or updated in the Siebel Universal Customer Master Application. The workflow policies track changes to records within the Siebel Universal Customer Master Application and flag them for production; Siebel workflows process the information collected by the workflow policies and guarantee appropriate publication of the changes to subscribed systems. The messages are predefined and used for the publication of records flagged by the workflow policies.

The following procedures describe:

- Configuring real-time publish and subscribe for a system
- Configuring batch-time publish and subscribe for a system

To configure real-time publish and subscribe for a system

1. From the application-level menu, choose Navigate > Site Map > Administration—Universal Customer Master screen.
2. From the link bar, select System Registrations.
3. In the System Registrations list, select the system of interest, and drill down on the hyper link in the System Id field.
4. From the System Detail view, select the Publish/Subscribe tab.
5. In the Publish/Subscribe view, select an existing record to configure or click New.

6 Enter the appropriate information in the available fields to define the publish and subscribe privileges for each object in the system.

Use the following table for information on each of the fields. Make sure to set the Publish Frequency field to real-time.

Field	Description
Object Name	An object stored in Siebel Master Data Applications like Contact, Account, Household, and so on.
Publish Frequency	How often the Siebel Master Data Applications update the system's information for the selected object. Select the real-time value for real-time publish and subscribe functionality. For the batch-time procedure, see "To configure batch-time publish and subscribe for a system" on page 75.
Last Published	The last time the Siebel Master Data Applications published record changes for the selected object to this selected system.
Event	Signifies if the object subscribes to event publishing.
Start Date	Start date of the publish and subscribe functionality.
End Date	End date of the publish and subscribe functionality.
Comment	Comments on the publish and subscribe definition for the selected object of the selected system.

7 From the application-level menu, choose Navigate > Site Map > Administration—Business Process screen.

8 From the link bar, select Workflow Processes.

9 Query for UCM Customer Profile Integration Process or a prepared customized workflow.

For further information on the UCM Customer Profile Integration Process, see "About Siebel Master Data Application Workflows" on page 35.

10 Make sure that decision point in the workflow process is set to true, which routes incoming messages to the UCM Publish/Subscribe Service. UCM Customer Profile Integration Process

For detailed information on the Siebel Business Process Designer and configuring workflow processes, see *Siebel Business Process Framework: Workflow Guide.*

Figure 9 displays the definition of real-time publish and subscribe capabilities for the example system.

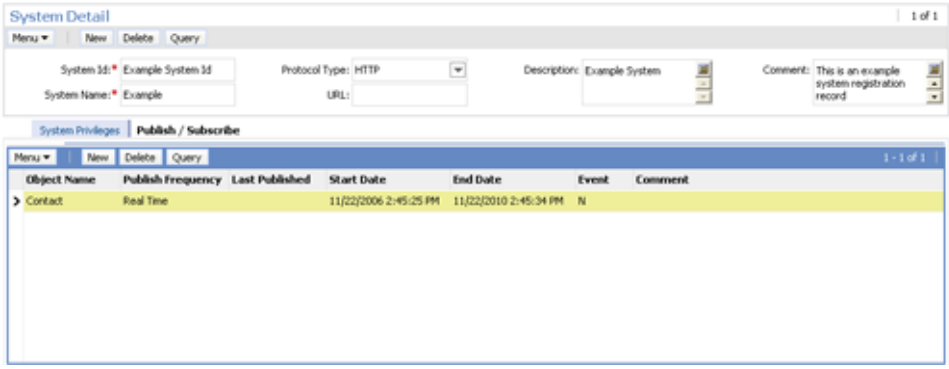

Figure 9. Defining System Real-time Publish and Subscribe Capabilities

To configure batch-time publish and subscribe for a system

1. From the application-level menu, choose Navigate > Site Map > Administration—Universal Customer Master screen.
2. From the link bar, select System Registrations.
3. In the System Registrations list, select the system of interest, and drill down on the hyper link in the System Id field.
4. From the System Detail view, select the Publish/Subscribe tab.
5. In the Publish/Subscribe view, select an existing record to configure or click New.
6. Enter the appropriate information in the available fields to define the publish and subscribe privileges for each object in the system.

   Use the following table for information on each of the fields. Make sure to set the Publish Frequency field to Daily Batch.

Field	Description
Object Name	An object stored in Siebel Master Data Applications like Contact, Account, Household, and so on.
Publish Frequency	How often the Siebel Master Data Applications update the system's information for the selected object. Select the daily batch value for batch-time publish and subscribe functionality. For the real-time publish and subscribe procedure, see "To configure real-time publish and subscribe for a system" on page 73.
Event	Signifies if the object subscribes to event publishing.
Last Published	The last time the Siebel Master Data Applications published record changes for the selected object to this selected system.

Field	Description
Start Date	Start Date is a required field to receive updates from UCM. It is the date and time which determines which records are considered for published to a particular system.
End Date	End date of the publish and subscribe functionality.
Comment	Comments on the publish and subscribe definition for the selected object of the selected system.

7 Step off the record to save the system publish and subscribe functionality.

Figure 10 displays the definition of batch-time publish and subscribe capabilities for the example system.

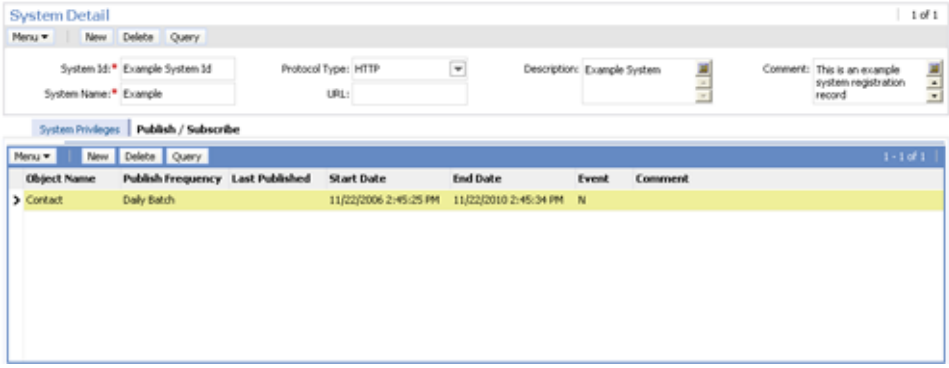

Figure 10. Defining System Batch-time Publish and Subscribe Capabilities

About UCM Survivorship Rules

Survivorship rules are an automated means of controlling the quality of customer data stored in the Universal Customer Master (UCM). Multiple systems connect to the UCM with record update and insert requests, and, to make sure that UCM presents the most trusted information contributed by each of these systems, administrators can create and use survivorship rules to govern which updates from which external systems can be trusted at the field level based on key criteria.

Survivorship rules use attribute groups, which determine the set of fields to be evaluated by the survivorship rule, and use comparison criteria, such as confidence level of the publishing system or the most recent published data, to evaluate whether inputs from given external systems can be used to update the UCM.

There is an implicit default attribute group, which includes the fields that are not explicitly defined in an attribute group definition. There is also a default criteria for each survivorship rule, that is, if there is no explicit rule definition for a certain attribute group (including the default attribute group), then the default criteria applies to determine whether an external system can update the UCM.

NOTE: Do not modify or delete the names of the default attribute groups for Account and Contact. If these names are modified, make sure they match the values in the UCM Survivorship Engine User Properties: DefaultAttrGrpName_Account or DefaultAttrGrpName_Contact.

There are three comparison methods used by survivorship rules to evaluate the quality of the object fields associated with an attribute group:

- Recent—Compares the date of the object's attribute group field's data against the date of the updating message. The most recent data survives in the record.
- History—Compares the date of the object's attribute group field's data against the date of the updating message. The oldest data survives in the record.
- Source—Compares the confidence level of the external system that contributed data to attribute group fields in the object against the confidence level of the external system of the updating message. The data of the highest confidence level survives in the record.

Note the following restrictions and recommendations when defining survivorship rules:

- Although administrators can define multiple rules for each object (account, contact), use only one active rule for each object.
- Each individual field within a given object can only be included in one attribute group definition, otherwise conflicts may occur when survivorship rules are evaluated.

For details on the process of creating survivorship rules, see "Process of Creating Survivorship Rules" on page 77.

Process of Creating Survivorship Rules

This topic lists the ordered tasks of the process for creating survivorship rules for Master Data Applications. For background information on survivorship rules, see "About UCM Survivorship Rules" on page 76.

Perform the following tasks to create survivorship rules:

1. Create Attribute Groups for the Survivorship Rule. For information on this task, see "Creating Attribute Groups for Survivorship Rules" on page 78.
2. Create the Survivorship Rule. For information on this task, see "Creating UCM Survivorship Rules" on page 78.
3. Activate and test the Survivorship Rule. For information on this task, see "Activating UCM Survivorship Rules" on page 80.

Creating Attribute Groups for Survivorship Rules

This task is a step in "Process of Creating Survivorship Rules" on page 77. For further information on survivorship rules, see "About UCM Survivorship Rules" on page 76.

To create an attribute group

1. From the application-level menu, choose Navigate > Site Map > Administration - Universal Customer Master screen.
2. From the link bar, select Survivorship Rules.
3. Select the Attribute Group Fields tab.
4. In the Attribute Group list, click the New button.
5. Enter a descriptive name for the attribute group record and select an object name.
6. In the Attribute Group Field Name list, click New.
7. Add one or more fields to define this attribute group.

Creating UCM Survivorship Rules

Creating survivorship rule records is a task in the "Process of Creating Survivorship Rules" on page 77. For further information on survivorship rules, see "About UCM Survivorship Rules" on page 76.

To create a survivorship rule

1. From the application-level menu, choose Navigate > Site Map > Administration - Universal Customer Master screen.
2. From the link bar, select Survivorship Rules.
3. In the Rule Name list, click New.

4 Enter values for the columns in the record. See the following table for details on the available columns.

Column Name	Description
Object Name	Use this column to designate what objects are affected by the survivorship rule. For example, select the contact object to create a survivorship rule based on contact information.
Default Criteria	Use this column to configure the default criteria used for attribute groups with no explicit rule definition. It is also the default comparison rule to be used when the comparison rule at the Attribute Group level cannot be used to determine the winning data. For example, when comparing by source and the confidence level of the systems are tied, the survivorship rule uses the Default Criteria as the comparison rule to compare.
Status	Sets the survivorship rule to active or inactive. Select inactive while creating and defining the survivorship rule.
Default Rule Set	When no active survivorship rule is defined or the active survivorship rule has expired, the rule with the Default Rule Set set to Y is used as the active survivorship rule. Only check Default Rule Set for one survivorship rule for each object.
Start Date	Configures the start date at which the survivorship rule becomes valid.
End Date	Configures the end date at which the survivorship rule is invalidated.

5 In the Attributes Group list click the new button to add attribute groups explicitly governed by the survivorship rule.

6 From the Attributes Group pick list, select the attribute group of focus for this survivorship rule; that is, the attributes that are evaluated by the survivorship rule to determine the most current data.

You can select preconfigured attributes groups or define unique attribute groups. For information on creating attribute groups, see "Creating Attribute Groups for Survivorship Rules" on page 78.

7 Enter values for the columns in the record. See the following table for details on the available columns.

Column Name	Description
Comparison Rule	Select either History, Recent, or Source as the rule for comparison. Based on rule selected, UCM evaluates whether the input from external system can be used to update the specified attribute group values. For more information on each comparison method, see "About UCM Survivorship Rules" on page 76.
Sequence	The order in which to evaluate multiple attribute groups.

8 Refresh the GUI view by stepping off the record.

9 If comparison rule is by source for the attribute group, set the source confidence level as follows:

   a Select the Source Confidence tab.

   b In the System ID list, click New.

   c In the new record, define the source system and set a confidence level for the individual source system.

   d Add other systems as necessary.

10 Add multiple attribute groups, as in the previous step, to further define the survivorship rule.

Activating UCM Survivorship Rules

Activating UCM survivorship rules is a task in the "Process of Creating Survivorship Rules" on page 77. For further information on survivorship rules, see "About UCM Survivorship Rules" on page 76.

NOTE: Review UCM survivorship rules thoroughly in a test environment before activating the rules.

To activate UCM Survivorship Rules

1 From the application-level menu, choose Navigate > Site Map > Administration - Universal Customer Master screen.

2 From the link bar, select Survivorship Rules

3 Select the survivorship rule of interest.

4 Select the Summary tab.

5 Review the information contained in this view for the survivorship rule.

6 Select the Attribute Groups tab.

7 Select the survivorship rule of interest and set the status field to Active.

Example of a Survivorship Rule

This topic describes the default Survivorship rules included in the Universal Customer Master (UCM), and provides an example of a default rule with modifications. You may use this feature differently, depending on your business model.

The Universal Customer Master (UCM) includes two default rules named Contact Default and Account Default. There are no explicitly defined attribute groups for either of these rules in the attribute group list in the lower view, see Figure 11. This status implies that the default attribute groups—those visible in the Attribute Groups list after selecting the Attribute Group Fields tab—are evaluated by the default criteria field in the default survivorship rule.

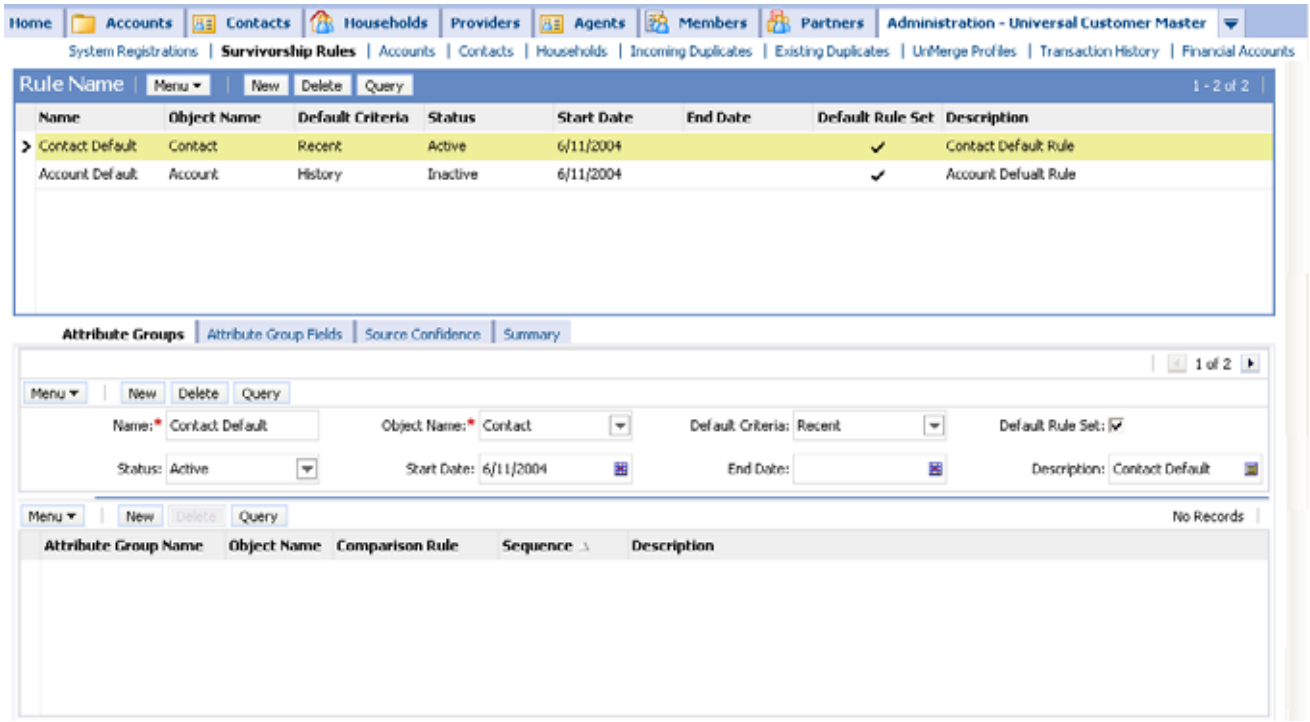

Figure 11. Default Contact and Account Survivorship Rules

The Status field for both rules is Active and the Start Date field for both rules is June 11, 2004, which indicates that this rule is ready to evaluate updates to the UCM for the default attribute groups and their associated fields.

The default criteria field for the Contacts Default rule is recent. This default criteria indicates that the Contacts Default rule evaluates the *date* of the incoming external system update against the *date* of the existing data saved in the default attribute groups. The most recent date of the two data records survives in the UCM record.

The default criteria field for the Accounts Default rule is history. This default criteria indicates that the Accounts Default rule evaluates the same incoming updates by date; however, the older data survives in the UCM record.

The following example describes creating a modification to the Contacts Default rule to enhance the survivorship rule by comparing updates to a Contact's income data. Start by creating a new attribute group:

To create a new attribute group for the Contacts Default rule

1. From the application-level menu, choose Navigate > Site Map > Administration - Universal Customer Master screen.
2. From the link bar, select Survivorship Rules, and select the Attribute Group Fields tab.
3. In the Attribute Group list, click the New button.
4. Define the new attribute group as follows:

   Attribute Group Name: Income Attribute Group
   Object: Contact
   Description: Defines rule for income field in UCM

5. In the Attribute Group Field list, click the New button:
6. Add the field name Income to the new attribute group from the drop-down menu.

The new attribute group is now defined and references the Income field. Next, attach the new attribute group to the Contacts Default rule and define the comparison method as by source:

To attach the new attribute group to the Contacts Default rule

1. From the link bar, select Survivorship Rules.
2. Set the Status field to inactive while defining the modification to the default rule.
3. In the Survivorship Rules list, select the Contact Default rule.
4. In the Attribute Group list, add a new attribute group by clicking New.
5. From the Attribute Group picklist, query for and select the attribute group created previously, Income Attribute Group.
6. Define the Comparison Rule for this attribute group as By Source.
7. Step off or save the attribute group record for the Contacts Default rule.
8. Because the comparison method is By Source, define the confidence level of the source external systems for the new attribute group:
   a. Select the Source Confidence tab.
   b. In the System ID list, click New.
   c. In the new record, define the source system. As an example, let Source A be the name of one external system and set this system's confidence level to 75.
   d. Add another source record. As an example, let Source B be the name of a second external system and set this system's confidence level to 95.

This survivorship rule evaluates all updates to the Income field based on the confidence level of the source. For example, if Source B updates the field originally, and then Source A updates at a later date, Source B's data survives because the confidence level is greater than Source A's. Alternatively, if the comparison method was Recent, Source A's data for the Income field would update the UCM record and replace Source B's data.

The final task in the process is to review and activate the enhanced rule:

To review and activate the enhanced Default Contact rule:

1. From the link bar, select Survivorship Rules
2. Select the Contact Default survivorship rule.
3. Select the Summary tab, and review the details of the modified Contact Default rule.
4. Navigate back to the survivorship rule list, and select the Contact Default rule.
5. Set the Status field to Active, and make sure the Start and End date fields do not invalidate this rule.

The modified Contact Default rule is now active and evaluates updates to the Income field.

About Managing UCM Data

Managing data stored in the Universal Customer Master (UCM) is necessary to make sure that data is accurate and up-to-date. Several screens and views are available to the administrator to monitor the history of UCM data and the source of any updates. Administrators can also manually merge duplicate records identified by UCM or roll-back updates that were previously merged. For information on the these and other management tasks, see the following sections:

- "Cross-Referencing Records with External Systems" on page 84
- "About SDH Tables" on page 85
- "Monitoring Source Data of UCM Records" on page 87
- "Purging Source UCM Data" on page 87
- "Managing Pending Updates from External Systems" on page 88
- "Managing Duplicate UCM Records" on page 88
- "Unmerging Previously Merged UCM Records" on page 89

About Cross-Referencing Records with External Systems

Cross-referencing of customer identification allows organizations to store the Customer Identification from external systems within the Siebel Universal Customer Master application. This cross-reference permits a one-to-many mapping of customer records across multiple systems throughout the organization. In other words, one record in Siebel UCM can map to one or more records in each registered application in UCM. Siebel Master Data Applications support Account, Contact, and Household party object cross-referencing.

When external systems send a request message to the Siebel Server with a request to insert a party record or a request to update a party record, the reference records are systematically entered into the Siebel UCM. The Unique Customer Identification number from external systems is embedded within the <Id> tag of the request message.

The System Administrator can manually cross-reference records from the Siebel Master Data Applications Administration Screen. For information on this task, see "Cross-Referencing Records with External Systems" on page 84.

Cross-Referencing Records with External Systems

Cross-reference Universal Customer Master (UCM) records with external systems to store source identification data in the UCM. For further information on cross-referencing records, see "About Cross-Referencing Records with External Systems" on page 84.

For further information on managing data in the UCM, see "About Managing UCM Data" on page 83.

To cross-reference a contact, account, or household record

1. From the application-level menu, choose Navigate > Site Map > Administration - Universal Customer Master screen.
2. From the link bar, select either Contacts, Accounts, or Households.
3. In the Contacts, Accounts, or Households list, select the record of interest.
4. Select the External IDs tab for Contacts, Accounts, or Households.
5. In the External Account IDs view, click New.

6 Enter the appropriate information in the available fields to define the external customer and the Unique Account Identification, for each Contact, Account, or Household party record in Siebel UCM.

Use Table 19 for information on each of the fields.

Table 19. Reference Record Fields

Field	Description
System Id	The System Id represents an identifier for an application that accesses Siebel Master Data Applications. Pick this number from the Systems entered in the System Registrations view.
System Name	The System name is the name of the application that accesses Siebel Universal Customer Master Application. This value is defaulted when a System Number is picked.
External Id1	The External Id1 stores the Unique Identifier of the external system record. This value is the only field stored in the <Id> tag in request message.
External Id2	The External Id2 stores an optional second identifier of the external system record.
External Id3	The External Id3 stores an optional third identifier of the external system record.
Comment	Comments regarding the external system record referencing the record in the Siebel Universal Customer Master Application.

About SDH Tables

The UCM Source Data History tables (SDH tables) contain the transactional contact and account data records pertaining to Siebel UCM and registered external systems. These tables hold incoming, best version, and historical data records, and provide the content for the UCM administration views. The storage of this data allows for Siebel UCM features such as survivorship, merge and unmerge, and so on.

The SDH tables are named with the S_UCM_* prefix such as S_UCM_CONTACT and S_UCM_ORG_EXT. The UCM base tables are the tables that are visible from the Siebel application such as S_PARTY, S_CONTACT, S_USER, and stores the master copy of the record.

The SDH tables ultimately contain a large amount of data and must be purged at regular intervals. For further information on this task, see "Purging Source UCM Data" on page 87.

About SDH Type Codes

Table 20 lists the Source Data History Type Codes and when they are used by UCM.

Table 20. Source Data History Type Codes

Type	Indicates
Batch	Records loaded through EIM for batch processing.
Cleansed	When results in a manual match, a cleansed record is stored temporarily for Promote or Link and Update.
Deleted	Not used
History	A version of “Best Version” prior to any modifications.
Incomplete	A record fails to process completely due to error during real-time or batch processing.
Merged	A version of the Victim record before it is merged and deleted.
Queued	Not used
Queued-Insert	Insert Request queued in SDH for batch processing.
Queued-Update	Update Request record queued in SDH for batch processing.
Queued-Upsert	Upsert Request record queued in SDH for batch processing.
Source	Request as is from an external system after it is processed in real-time.
Source-Insert	Insert Request as is from an external system after it is processed (Batch workflow changes the type from Queued-Insert).
Source-Update	Update Request as is from an external system after it is processed (Batch workflow changes the type from Queued-Update).
Source-Upsert	Upsert Request as is from an external system after it is processed (Batch workflow changes the type from Queued-Upsert).
UnMerged	A version of the Victim record that has been unmerged or brought back to life.

Monitoring History of UCM Records

Monitoring the history of UCM records reveals how the data stored in the UCM evolved and from what external system or systems the UCM record takes its content.

For information on other UCM data management tasks, see “About Managing UCM Data” on page 83.

To monitor history of UCM records

1. From the application-level menu, choose Navigate > Site Map > Administration - Universal Customer Master screen.

2 From the link bar, select either Contacts or Accounts.

3 In the Contacts or Accounts list, select the record of interest.

4 Select the Historical Version tab.

5 In this list, review the records from different source systems that make up the UCM version of the record in the top form. This list also contains historical versions of that particular UCM record within UCM.

Monitoring Source Data of UCM Records

Monitoring the source data and history of UCM records from this view reveals a global view of the evolution of UCM records.

For information on other UCM data management tasks, see "About Managing UCM Data" on page 83.

To monitor source data for Contacts or Accounts

1 From the application-level menu, choose Navigate > Site Map > Administration - Universal Customer Master screen.

2 From the link bar, select Transaction History.

3 Select either Account Source Data or Contact Source Data.

4 In this list, review the records that show the source data and history records for contacts or accounts (content is similar to the Historical Version view except that the Historical Version view only shows source and history data for a specific UCM record; this view show source data and history data for all UCM records).

Purging Source UCM Data

Maintaining historical source data for Universal Customer Master (UCM) records leads to large amounts of information stored in the database (SDH Tables). UCM provides a purge functionality to clear this data when necessary.

For information on other UCM data management tasks, see "About Managing UCM Data" on page 83.

CAUTION: The purge data operation removes all records and cannot be undone.

To purge source data

1 From the application-level menu, choose Navigate > Site Map > Administration - Universal Customer Master screen.

2 From the link bar, select Transaction History.

3 Select either Account Source Data or Contact Source Data.

4 In this list, click Purge.

Managing Pending Updates from External Systems

Duplicate Universal Customer Master (UCM) records are identified by the application and listed in the Incoming Duplicates screen. This screen displays incoming record updates from external systems that match to one existing record within UCM below the automatic update threshold or match to multiple existing records within UCM above the manual update threshold. Administrative users can evaluate the records and decide whether to update the records manually or create new records to add to the UCM.

When updating records—by selecting the Link and Update button—the UCM application:

- Saves a copy of the existing best version record into the record's History.
- Links the record from the external system to the existing best version in UCM.
- Invokes survivorship rules to produce a new best version of the record.

For information on other UCM data management tasks, see "About Managing UCM Data" on page 83.

To update or promote duplicate UCM records

1. From the application-level menu, choose Navigate > Site Map > Administration - Universal Customer Master screen.
2. From the link bar, select Incoming Duplicates.
3. Select either the Duplicate Accounts tab or Duplicate Contacts tab.
4. In the Account or Contact Source list, select the source record of interest.

   The records in this list are identified as duplicates of the base UCM record in the lower view. The record columns provide detail on each Account or Contact, including the source external system where the record originated.
5. In the lower Duplicate tab, review the UCM record that is in conflict with the record or records in the Source list.

   The record columns in this list provide details on the existing Account or Contact in the UCM. The score field determines how close the incoming record matches the existing record.
6. After review, select the source record of interest and update the two records by clicking Link and Update or promote the new record by clicking Create New.

Managing Duplicate UCM Records

The Universal Customer Master (UCM) identifies and lists duplicate Contact and Account records for review by administrators after running data quality data matching in batch mode. The administrator can resolve the duplicate records with one of the following tasks:

- Merging the duplicate record with the existing UCM record. The merging process:
  - Stores the duplicate record as type Merged and the surviving record as History in the Source Data History table

- Merges the parent records using survivorship rules to generate the new merged surviving parent record
- Updates the parent surviving record with the new merged surviving record
- Adds children records from duplicate record to the surviving record.
- Deletes the duplicate record.

- Deleting the duplicate record and leaving the existing UCM record unchanged.

To review and manage duplicate UCM records

1. From the application-level menu, choose Navigate > Site Map > Administration - Universal Customer Master screen.
2. From the link bar, select Existing Duplicates.
3. Select the Duplicate Contacts or Duplicate Accounts tab.

   Review the duplicate contacts or accounts in this view.
4. Review an individual duplicate record by drilling down on the record's hyperlink.

   The Duplicate Account Resolution or Duplicate Contact Resolution view appears. The top record is the current UCM version of the account or contact. The bottom view is the duplicate record.
5. Resolve the duplicate record by:
   a. Merging the duplicate record with the UCM record, by clicking the Merge button.
   b. Deleting the duplicate record by clicking the Menu button and then Delete Record.
   c. Deleting the UCM record by clicking the Delete button.

Unmerging Previously Merged UCM Records

A Siebel Universal Customer Master (Siebel UCM) record that results from the merging of two preexisting records, as defined by survivorship rules, can be unmerged to the state of their original records. After unmerging the merged record, Siebel UCM restores the parent victim record and reverts the parent merged record back to its previous state. Siebel UCM removes children of the victim record from the surviving merged record and adds them back to the victim record. (Siebel UCM supports a limited number of children.)

If the merged record is subsequently updated prior to rolling back, the surviving unmerged record retains the updates, while the record that did not survive during the original merge reverts to its original state.

To unmerge a merged UCM record

1. From the application-level menu, choose Navigate > Site Map > Administration - Universal Customer Master screen.

2 From the link bar, select Unmerge Profiles.

3 Select either the Merged Accounts tab or Merged Contacts tab.

4 In the Account or Contact Source list, select the source record of interest.

5 Click the Unmerge button.

Running Data Management in Batch Mode

To run Universal Customer Master (UCM) data management processes in batch mode, you create a component job based on the server component UCM Batch Manager (alias: UCMBatchProcess).

This component job fetches the IDs of the records and invokes the batch workflow. The workflow queries, cleanses, matches, and publishes the UCM records. There are three match cases that can occur: no match, one match, or multiple matches. The workflow internally calls the transaction manager and survivorship rules, depending upon the match case type.

For background information on administering component jobs, see *Siebel System Administration Guide*.

To run UCM data management in batch mode by scheduling a batch workflow

1 From the application-level menu, choose Navigate > Site Map > Administration - Server Management.

2 From the link bar, click Jobs.

3 In the Jobs list, click New.

The component job status field changes to Creating.

4 In the Component/Job field, click the drop-down list and select UCM Batch Manager.

5 In the Job Detail view, enter data in appropriate fields to describe the start time and so on of the component job; click the menu button, and then click Save Record.

6 In the Job Parameters list, add the following parameters for the component job:

Parameter Name	Value
UCM Batch Object Type	Select Contact or Account
UCM Batch Size	Select the page size of the batch job. The default value is 10.
UCM Data Management Cleanse Flag	Make sure this value is set to TRUE.
UCM Data Management Exact Match Flag	Make sure this value is set to TRUE.
UCM Data Management Match Flag	Make sure this value is set to TRUE.

Parameter Name	Value
UCM Search Specification	A Search Specification is required. Make sure a search specification is set on UCM Type Code. for example: [Contact.UCM Type Code] = 'Queued-Insert'
UCM Sort Specification	Optional Sort Specification of what order the records to be sorted for processing. Example for Contact: Created (ASCENDING), Last Name, First Name
UCM Survivorship Engine Flag	Make sure this value is set to TRUE.

7 In the Jobs list, click the Start button.

The Status field changes from Creating to Queued.

About System Preferences for Siebel Universal Customer Master

Siebel UCM system preferences control how the application operates and what functionality is enabled in your environment. You may need to review default settings and modify these as appropriate based on your business needs. For further information on setting system preferences, see *Siebel Applications Administration Guide*.

Table 21 lists and briefly describes the Siebel UCM system preferences.

Table 21. Siebel UCM System Preferences

System Preference Name	Default Value	Description
Enable CDM Cleanse 1	Account	Turn on CDM Cleanse for Account
Enable CDM Cleanse 2	Contact	Turn on CDM Cleanse for Contact
Enable CDM EM 1	Account	Turn on CDM Exact Match for Account
Enable CDM EM 2	Contact	Turn on CDM Exact Match for Contact
Enable CDM Match 1	Account	Turn on CDM Match for Account
Enable CDM Match 2	Contact	Turn on CDM Match for Contact
Enable Survivorship Engine	FALSE	Turn on Survivorship Engine
Enable UCM Processes	FALSE	Set this value to TRUE to enable integration with other applications and Siebel UCM. For further information, see "About Siebel Universal Customer Master and CRM Integration" on page 61.

Table 21. Siebel UCM System Preferences

System Preference Name	Default Value	Description
UID Generation Service	"FINS Teller Converter Extensions, ""GenerateID"", ""IDType|GUID"" "	The business service name, method, and input combination to generate unique ID value. It is used by both transaction manager and UI Interaction service.
Unique Id: Account	Party UId	Unique ID field for Account is used with transaction manager and UI Interaction
Unique Id: Contact	Party UId	Unique ID field for Contact is used with transaction manager and UI Interaction
Unique Id: FINCORP Account	Account Number	Unique ID field for FINCORP Account is used with transaction manager and UI Interaction
Unique Id: Household	Party UId	Unique ID field for Household is used with transaction manager and UI Interaction

Configuring Siebel Data Quality Cleansing for Siebel UCM

Siebel Universal Customer Master (Siebel UCM) supports data cleansing using the Siebel Data Quality module. To configure cleansing of Siebel UCM data, perform the following task.

NOTE: Siebel Data Quality module is licensed separately from Siebel UCM.

For further information on Siebel Data Quality, see *Siebel Data Quality Administration Guide*.

NOTE: The following task contains references to many system administration tasks. For further detail and background information on system administration tasks, see *Siebel System Administration Guide*.

To configure Siebel Data Quality cleansing for Siebel UCM

1. Log in to Siebel Business Applications. You must have appropriate system administration responsibilities to make the following configurations.
2. From the application-level menu, choose Navigate > Site Map > Administration - Server Configuration > Enterprises > Parameters to make sure the following enterprise parameters are set:

Parameter	Value
Data Cleansing Type	Vendor1
UCM Data Management Cleanse Flag	TRUE

3 From the application-level menu, choose Navigate > Site Map > Administration - Application screen > System Preferences view, and set the following system preferences are set. (To turn off system preferences add ".FALSE" to the value. For example: "Account.FALSE")

System Preference	Value
Enable CDM Cleanse 1:	Account
Enable CDM Cleanse 2:	Contact

For background information on setting system preferences, see *Siebel Applications Administration Guide*.

4 From the application-level menu, choose Navigate > Site Map > Administration - Server Configuration, and click the Component Definitions view tab.

5 For each of the following component definitions:

- Data Quality Manager (alias Dqmgr)
- UCM Object Manager (alias UCMObjMg)
- UCM Batch Manager (alias UCMBatchProcess)
- UCM Batch Publish Subscribe (alias UCMBatchPubSub)
- Workflow Process Batch Manager (alias WfProcBatchMgr)
- Workflow Process Manager (alias WfProcMgr)

set the following parameters:

Parameter	Value
Data Cleansing Enable Flag	TRUE
Data Cleansing Type	Vendor1

6 From the application-level menu, choose Navigate > Site Map > Administration - Server Configuration > Enterprises > Synchronize, and click the Synchronize button to synchronize the changes made to the batch-mode server components.

7 From the application-level menu, choose Navigate > Site Map > Administration - Data Quality > Data Quality Settings, and set the following parameters:

Name	Value
Disable Data Cleansing	No
Key Type	Standard
Match Threshold	90
Search Type	Typical

8 From the application-level menu, choose Navigate > Site Map > User Profile Preferences > Data Quality, and set the following parameters:

Parameter	Value
Disable Data Cleansing	No

9 Configure the UCM configuration file (ucm.cfg) by setting the following parameters in the [DataCleansing] section:

- Data Cleansing Enable = TRUE
- Data Cleansing Type = Vendor1

10 Stop and restart the Siebel Server and Siebel Gateway Name Server system services for changes to take effect.

Configuring SSA Data Matching for Siebel UCM

Siebel Universal Customer Master (Siebel UCM) support data matching using SSA-NAME3 (SSA) third-party software. To configure this functionality for Siebel UCM, perform the following task.

For further information on SSA configurations, see *Siebel Data Quality Administration Guide*.

NOTE: The following task contains references to many system administration tasks. For further detail and background information on system administration tasks, see *Siebel System Administration Guide*.

To configure SSA data matching for Siebel UCM

1 Log in to Siebel Business Applications. You must have appropriate system administration responsibilities to make the following configurations.

2 From the application-level menu, choose Navigate > Site Map > Administration - Server Configuration > Enterprises > Parameters to make sure the following enterprise parameters are set:

Parameter	Value
DeDuplication Data Type	SSA
UCM Data Management Match Flag	TRUE

3 From the application-level menu, choose Navigate > Site Map > Administration - Application screen > System Preferences view, and make sure the following system preferences are set.

System Preference	Value
Enable CDM EM 1	Account
Enable CDM EM 2	Contact
Enable CDM Match 1	Account
Enable CDM Match 2	Contact

4 To turn off system preferences add ".FALSE" to the value.
Example: Account.FALSE

For background information on setting system preferences, see *Siebel Applications Administration Guide*.

5 From the application-level menu, choose Navigate > Site Map > Administration - Server Configuration, and click the Component Definitions view tab.

6 For each of the following component definitions:

- Data Quality Manager (alias Dqmgr)
- UCM Object Manager (alias UCMObjMg)
- UCM Batch Manager (alias UCMBatchProcess)
- UCM Batch Publish Subscribe (alias UCMBatchPubSub)
- Workflow Process Batch Manager (alias WfProcBatchMgr)
- Workflow Process Manager (alias WfProcMgr)

set the following parameters:

Parameter	Value
DeDuplication Data Type	SSA
DeDuplication Enable Flag	TRUE

7 From the application-level menu, choose Navigate > Site Map > Administration - Server Configuration > Enterprises > Synchronize, and click the Synchronize button to synchronize the changes made to the batch-mode server components.

8 From the application-level menu, choose Navigate > Site Map > Administration - Data Quality > Data Quality Settings, and set the following parameters:

Name	Value
Disable Deduplication	No
Key Type	Standard

Name	Value
Match Threshold	90
Search Type	Typical

9 From the application-level menu, choose Navigate > Site Map > User Profile Preferences > Data Quality, and set the following parameters:

10 Configure the UCM configuration file (ucm.cfg) by setting the following parameter in the [DeDuplication] section:

- DeDuplication Enable = TRUE

11 Stop and restart the Siebel Server and Siebel Gateway Name Server system services for changes to take effect.

About Siebel Universal Customer Master Data Hierarchy

Siebel UCM supports the use of account data hierarchies with the master customer data, and includes screens to display and view the account hierarchies. Siebel UCM also supports copying existing hierarchies with their existing relationships.

Three new tables allow for this functionality: S_DYN_HRCHY, which stores hierarchy information as a header; S_DYN_HRCHY_REL, which stores relationships; and S_DYNHR_RPT_REL, which stores denormalized hierarchy relationships.

For further background information on data hierarchy concepts, see *Siebel Applications Administration Guide*.

Troubleshooting Siebel Master Data Applications

This topic provides guidelines for resolving problems with Siebel Master Data applications.

To resolve the problem, look for it in the list of error messages in Table 22.

Table 22. Resolving Siebel Master Data Application Error Messages

Error Number	Error Text	Explanation	Resolution
SBL-IAI-00427	%1 Checking if privilege is allowed for %2 operation	Checking if system has privilege to perform the operation.	Informational: no action required.
SBL-IAI-00428	%1 Checking if security is enforced	Checking if security is enabled in UCM.	Informational: no action required.
SBL-IAI-00429	%1 Checking system registration for system %2	Checking if system is registered in UCM.	Informational: no action required.
SBL-IAI-00430	%1 Generating Unique Id for %2	UCM is generating a new Unique ID for this message.	Informational: no action required.
SBL-IAI-00431	%1 Inserting reference record for %2 with Id %3	Inserting External ID record.	Informational: no action required.
SBL-IAI-00432	%1 Failed to find %2	Business Service failed to find an object.	Check to make sure object exists.
SBL-IAI-00433	%1 Failed to find integration object instance	Business Service failed to find an Integration Object instance.	This error indicates that no data or no valid data was received. Check that an integration object instance exists.
SBL-IAI-00434	%1 Failed to find Integration Object name	Business Service failed to find an Integration Object name.	Check that an integration object instance exists in the property set.
SBL-IAI-00435	%1 Failed to find message instance under XML Hierarchy	Not able to find a message in input XML.	Check that the XML document sent to UCM contains a message body.
SBL-IAI-00436	%1 Failed to find System Id in input message	No system number matches the system ID in input XML.	The System ID value is missing in the message. Provide a valid System ID in the input XML.
SBL-IAI-00437	%1 Failed to find XML Hierarchy	Input XML message does not contain valid XML.	Check if the input property set contains a child XML hierarchy property set.

Table 22. Resolving Siebel Master Data Application Error Messages

Error Number	Error Text	Explanation	Resolution
SBL-IAI-00438	%1 Failed to find Privilege record for %2 object in UCM Administration	Required privilege is not allowed on the object.	Check in the UCM administration views that the system is registered and has privileges defined for it.
SBL-IAI-00439	%1 Failed to find %2 in System Registration	Unable to find system in System Registration view.	Check that the incoming message has a valid System ID by verifying that the SystemId field value is not null, and that the SystemId is registered in the UCM Administration views.
SBL-IAI-00440	%1 Updating Cross Reference record for %2 with Id %3	Updating external ID for the record.	Informational: no action required.
SBL-IAI-00442	%1 Found match	Match found.	Informational: no action required.
SBL-IAI-00443	%1 Found no match	No match found.	Informational: no action required
SBL-IAI-00444	%1 Found multiple instances within the input	Multiple accounts or contacts found in the input from SDH record. Currently CDM supports only one record.	Check the input data and verify that only one instance (for example, of an account or contact) is supplied.
SBL-IAI-00445	%1 Found no instance within input	No instance found in the input data.	Verify that the input data is not null, and the data is in a valid format.
SBL-IAI-00446	%1 has no Manual Threshold definition in user property	No Manual Threshold definition found in the user property of CDM service.	Check if a Manual Threshold definition exists in UCM Data Quality Manager Business Service user properties, and make sure the value is between 0 and 100.
SBL-IAI-00447	%1 Found no data cleansing output	No data cleansing output found. Data quality cleanse did not return any valid output.	Check the data cleansing service and parameters to verify it has been configured correctly.

Table 22. Resolving Siebel Master Data Application Error Messages

Error Number	Error Text	Explanation	Resolution
SBL-IAI-00448	%1 PropertySet doesn't match the type %2	The type for the PropertySet is not right. Type=Match Info. Data quality match should return PropertySet of type Match Info.	Check the output PropertySet of your Deduplication Business Service. The PropertySet should have a Type of Match Info.
SBL-IAI-00449	%1 Search failed to return results from %2	The search failed to return active row back.	The search failed to return an active row and one active row is expected. You may receive this error in following cases: ■ When updating the Duplicate Flag in UCM table, UCM does a query for the duplicate record and does not find it. ■ When querying for the cleansed version of source data, UCM does a query for the cleansed record and does not find it. ■ When querying the business component within the Deduplication business object, UCM does a query for the field used in Deduplication, and does not find the field. ■ When cleaning up the selected data in the business component within the Deduplication business object, UCM does a query for the business component record and does not find a record. ■ When clearing the UCM source table Duplicate Flag, UCM does a query for the duplicate record and does not find it.
SBL-IAI-00451	%1 Failed to find %2 with %3	Business Service failed to find the object.	Check to make sure an object exists for the Id supplied.
SBL-IAI-00455	%1 Failed to find a valid Rule Set in UCM Administration	Business Service failed to find configured default Rule Set.	Check to make sure a default and nonexpired Rule Set for the Object is configured in UCM Administration for Survivorship.

Table 22. Resolving Siebel Master Data Application Error Messages

Error Number	Error Text	Explanation	Resolution
SBL-IAI-00457	%1 Failed to find Default Attribute Group	Business Service failed to find configured default Attribute Group.	Check to make sure a default and nonexpired Attribute Group for the object is configured in UCM Administration for Survivorship.
SBL-IAI-00458	%1 Invalid value %2 for parameter Batch Size	Business Service found invalid value for input parameter Batch Size.	Check to make sure the input argument for the Batch Size is an integer and greater than zero.
SBL-IAI-00459	%1 Invalid value %2 for parameter Sleep Time	Business Service found invalid value for input parameter Sleep Time.	Check to make sure the input argument Sleep Time is an integer and equal to or greater than zero.
SBL-IAI-00460	%1 Failed to find required parameter Object Type	Business Service failed to find required input parameter Object Type.	Check to make sure the input argument has an Object Type. The Object Type choices are Contact, Account, or Household.
SBL-IAI-00461	%1 Invalid Match Id %2 for AutoMatch	Business Service found invalid Match Id for AutoMatch.	Check to make sure there is a valid and not null Match Id value supplied for the AutoMatch function.
SBL-IAI-00462	%1 Failed to find current record for incoming record with SDH Id %2	Unable to query Best Version record from base table while preparing History record. This could be because no record is satisfying the User Key in IO (for example, like fields not present in input, or fields are empty in input).	Check to make sure the incoming record contains either a Party_UID or Row_Id, or both. If any of these Id values are supplied, check that there is one and only one matching record in the SDH tables with that value.

Table 22. Resolving Siebel Master Data Application Error Messages

Error Number	Error Text	Explanation	Resolution
SBL-IAI-00463	%1 Failed to link source record with Id %2 to base record	After the operation, when linking SDH record to the Best Version record, unable to update Best Version Id in SDH record in SDH table. Reason could be because no User Key fields are in input.	Check to make sure UCM UId is valid.
SBL-IAI-00465	%1 Found no User Property key definition for %2 in Dispatch Map	Cannot find the User Property key in the Dispatch Map Integration Object definition.	Check that there is a User Property defined in the UCM Dispatch Map Integration Object, such as CIFDispMap.
SBL-IAI-00466	%1 Found no Integration Object definition for %2	No Integration Object definition is found in the repository. Check that you have the correct Integration Object name in the message.	Make sure the integration object name is correct.
SBL-IAI-00467	%1 Found multiple selections from UI	Multiple selection from the dedup UI.	Select only one row in the applet when running Link and Update.
SBL-IAI-00474	%1: Child object %2 with Id %3 not found. Record not reparented	Reparenting of one of the child record has failed because child record Id is not found.	The selected record could not be unmerged. This may have been due to the other child record being deleted.
SBL-IAI-00475	%1 Failed to reparent child %2	Reparenting of one of the child record is failed because child record Id is not found.	Informational: no action required.
SBL-IAI-00478	Failed to find Case Type	Valid CaseType values are: NoMatch, AutoMatch, and Review.	Informational: no action required.

Table 22. Resolving Siebel Master Data Application Error Messages

Error Number	Error Text	Explanation	Resolution
SBL-IAI-00479	%1: Unmerge of parent record failed because merged SDH victim record is not found for object %2	Unmerge of Parent record failed.	Informational: no action required.
SBL-IAI-00480	%1: Unmerge of parent record failed because surviving record is not refreshed for object %2	When unmerge is done after merge operation (without any update operation), then surviving record is refreshed. If this refresh fails then this error would occur.	Informational: no action required.
SBL-IAI-00481	Parent object %1 with Id %2 not found	Unmerge failed: Parent record not found.	The selected record could not be unmerged. This may have been due to the deletion of parent record.
SBL-IAI-00482	Unmerge complete: Deletion of last unmerged children for SDH Id %1 from merge table failed for Object %2	After Unmerge process is completed, deletion of children record from merge table has failed.	The Unmerge process completed successfully but the children records were not deleted. Please review the application logs for detailed messages on why the child records were not deleted. Most likely there is a database issue you need to resolve. You need to work with your database administrator to manually delete the child records from the database. The SQL statement ia in the format of 'DELETE FROM <tbo>.S_UCM_XXX_MERGE WHERE UCM_XXX_ID=<ID value from message>'. If you are working with accounts, substitute XXX with ORG. If you are working with Contacts, substitute XXX with CON.

Table 22. Resolving Siebel Master Data Application Error Messages

Error Number	Error Text	Explanation	Resolution
SBL-IAI-00483	Unmerge complete: Updating unmerged record to Unmerge state failed for SDH Id %1 for Object %2	After Unmerge process is completed, when setting state of victim record from Merged to Unmerged, failed.	The Unmerge process completed successfully but the victim record status was not changed from Merged to Unmerged. Please review the application logs for detailed messages on why the status could not be updated. Most likely there is a database issue you need to resolve. You need to work with your database administrator to manually update the status value in the database. If you are working with Accounts, the SQL statement is "UPDATE <tbo>.S_UCM_CONTACT SET UCM_TYPE_CD='Unmerged' WHERE ROW_ID=<ID value from message>'. If you are working with contacts, the SQL statement is "UPDATE <tbo>.S_UCM_ORG_EXT SET UCM_TYPE_CD='Unmerged' WHERE ROW_ID=<ID value from message>"
SBL-IAI-00484	Message is received and queued for batch processing	When Message is received by UCM in Batch Mode, this message is queued for batch processing.	Informational: no action required.
SBL-IAI-00485	valid protocol type	Passed as parameter 2 to error message: IDS_SIA_CIF_MISSING_GENERAL.	Informational: no action required.
SBL-IAI-00486	%1 Creating Default Attribute Group Data	Creating default attribute group data for the new object instance.	Informational: no action required.
SBL-IAI-00487	%1 Getting Attribute Group Data	Retrieving attribute group data for the object.	Informational: no action required.
SBL-IAI-00488	%1 Getting survivorship rule	Retrieving active or default Survivorship Rule.	Informational: no action required.

Table 22. Resolving Siebel Master Data Application Error Messages

Error Number	Error Text	Explanation	Resolution
SBL-IAI-00489	%1 Inserting Attribute Group Data	Inserting Attribute Group data for the corresponding object.	Informational: no action required.
SBL-IAI-00490	%1 Processing base fields	Processing and intelligently merging the supported base fields in the object.	Informational: no action required.
SBL-IAI-00495	Prepare Rule Engine Input	Business Service Method Name. The first step to prepare input argument for Rule Engine.	Check if argument is missing for this method.
SBL-IAI-00496	Siebel Message	Business Service Method Argument.	Provide this argument for the business service method.
SBL-IAI-00497	Object Id	Business Service Method Argument.	Provide this argument for the business service method.
SBL-IAI-00498	Top Level Business Component Name	Business Service Method Argument.	Provide this argument for the Business Service method.
SBL-IAI-00499	Invalid input argument %1 found in the %2 method in Business Service %3	Input argument is invalid.	Check that input argument is not empty and contains valid data.
SBL-IAI-00500	Prepare History Record	Business Service Method Name. The method used to prepare History record.	Check if argument is missing for this method.

Table 22. Resolving Siebel Master Data Application Error Messages

Error Number	Error Text	Explanation	Resolution
SBL-IAI-00501	Failed to validate Account Hierarchy relationship - can't find %1 with parent %2 in the relationship	The current global Account Hierarchy has an account that does not exist in the hierarchy. For example, if you got the error message: “Failed to validate Account Hierarchy relationship - can't find B with parent D in the relationship.” Given the following relationships for a hierarchy H (Accnt, Parent Accnt):(A, B) and (B, D) The relationship (B, D) is invalid because D itself is not in the relationship. Account D may not exist. Please check the relationship between B and D before copying Hierarchy H.	Verify that the relationship between the accounts identified in the error message is valid and that both accounts exist before copying the account hierarchy.
SBL-IAI-00502	%1 service: Existing %2 Best Version Record with RowId = %3 not found for System Id %4	UCM failed to find an existing record for incoming message.	Check if the External Id value in the incoming message has a correct cross-reference Id and this Id exists in UCM. This could happen if the best version record is deleted and the cross reference record still exists in the database.
SBL-IAI-00503	%1 service: Delete operation from the System Id %2 cannot be completed because record with Row Id='%3' is not found	Delete operation from External System failed because the record does not exist in UCM.	Check to make sure at least one User Key field is present in input message and the corresponding record exists in UCM.
SBL-IAI-00504	%1 service cannot interpret the output from %2 service	Business Service cannot understand the input.	Make sure that Business Service is configured correctly and that it returns valid output argument.

Table 22. Resolving Siebel Master Data Application Error Messages

Error Number	Error Text	Explanation	Resolution
SBL-IAI-00505	Service %1 method %2 requires an input parameter %3 which is missing	Invalid input parameter.	Check to make sure that input parameter has a valid value and is not empty.
SBL-IAI-00506	%1 Failed to find MatchOutput in MatchAllOutputs PropertySet in parameter XMLHierarchy in method %2	Business Service method failed to find the MatchOutput PropertySet in the MatchAllOutputs PropertySet in XMLHierarchy PropertySet.	Check to make sure that the MatchAllOutputs child PropertySet in input argument XMLHierarchy has one or more MatchOutput child PropertySets. Check that the output from the Matching service is valid. Check your log file to see more details.
SBL-IAI-00507	%1 service: Matching Component user property is not defined	Business Service was not configured correctly for the Match method.	Check to make sure the corresponding user properties 'Matching Component' and 'Matching Field' are defined in 'UCM Data Quality Manager' Business Service and has a valid value.
SBL-IAI-00508	%1 service: Exact Match Object user property is not defined	Business Service was not configured correctly for the ExactMatch method.	Check to make sure the user properties 'ExactMatch Object' is defined in 'UCM Data Quality Manager' Business Service and has a valid value.
SBL-IAI-00509	Parameter Case Type is required for method %2 in service %1	Business Service method failed to find required input parameter called Case Type.	Check to make sure the input argument has a Case Type parameter defined. The Case Type choices are Default or NoMatch or AutoMatch or Review.
SBL-IAI-00510	%1 Invalid value %2 for input parameter Position	Business Service found invalid value for input parameter Position.	Check to make sure the input argument Position is an integer, not empty, equal to or greater than zero, and less than number of records in the input PropertySet.
SBL-IAI-00511	%1 Failed to find MatchAllOutputs in input parameter XMLHierarchy in method %2	Business Service method failed to find the MatchAllOutputs PropertySet in the input XMLHierarchy PropertySet.	Check to make sure the input argument XMLHierarchy has at least one MatchAllOutputs child PropertySet. Check that the output from the Matching service is valid. Check your log file to see more details.

Table 22. Resolving Siebel Master Data Application Error Messages

Error Number	Error Text	Explanation	Resolution
SBL-IAI-00512	%1 service failed to process the record with SDH Row Id=%2	Business Service failed to process the record.	SDH Id is the Row Id of the 'Incomplete' status record in the History version. Go to the UCM Admin screen, Transaction History View and check for records with an Incomplete Status. Find the RowId of those records and check the log file for the error message stack to determine the cause of the error.
SBL-IAI-00513	%1 service: Internal error when %2 method is invoked	Fatal error encountered when Business Service method is invoked.	Fatal error encountered. For Batch Process: Restarting UCM Publish Subscribe task may solve this issue. For Real time Process: The message may have been lost, so you need to manually reprocess it.
SBL-IAI-00514	%1 service: Failed to find the PartyPackage tag in incoming message	The XML input message received does not contain PartyPackage tag.	Check to make sure that input XML message has a PartyPackage tag defined in it.
SBL-IAI-00515	%2 is not a registered object type with %1 service	The External name of the Integration Object name given does not match with registered Object Types.	To support or add a new object type follow these steps: 1 Add a value to CIF_BUSINESS_OBJECT_TYPE LOV. 2 Make sure that External Name of Integration Object definition matches with the LOV values.
SBL-IAI-00516	%1 service: Cannot find JMS secure queue authentication parameters for queue: %2	JMS secure queue usernames and passwords can be provided in the CFG file.	Cannot find usernames or passwords for JMS secure queue connection.
SBL-IAI-00517	%1: '%2' System does not have privilege for %3 operation	System does not have privilege for the specified operation.	Navigate to System Registrations screen and define privileges for this system.
SBL-IAI-00518	%1 Failed to find required parameter Search Spec	Business Service failed to find required input parameter Search Spec.	Check to make sure the input argument has an non-empty Search Spec.

7 Siebel Master Data Applications Messages

This chapter contains the following sections:

- Siebel Master Data Applications Application Services Interfaces on page 109
- Siebel Master Data Application Messages on page 122

Siebel Master Data Applications Application Services Interfaces

Siebel Master Data Applications provide application services interfaces to interact and manage the customer information stored in the Customer Master.

The application service interfaces supported with Siebel Master Data Applications are:

- **Manage Customer**—Inbound message allowing querying, inserting, updating, and deleting of Contact, Account, and Household information, including relationships.
- **Manage Address**—Inbound message allowing querying, inserting, updating, and deleting of Addresses for Contacts, Accounts, and Households.
- **Manage Profile**—Inbound message allowing querying, inserting, updating, and deleting of Contact, Account, and Household profile information.
- **Manage Activities**—Inbound message allowing querying, inserting, updating, and deleting of Contact, Account, and Household activities.
- **Manage Products**—Inbound message allowing querying, inserting, updating, and deleting of Products associated with Contacts, Accounts, and Households.
- **Manage Cross Reference**—Request and response message allowing querying of Contact, Account, and Household cross references.
- **Lookup Customer by Product**—Inbound message allowing querying Contacts, Accounts, and Households based on a Product Identification number.
- **Lookup Relationships**—Inbound message allowing querying Relationships between Contacts, Accounts, and Households.
- **Lookup Batch Address**—Inbound message allowing querying of Contacts, Accounts, and Households based on Address criteria.
- **Publish Customer**—Outbound message used for publishing.

Table 23 lists each of the messages from Siebel Master Data Applications specifications.

Table 23. Siebel Universal Customer Master Application Messages

Integration Object Name	Business Processes/Operations
UCMAccountExternalID	LookUpAccountExternalID
UCMAccountExternalIDRs	LookUpAccountExternalIDResponse
UCMContactExternalID	LookUpContactExternalID
UCMContactExternalIDRs	LookUpContactExternalIDResponse
UCMHouseholdExternalID	LookUpHouseholdExternalID
UCMHouseholdExternalIDRs	LookUpHouseholdExternalIDResponse
CIFAccount	LookUpAccount
CIFAccountRs	LookUpAccountResponse
CIFAccountActivity	LookUpAccountActivity
CIFAccountActivityRs	LookUpAccountActivityResponse
CIFAccountActivityByNameRs	LookUpAccountActivityByNameResponse
CIFAccountActivityDelete	DeleteAccountActivity
CIFAccountActivityDeleteRs	DeleteAccountActivityResponse
CIFAccountActivityInsert	InsertAccountActivity
CIFAccountActivityInsertRs	InsertAccountActivityResponse
CIFAccountActivityUpdate	UpdateAccountActivity
CIFAccountActivityUpdateRs	UpdateAccountActivityResponse
CIFAccountAddress	LookUpAccountAddress
CIFAccountAddressRs	LookUpAccountAddressResponse
CIFAccountAddressDelete	DeleteAccountAddress
CIFAccountAddressDeleteRs	DeleteAccountAddressResponse
CIFAccountAddressInsert	InsertAccountAddress
CIFAccountAddressInsertRs	InsertAccountAddressResponse
CIFAccountAddressUpdate	UpdateAccountAddress
CIFAccountAddressUpdateRs	UpdateAccountAddressResponse
CIFAccountDelete	DeleteAccount
CIFAccountDeleteRs	DeleteAccountResponse
CIFAccountInsert	InsertAccount
CIFAccountInsertRs	InsertAccountResponse

Table 23. Siebel Universal Customer Master Application Messages

Integration Object Name	Business Processes/Operations
CIFAccountProduct	LookUpAccountProduct
CIFAccountProductRs	LookUpAccountProductResponse
CIFAccountProductDelete	DeleteAccountProduct
CIFAccountProductDeleteRs	DeleteAccountProductResponse
CIFAccountProductInsert	InsertAccountProduct
CIFAccountProductInsertRs	InsertAccountProductResponse
CIFAccountProductUpdate	UpdateAccountProduct
CIFAccountProductUpdateRs	UpdateAccountProductResponse
CIFAccountProfile	LookUpAccountProfile
CIFAccountProfileRs	LookUpAccountProfileResponse
CIFAccountProfileByName	LookUpAccountProfileByName
CIFAccountProfileByNameRs	LookUpAccountProfileByNameResponse
CIFAccountProfileDelete	DeleteAccountProfile
CIFAccountProfileDeleteRs	DeleteAccountProfileResponse
CIFAccountProfileInsert	InsertAccountProfile
CIFAccountProfileInsertRs	InsertAccountProfileResponse
CIFAccountProfileUpdate	UpdateAccountProfile
CIFAccountProfileUpdateRs	UpdateAccountProfileResponse
CIFAccountRelationship	LookUpAccountRelationship
CIFAccountRelationshipRs	LookUpAccountRelationshipResponse
CIFAccountRelationshipDelete	DeleteAccountRelationship
CIFAccountRelationshipDeleteRs	DeleteAccountRelationshipResponse
CIFAccountRelationshipInsert	InsertAccountRelationship
CIFAccountRelationshipInsertRs	InsertAccountRelationshipResponse
CIFAccountRelationshipUpdate	UpdateAccountRelationship
CIFAccountRelationshipUpdateRs	UpdateAccountRelationshipResponse
CIFAccountUpdate	UpdateAccount
CIFAccountUpdateRs	UpdateAccountResponse
CIFContact	LookUpContact
CIFContactRs	LookUpContactResponse
CIFContactActivity	LookUpContactActivity

Table 23. Siebel Universal Customer Master Application Messages

Integration Object Name	Business Processes/Operations
CIFContactActivityRs	LookUpContactActivityResponse
CIFContactActivityByName	LookUpContactActivityByName
CIFContactActivityByNameRs	LookUpContactActivityByNameResponse
CIFContactActivityDelete	DeleteContactActivity
CIFContactActivityDeleteRs	DeleteContactActivityResponse
CIFContactActivityInsert	InsertContactActivity
CIFContactActivityInsertRs	InsertContactActivityResponse
CIFContactActivityUpdate	UpdateContactActivity
CIFContactActivityUpdateRs	UpdateContactActivityResponse
CIFContactAddress	LookUpContactAddress
CIFContactAddressRs	LookUpContactAddressResponse
CIFContactAddressByName	LookUpContactAddressByName
CIFContactAddressByNameRs	LookUpContactAddressByNameResponse
CIFContactAddressDelete	DeleteContactAddress
CIFContactAddressDeleteRs	DeleteContactAddressResponse
CIFContactAddressInsert	InsertContactAddress
CIFContactAddressInsertRs	InsertContactAddressResponse
CIFContactAddressUpdate	UpdateContactAddress
CIFContactAddressUpdateRs	UpdateContactAddressResponse
CIFContactByName	LookUpContactByName
CIFContactByNameRs	LookUpContactByNameResponse
CIFContactDelete	DeleteContact
CIFContactDeleteRs	DeleteContactResponse
CIFContactExternalId	LookUpContactExternalId
CIFContactExternalIdRs	LookUpContactExternalIdResponse
CIFContactInsert	InsertContact
CIFContactInsertRs	InsertContactResponse
CIFContactProduct	LookUpContactProduct
CIFContactProductRs	LookUpContactProductResponse
CIFContactProductByName	LookUpContactProductByName
CIFContactProductByNameRs	LookUpContactProductByNameResponse

Table 23. Siebel Universal Customer Master Application Messages

Integration Object Name	Business Processes/Operations
CIFContactProductDelete	DeleteContactProduct
CIFContactProductDeleteRs	DeleteContactProductResponse
CIFContactProductInsert	InsertContactProduct
CIFContactProductInsertRs	InsertContactProductResponse
CIFContactProductUpdate	UpdateContactProduct
CIFContactProductUpdateRs	UpdateContactProductResponse
CIFContactProfile	LookUpContactProfile
CIFContactProfileRs	LookUpContactProfileResponse
CIFContactProfileByName	LookUpContactProfileByName
CIFContactProfileByNameRs	LookUpContactProfileByNameResponse
CIFContactProfileDelete	DeleteContactProfile
CIFContactProfileDeleteRs	DeleteContactProfileResponse
CIFContactProfileInsert	InsertContactProfile
CIFContactProfileInsertRs	InsertContactProfileResponse
CIFContactProfileUpdate	UpdateContactProfile
CIFContactProfileUpdateRs	UpdateContactProfileResponse
CIFContactRelationship	LookUpContactRelationship
CIFContactRelationshipRs	LookUpContactRelationshipResponse
CIFContactRelationshipByName	LookUpContactRelationshipByName
CIFContactRelationshipByNameRs	LookUpContactRelationshipByNameResponse
CIFContactRelationshipDelete	DeleteContactRelationship
CIFContactRelationshipDeleteRs	DeleteContactRelationshipResponse
CIFContactRelationshipInsert	InsertContactRelationship
CIFContactRelationshipInsertRs	InsertContactRelationshipResponse
CIFContactRelationshipUpdate	UpdateContactRelationship
CIFContactRelationshipUpdateRs	UpdateContactRelationshipResponse
CIFContactUpdate	UpdateContact
CIFContactUpdateRs	UpdateContactResponse
CIFFINCORPAccount	LookUpFINCORPAccount
CIFFINCORPAccountRs	LookUpFINCORPAccountResponse
CIFFINCORPAccountByName	LookUpFINCORPAccountByName

Table 23. Siebel Universal Customer Master Application Messages

Integration Object Name	Business Processes / Operations
CIFFINCORPAccountByNameRs	LookUpFINCORPAccountByNameResponse
CIFHousehold	LookUpHousehold
CIFHouseholdRs	LookUpHouseholdResponse
CIFHouseholdByName	LookUpHouseholdByName
CIFHouseholdByNameRs	LookUpHouseholdByNameResponse
CIFHouseholdDelete	DeleteHousehold
CIFHouseholdDeleteRs	DeleteHouseholdResponse
CIFHouseholdInsert	InsertHousehold
CIFHouseholdInsertRs	InsertHouseholdResponse
CIFHouseholdUpdate	UpdateHousehold
CIFHouseholdUpdateRs	UpdateHouseholdResponse
CIFPersonalAddress	LookUpPersonalAddress
CIFPersonalAddressRs	LookUpPersonalAddressResponse
CIFPersonalAddressByState	LookUpPersonalAddressByState
CIFPersonalAddressByStateRs	LookUpPersonalAddressByStateResponse

About CRMML Messages

CRMML messages (Customer Relationship Management Markup Language) are a Siebel XML specification for customer-centric data exchange with other enterprise applications and Siebel Universal Customer Master (Siebel UCM) applications. External applications send CRMML request messages to Siebel applications and receive CRMML messages in response.

CRMML messages contain an envelope, a header section, and a body section. The message specification focuses on the hierarchy and definitions of header elements. It allows users to extend the body section with user-defined elements as long as the object descriptors in the header section describe the usages of the elements in a body section.

The following example illustrates the layers of a CRMML message:

```
<?xml version="1.0" encoding="UTF-8>

<CRMML>

<! CRMML Header section starts>

      <CRMHeader>

            <FromSystem>

                  <SystemInfo type="ID">
```

```
                <SystemId>www.mycompany.com</SystemId>
                <SystemRole>Insurance Application</SystemRole>
                <SystemName>Client UI System</SystemName>
                </SystemInfo>
            </FromSystem>
            <ToSystem>
                <SystemInfo type="ID">
                <SystemId>123-456-789</SystemId>
                <SystemRole>Service Provider</SystemRole>
                <SystemName>Siebel UCM System</SystemName>
                </SystemInfo>
            </ToSystem>
            <MessageDescriptor>
                <MessageId>11SC3I2002053116:43:45</MessageId>
                <TransactionScope>any</TransactionScope>
                <Timestamp>01/01/2001</Timestamp>
                <TimeToLive>365</TimeToLive>
            </MessageDescriptor>
        </CRMHeader>
<!CRMML Header section ends>

        <PartyPackage>
            <InsertContactProfile>
                <Contact>
                    <Id>ext2223</Id>
                    <EmailAddress>cifemail</EmailAddress>
                    <FaxPhone>5555555555</FaxPhone>
                    <FirstName>0099INFN</FirstName>
                    <HomePhone>1111111111</HomePhone>
                    <LastName>0099INLN</LastName>
```

```
                </Contact>
            </InsertContactProfile>
        </PartyPackage>
</CRMML>
```

Envelope and Header Section. The CRMML message has one envelope element <CRMML>. <CRMML> is the root element for a message. It contains one optional header element <CRMHeader> and one or more body elements. The envelope has no attributes.

The header includes elements for sender and receiver. The following is an example for communication parties:

```
<! CRMML Header section starts>
<CRMHeader>
   <FromSystem>
      <SystemInfo type="ID">
            <SystemId>www.mycompany.com</SystemId>
            <SystemRole>Insurance Application</SystemRole>
         <SystemName>Client UI System</SystemName>
      </SystemInfo>
   </FromSystem>
   <ToSystem>
      <SystemInfo type="ID">
         <SystemId>123-456-789</SystemId>
         <SystemRole>Service Provider</SystemRole>
         <SystemName>Siebel UCM System</SystemName>
      </SystemInfo>
   </ToSystem>
   …
<CRMHeader>
```

<FromSystem> is the sender and <ToSystem> is the receiver in this example. Both elements contain <SystemInfo> aggregate, which holds system ID, system role, and the name of the system. In the example, Sender's name is Client UI System and receiver is Siebel Master Data Applications System. Sender plays an insurance application role and receiver plays a service provider's role.

The header also contains a message descriptor that depicts the body of the message and objects inside the body sections. An example of the message descriptor follows:

```
<CRMML>

<! CRMML Header section starts>

      <CRMHeader>

            <MessageDescriptor>

                  <MessageId>11SC3I2002053116:43:45</MessageId>

                  <TransactionScope>any</TransactionScope>

                  <Timestamp>Sunday</Timestamp>

                  <TimeToLive>OneYear</TimeToLive>

            </MessageDescriptor>

      </CRMHeader>
```

The <MessageId> element can be unique to identify a single message, a pair of request or response messages, or a group of communication messages. Transaction scope can be any or all. <Timestamp> is the creation time of a message and <TimeToLive> is the expiration time of a message. The receiver disregards a message if the message has expired.

The <MessageDescriptor> can have zero or more <ObjectDescriptor> child elements. The following object descriptor example illustrates a metadata for remote procedure call in a message header:

```
<CRMHeader>

      <MessageDescriptor>

      …

      <ObjectDescriptor>

            <ObjectId type="HREF" sytem="this">p1</ObjectId>

            <ObjectUsage>SoftwarePackage</ObjectUsage>

      </ObjectDescriptor>

      <ObjectDescriptor>

            <ObjectId type="HREF" sytem="this">m1</ObjectId>

            <ObjectUsage>Method</ObjectUsage>

      </ObjectDescriptor>

      <ObjectDescriptor>

            <ObjectId type="HREF" sytem="this">a1</ObjectId>

            <ObjectUsage>Argument</ObjectUsage>
```

```
      </ObjectDescriptor>

      </MessageDescriptor>

</CRMHeader>

<PartyPackage ID="p1">

      <InsertContactProfile ID="m1">

            <Contact ID="a1">

                  <Id>ext2223</Id>
```

The first <ObjectDescriptor> refers to the <PartyPackage> element in the message body by HREF. It indicates that <PartyPackage> is a software package. In this example, it is the Siebel Universal Customer Master Application. The second descriptor indicates that <InsertContactProfile> is a method in the package that the message intends to invoke. The third descriptor shows that <Contact> aggregate is the input argument for <InsertContactProfile>. For UCM implementation, the default usages of the first three elements are package, method, and argument; therefore, no objective descriptor is required for Universal Customer Master Application specific messages.

Certain values in the header are automatically generated by the connector, but you can specify others in the envelope integration object. Table 24 describes the elements and how they are configured. For system generated attributes, there is no configuration available.

NOTE: The CRMML envelope object information can be configured, contain custom fields, and works for both input and output formats. However, UCM business services refer to information in this object and look for specific XML values. Make sure specific fields exist in the business services before configuring the CRMML for your needs.

Table 24. Header Elements

Element	Structure	Attribute	Description	Default	Configuration
CRMML	(top level)				
CRMHeader	In CRMML		Header aggregate		
FromSystem	In CRMHeader		FromSystem aggregate		

Table 24. Header Elements

Element	Structure	Attribute	Description	Default	Configuration
SystemInfo	In FromSystem		SystemInfo aggregate		
		Type	Describes the type of SystemID such as ID or URL	ID	UCM Envelope Integration Object->SystemInfo Integration Component->type Integration Component Field, XML Literal Value
		SystemId	Identifier of the system	123-456-789	UCM Envelope Integration Object->SystemInfo Integration Component->SystemId Integration Component Field, XML Literal Value
		SystemName	Name of the system	SiebelCIF System	UCM Envelope Integration Object->SystemInfo Integration Component->SystemName Integration Component Field, XML Literal Value
		SystemRole	Role the system has	Service Provider	UCM Envelope Integration Object->SystemInfo Integration Component->SystemRole Integration Component Field, XML Literal Value

Table 24. Header Elements

Element	Structure	Attribute	Description	Default	Configuration
ToSystem	In CRMHeader		ToSystem aggregate		
SystemInfo	In ToSystem		SystemInfo aggregate		
		Type	Describes the type of SystemID such as ID or URL.	ID	UCM Envelope Integration Object->SystemInfo Integration Component->ToSystem_type Integration Component Field, XML Literal Value
		SystemId	Identifier of the system		System Generated
		SystemName	Name of the system		System Generated
		SystemRole	Role the system has		System Generated

Table 24. Header Elements

Element	Structure	Attribute	Description	Default	Configuration
Message Descriptor	In CRMHeader	MessageId	Identifier of the message		System Generated
		Transaction Scope	Currently, the only value supported for transaction scope is any		UCM Envelope Integration Object->MessageDescriptor Integration Component->MessageDescriptor_Trasaction Scope Integration Component Field, XML Literal Value
		Timestamp	Time stamp of when the message is constructed		System Generated
		TimeToLive	Expiration length of the message from Timestamp value		UCM Envelope Integration Object->MessageDescriptor Integration Component->MessageDescriptor_ToLive Integration Component Field, XML Literal Value

Body Section. The body section contains user-defined message elements. The following is an example body section and is one of the messages supported by the Siebel Master Data Applications.

```
<CRMML>

    <CRMHeader>

    …

    </CRMHeader>


    <PartyPackage ID="p1">
```

```
<InsertContactProfile ID="m1">
      <Contact ID="a1">
            <Id>ext2223</Id>
            …
```

The example indicates that an external application requests to insert a contact profile to the Siebel Master Data Applications database. The contact detail information is contained within the <Contact> aggregate.

Siebel Master Data Application Messages

The Siebel Universal Customer Master Application messages provide a solution for enterprise customer, partner, and prospect data management and access. The Siebel Universal Customer Master Application messages, embedded in Customer Relationship Management Markup Language (CRMML) XML messages, interact with systems across the enterprise application network and function as the system of record for customer data.

Contact Profile

UCM Contact Profile messages are used in insert, query, update, and delete operations to maintain Contact Profile Information. LookUpContact Profile and LookUpContactProfileByName are used for query operations. UpdateContactProfile, InsertContactProfile, and DeleteContactProfile are message pairs for request and response operations. Figure 12 describes the structure of this message.

Figure 12. UCM Contact Profile Structure

Table 25 lists the Contact Profile Query Integration objects.

Table 25. Contact Profile Query Integration Objects

Integration Object	Business Component	XML Tag	User Keys
CIFContactProfile	Contact	LookUpContactProfile	UK1:PartyUId UK2: Id
CIFContactProfileRs	Contact	LookUpContactProfileResponse	UK1:PartyUId UK2: Id

Table 26 lists the Contact Profile Query by Name Integration objects.

Table 26. Contact Profile Query by Name Integration Objects

Integration Object	Business Component	XML Tag	User Keys
CIFContactProfileByName	Contact	LookUpContactProfileByName	UK1:PartyUId UK2: Id
CIFContactProfileByNameRs	Contact	LookUpContactProfileByNameResponse	UK1:PartyUId UK2: Id

Table 27 lists the Contact Profile Update Integration objects.

Table 27. Contact Profile Update Integration Objects

Integration Object	Business Component	XML Tag	User Keys
CIFContactProfileUpdate	Contact	UpdateContactProfile	UK1:PartyUId UK2: Id
CIFContactProfileUpdateRs	Contact	UpdateContactProfileResponse	UK1:PartyUId UK2: Id

Table 28 lists the Contact Profile Insert Integration objects.

Table 28. Contact Profile Insert Integration Objects

Integration Object	Business Component	XML Tag	User Keys
CIFContactProfileInsert	Contact	InsertContactProfile	UK1:PartyUId UK2: Id
CIFContactProfileUpdateRs	Contact	InsertContactProfileResponse	UK1:PartyUId UK2: Id

Table 29 lists the Contact Profile Delete Integration objects.

Table 29. Contact Profile Delete Integration Objects

Integration Object	Business Component	XML Tag	User Keys
CIFContactProfileDelete	Contact	DeleteContactProfile	UK1:PartyUId UK2: Id
CIFContactProfileDeleteRs	Contact	DeleteContactProfileResponse	UK1:PartyUId UK2: Id

Table 30 lists the Contact Profile Integration component and Integration Component Fields.

Table 30. Contact Profile Integration Component and Integration Component Fields

Name	XML Tag	Data Type	Length	Req	Note
Alias	Alias	DTYPE_TEXT	50		
Assistant	Assistant	DTYPE_TEXT	50		
Assistant Phone #	AssistantPhone	DTYPE_TEXT	40		
Best Call Time	BestCallTime	DTYPE_TEXT	30		Bound by FINCORP_CONTACT_CALL_TIME
Cellular Phone #	CellularPhone	DTYPE_TEXT	40		
Customer Value	CustomerValue	DTYPE_TEXT	30		Bound by FIN_CON_VALUE_TYPE
Date of Birth	DateofBirth	DTYPE_DATE	7		
Email Address	EmailAddress	DTYPE_TEXT	50		
Email Address 2	EmailAddress2	DTYPE_TEXT	50		
Fax Phone #	FaxPhone	DTYPE_TEXT	40		
First Name	FirstName	DTYPE_TEXT	50	✔	
Home Phone #	HomePhone	DTYPE_TEXT	40		
Id	Id	DTYPE_ID	15	✔	This tag is required for Id cross-referencing. Tag stores incoming external Id.
Integration Id	IntegrationId	DTYPE_TEXT	30		

Table 30. Contact Profile Integration Component and Integration Component Fields

Name	XML Tag	Data Type	Length	Req	Note
Job Title	JobTitle	DTYPE_TEXT	75		
Last Name	LastName	DTYPE_TEXT	50	✔	
M/F	MF	DTYPE_TEXT	30		Bound by SEX_MF
M/M	MM	DTYPE_TEXT	15		Bound by MR_MS
Marital Status	MaritalStatus	DTYPE_TEXT	30		Bound by MARITAL_STATUS
Middle Name	MiddleName	DTYPE_TEXT	50		
Mother Maiden Name	MotherMaidenName	DTYPE_TEXT	50		
Party UId	PartyUId	DTYPE_TEXT	100		
Person UId	PersonUId	DTYPE_TEXT	100		
Personal Contact	PersonalContact	DTYPE_TEXT	1		
Preferred Communications	PreferredCommunications	DTYPE_TEXT	30		Bound by OFFER_MEDIA
Privacy Code	PrivacyCode	DTYPE_TEXT	30		Bound by PRIVACY_CODE
Social Security Number	SocialSecurityNumber	DTYPE_TEXT	20		
Status	Status	DTYPE_TEXT	30		Bound by CONTACT_STATUS
Suppress All Calls	SuppressAllCalls	DTYPE_TEXT	1		
Suppress All Emails	SuppressAllEmails	DTYPE_TEXT	1		
Suppress All Faxes	SuppressAllFaxes	DTYPE_TEXT	1		
Suppress All Mailings	SuppressAllMailings	DTYPE_TEXT	1		
Suppress Share	SuppressShare	DTYPE_TEXT	1		
Time Zone Id	TimeZoneId	DTYPE_ID	15		
Time Zone Name	TimeZoneName	DTYPE_TEXT	100		Used for picking a related entity
Work Phone #	WorkPhone	DTYPE_TEXT	40		

Table 30. Contact Profile Integration Component and Integration Component Fields

Name	XML Tag	Data Type	Length	Req	Note
Work Phone Extension	WorkPhoneExtension	DTYPE_TEXT	10		
operation	operation	DTYPE_TEXT	30		System field

Contact Address

UCM Contact Address messages are used in insert, query, update, and delete operations to maintain Contact Address Information. LookUpContact Address and LookUpContactAddressByName are used for query operations. UpdateContactAddress, InsertContactAddresse, and DeleteContactAddress are message pairs for request and response operations. Figure 13 describes the structure of this message.

Figure 13. UCM Contact Address Structure

Table 31 lists the Contact Address Query Integration objects.

Table 31. Contact Address Query Integration Objects

Integration Object	Business Component	XML Tag	User Keys
CIFContactAddress	Contact	LookUpContactAddress	UK1:PartyUId UK2: Id
CIFContactAddressRs	Contact	LookUpContactAddressResponse	UK1:PartyUId UK2: Id

Table 32 lists the Contact Address Query by Name Integration objects.

Table 32. Contact Address Query by Name Integration Objects

Integration Object	Business Component	XML Tag	User Keys
CIFContactAddressByName	Contact	LookUpContactAddressByName	UK1:PartyUId UK2: Id
CIFContactAddressByNameRs	Contact	LookUpContactAddressByName Response	UK1:PartyUId UK2: Id

Table 33 lists the Contact Address Update Integration objects.

Table 33. Contact Address Update Integration Objects

Integration Object	Business Component	XML Tag	User Keys
CIFContactAddressUpdate	Contact	UpdateContactAddress	UK1:PartyUId UK2: Id
CIFContactAddressUpdateRs	Contact	UpdateContactAddressResponse	UK1:PartyUId UK2: Id

Table 34 lists the Contact Address Insert Integration objects.

Table 34. Contact Address Insert Integration Objects

Integration Object	Business Component	XML Tag	User Keys
CIFContactAddressInsert	Contact	InsertContactAddress	UK1:PartyUId UK2: Id
CIFContactAddressInsertRs	Contact	InsertContactAddressResponse	UK1:PartyUId UK2: Id

Table 35 lists the Contact Address Delete Integration objects.

Table 35. Contact Address Delete Integration Objects

Integration Object	Business Component	XML Tag	User Keys
CIFContactAddressDelete	Contact	DeleteContactAddress	UK1:PartyUId UK2: Id
CIFContactAddressDeleteRs	Contact	DeleteContactAddressResponse	UK1:PartyUId UK2: Id

Table 36 lists the Contact Address Integration Component and Integration Component Fields.

Table 36. Contact Address Integration Component and Integration Component Fields

Name	XML Tag	Data Type	Length	Req	Note
IC: Contact					
Fax Phone #	FaxPhone	DTYPE_TEXT	40		
First Name	FirstName	DTYPE_TEXT	50	✔	
Id	Id	DTYPE_ID	15	✔	This tag is required for Id cross-referencing. Tag stores incoming external Id.
Integration Id	IntegrationId	DTYPE_TEXT	30		
Job Title	JobTitle	DTYPE_TEXT	75		
Last Name	LastName	DTYPE_TEXT	50	✔	
M/M	MM	DTYPE_TEXT	15		Bound by MR_MS
Middle Name	MiddleName	DTYPE_TEXT	50		
Party UId	PartyUId	DTYPE_TEXT	100		
Person UId	PersonUId	DTYPE_TEXT	100		
Privacy Code	PrivacyCode	DTYPE_TEXT	30		
Status	Status	DTYPE_TEXT	30		Bound by CONTACT_STATUS
Work Phone #	WorkPhone	DTYPE_TEXT	40		

Table 36. Contact Address Integration Component and Integration Component Fields

Name	XML Tag	Data Type	Length	Req	Note
Work Phone Extension	WorkPhoneExtension	DTYPE_TEXT	10		
operation	operation	DTYPE_TEXT	30		
IC: Contact_INS Personal Address					
INS Personal Address Id	INSPersonalAddressId	DTYPE_ID	15		
INS Personal Address Name	INSPersonalAddressName	DTYPE_TEXT	100		
INS Personal Address Type	INSPersonalAddressType	DTYPE_TEXT	30		
INS Personal City	INSPersonalCity	DTYPE_TEXT	50	✔	
INS Personal Country	INSPersonalCountry	DTYPE_TEXT	30		
INS Personal County	INSPersonalCounty	DTYPE_TEXT	50		
INS Personal Integration Id	INSPersonalIntegrationId	DTYPE_TEXT	30		
INS Personal Postal Code	INSPersonalPostalCode	DTYPE_TEXT	30		
INS Personal State	INSPersonalState	DTYPE_TEXT	10		
INS Personal Street Address	INSPersonalStreetAddress	DTYPE_TEXT	200	✔	
INS Personal Street Address 2	INSPersonalStreetAddress2	DTYPE_TEXT	100		
IsPrimaryMVG	IsPrimaryMVG	DTYPE_TEXT	1		
operation	operation	DTYPE_TEXT	30		

Contact Activity

UCM Contact Activity messages are used in insert, query, update, and delete operations to maintain Contact Activity Information. LookUpContactActivity and LookUpContactActivityByName are used for query operations. UpdateContactActivity, InsertContactActivity, and DeleteContactActivity are message pairs for request and response operations. Figure 14 describes the structure of this message.

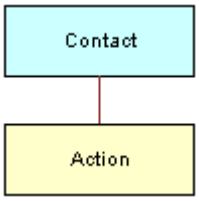

Figure 14. UCM Contact Activity Structure

Table 37 lists the Contact Activity Query Integration objects.

Table 37. Contact Activity Query Integration Objects

Integration Object	Business Component	XML Tag	User Keys
CIFContactActivity	Contact	LookUpContactActivity	UK1:PartyUId UK2: Id
CIFContactActivityRs	Contact	LookUpContactActivityResponse	UK1:PartyUId UK2: Id

Table 38 lists the Contact Activity Query by Name Integration objects.

Table 38. Contact Activity Query by Name Integration Objects

Integration Object	Business Component	XML Tag	User Keys
CIFContactActivityByName	Contact	LookUpContactActivityByName	UK1:Party UId UK2: Id
CIFContactActivityByNameRs	Contact	LookUpContactActivityByNameResponse	UK1:Party UId UK2: Id

Table 39 lists the Contact Activity Update Integration objects.

Table 39. Contact Activity Update Integration Objects

Integration Object	Business Component	XML Tag	User Keys
CIFContactActivityUpdate	Contact	UpdateContactActivity	UK1:PartyUId UK2: Id
CIFContactActivityUpdateRs	Contact	UpdateContactActivityResponse	UK1:PartyUId UK2: Id

Table 40 lists the Contact Activity Insert Integration objects.

Table 40. Contact Activity Insert Integration Objects

Integration Object	Business Component	XML Tag	User Keys
CIFContactActivityInsert	Contact	InsertContactActivity	UK1:PartyUId UK2: Id
CIFContactActivityInsertRs	Contact	InsertContactActivityResponse	UK1:PartyUId UK2: Id

Table 41 lists the Contact Activity Delete Integration objects.

Table 41. Contact Activity Delete Integration Objects

Integration Object	Business Component	XML Tag	User Keys
CIFContactActivityDelete	Contact	DeleteContactActivity	UK1:PartyUId UK2: Id
CIFContactActivityDeleteRs	Contact	DeleteContactActivityResponse	UK1:PartyUId UK2: Id

Table 42 lists the Contact Activity Integration Component and Integration Component Fields.

Table 42. Contact Activity Integration Component and Integration Component Fields

Name	XML Tag	Data Type	Length	Req	Note
IC: Contact					
Assistant	Assistant	DTYPE_TEXT	50		

Table 42. Contact Activity Integration Component and Integration Component Fields

Name	XML Tag	Data Type	Length	Req	Note
Assistant Phone #	AssistantPhone	DTYPE_TEXT	40		
Cellular Phone #	CellularPhone	DTYPE_TEXT	40		
Email Address	EmailAddress	DTYPE_TEXT	50		
Fax Phone #	FaxPhone	DTYPE_TEXT	40		
First Name	FirstName	DTYPE_TEXT	50	✔	
Home Phone #	HomePhone	DTYPE_TEXT	40		
Id	Id	DTYPE_ID	15	✔	This tag is required for Id cross-referencing. Tag stores incoming external Id.
Integration Id	IntegrationId	DTYPE_TEXT	30		
Job Title	JobTitle	DTYPE_TEXT	75		
Last Name	LastName	DTYPE_TEXT	50	✔	
M/F	MF	DTYPE_TEXT	30		Bound by SEX_MF
M/M	MM	DTYPE_TEXT	15		Bound by MR_MS
Middle Name	MiddleName	DTYPE_TEXT	50		
Party UId	PartyUId	DTYPE_TEXT	100		
Person UId	PersonUId	DTYPE_TEXT	100		
Preferred Communications	PreferredCommunications	DTYPE_TEXT	30		Bound by OFFER_MEDIA
Status	Status	DTYPE_TEXT	30		Bound by CONTACT_STATUS
Work Phone #	WorkPhone	DTYPE_TEXT	40		
Work Phone Extension	WorkPhoneExtension	DTYPE_TEXT	10		
operation	operation	DTYPE_TEXT	30		
IC: Action					
Account Id	AccountId	DTYPE_ID	15		

Table 42. Contact Activity Integration Component and Integration Component Fields

Name	XML Tag	Data Type	Length	Req	Note
Account Location	AccountLocation	DTYPE_TEXT	50		
Account Name	AccountName	DTYPE_TEXT	100		Used for picking a related entity.
Audience	Audience	DTYPE_TEXT	30		Bound by ACTIVITY_AUDIENCE_TYPE
Category	Category	DTYPE_TEXT	30		Bound by TODO_TYPE
Description	Description	DTYPE_TEXT	100		
Done	Done	DTYPE_DATETIME	7		
Done Flag	DoneFlag	DTYPE_TEXT	1		
Due	Due	DTYPE_DATETIME	7		
Duration Minutes	DurationMinutes	DTYPE_NUMBER	22		Bound by APPOINTMENT_DURATION
Id	Id	DTYPE_ID	15		
Integration Id	IntegrationId	DTYPE_TEXT	30		
MeetingLocation	MeetingLocation	DTYPE_TEXT	100		
No Sooner Than Date	NoSoonerThanDate	DTYPE_UTCDATETIME	7		
Planned	Planned	DTYPE_UTCDATETIME	7		
Planned Completion	PlannedCompletion	DTYPE_UTCDATETIME	7		
Priority	Priority	DTYPE_TEXT	30		
Started	Started	DTYPE_UTCDATETIME	7		
Status	Status	DTYPE_TEXT	30		Bound by EVENT_STATUS
Type	Type	DTYPE_TEXT	30	✔	Bound by TODO_TYPE
operation	operation	DTYPE_TEXT	30		

Contact Product

UCM Contact Products messages are used in insert, query, update, and delete operations to maintain Contact Product (Financial Account) Information. LookUpContactProduct and LookUpContactProductByName are used for query operations. UpdateContactProduct, InsertContactProduct, and DeleteContactProduct are message pairs for request and response operations. Figure 15 describes the structure of this message

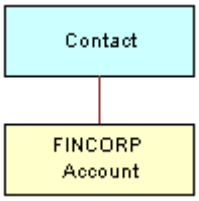

Figure 15. UCM Contact Product Structure

Table 43 lists the Contact Product Query Integration objects.

Table 43. Contact Product Query Integration Objects

Integration Object	Business Component	XML Tag	User Keys
CIFContactProduct	Contact	LookUpContactProduct	UK1:PartyUId UK2: Id
CIFContactProductRs	Contact	LookUpContactProductResponse	UK1:PartyUId UK2: Id

Table 44 lists the Contact Product Query by Name Integration objects.

Table 44. Contact Product Query by Name Integration Objects

Integration Object	Business Component	XML Tag	User Keys
CIFContactProductByName	Contact	LookUpContactProductByName	UK1:PartyUI d UK2: Id
CIFContactProductByNameR	Contact	LookUpContactProductByNameResponse	UK1:PartyUI d UK2: Id

Table 45 lists the Contact Product Update Integration objects.

Table 45. Contact Product Update Integration Objects

Integration Object	Business Component	XML Tag	User Keys
CIFContactProductUpdate	Contact	UpdateContactProduct	UK1:PartyUId UK2: Id
CIFContactProductUpdateRs	Contact	UpdateContactProductResponse	UK1:PartyUId UK2: Id

Table 46 lists the Contact Product Insert Integration objects.

Table 46. Contact Product Insert Integration Objects

Integration Object	Business Component	XML Tag	User Keys
CIFContactProductInsert	Contact	InsertContactProduct	UK1:PartyUId UK2: Id
CIFContactProductInsertRs	Contact	InsertContactProductResponse	UK1:PartyUId UK2: Id

Table 47 lists the Contact Product Delete Integration objects.

Table 47. Contact Product Delete Integration Objects

Integration Object	Business Component	XML Tag	User Keys
CIFContactProductDelete	Contact	DeleteContactProduct	UK1:PartyUId UK2: Id
CIFContactProductDeleteRs	Contact	DeleteContactProductResponse	UK1:PartyUId UK2: Id

Table 48 lists the Contact Product Integration Component and Integration Component Fields.

Table 48. Contact Product Integration Component and Integration Component Fields

Name	XML Tag	Data Type	Length	Req	Note
IC: Contact					
Assistant	Assistant	DTYPE_TEXT	50		

Table 48. Contact Product Integration Component and Integration Component Fields

Name	XML Tag	Data Type	Length	Req	Note
Assistant Phone #	AssistantPhone	DTYPE_TEXT	40		
Cellular Phone #	CellularPhone	DTYPE_TEXT	40		
Email Address	EmailAddress	DTYPE_TEXT	50		
Fax Phone #	FaxPhone	DTYPE_TEXT	40		
First Name	FirstName	DTYPE_TEXT	50	✔	
Home Phone #	HomePhone	DTYPE_TEXT	40		
Id	Id	DTYPE_ID	15	✔	This tag is required for Id cross-referencing. Tag stores incoming external Id.
Integration Id	IntegrationId	DTYPE_TEXT	30		
Job Title	JobTitle	DTYPE_TEXT	75		
Last Name	LastName	DTYPE_TEXT	50	✔	
M/F	MF	DTYPE_TEXT	255		Bound by SEX_MF
M/M	MM	DTYPE_TEXT	15		Bound by MR_MS
Middle Name	MiddleName	DTYPE_TEXT	50		
Party UId	PartyUId	DTYPE_TEXT	100		
Person UId	PersonUId	DTYPE_TEXT	100		
Preferred Communications	PreferredCommunications	DTYPE_TEXT	30		Bound by OFFER_MEDIA
Status	Status	DTYPE_TEXT	30		
Work Phone #	WorkPhone	DTYPE_TEXT	40		
Work Phone Extension	WorkPhoneExtension	DTYPE_TEXT	10		
operation	operation	DTYPE_TEXT	30		
IC: FINCORP Account					
Account Branch ABA	AccountBranchABA	DTYPE_TEXT	50		Used for picking a related entity.

Table 48. Contact Product Integration Component and Integration Component Fields

Name	XML Tag	Data Type	Length	Req	Note
Account Branch Id	AccountBranchId	DTYPE_ID	15		
Account Holder	AccountHolder	DTYPE_TEXT	100		Used for picking a related entity.
Account Name	AccountName	DTYPE_TEXT	100		
Account Number	AccountNumber	DTYPE_TEXT	100	✔	
Account Status	AccountStatus	DTYPE_TEXT	30		Bound by FINS_CRPST_LFLN_IPFS_PLC_MLOV
Account Type	AccountType	DTYPE_TEXT	30		Bound by FINCORP_PROD_ADMIN_CLASS_MLOV
Acct Gen - Branch Id	AcctGen-BranchId	DTYPE_ID	15		
Available Balance	AvailableBalance	DTYPE_CURRENCY	22		
Branch	Branch	DTYPE_TEXT	50		
Current Balance	CurrentBalance	DTYPE_CURRENCY	22		
Id	Id	DTYPE_ID	15		
Integration Id	IntegrationId	DTYPE_TEXT	30		
Name	Name	DTYPE_TEXT	100	✔	Used for picking a related entity
Organization Id	OrganizationId	DTYPE_ID	15		
Product Id	ProductId	DTYPE_ID	15	✔	
Relationship Limit	RelationshipLimit	DTYPE_CURRENCY	22		
Start Date	StartDate	DTYPE_DATE	7		
operation	operation	DTYPE_TEXT	30		

Contact Relationship

UCM Contact Relationship messages are used in insert, query, update, and delete operations to maintain Contact Relationships. LookUpContactRelationship and LookUpContactRelationshipByName are used for query operations. UpdateContactRelationship, InsertContactRelationship and DeleteContactRelationship are message pairs for request and response operations. Figure 16 describes the structure of this message.

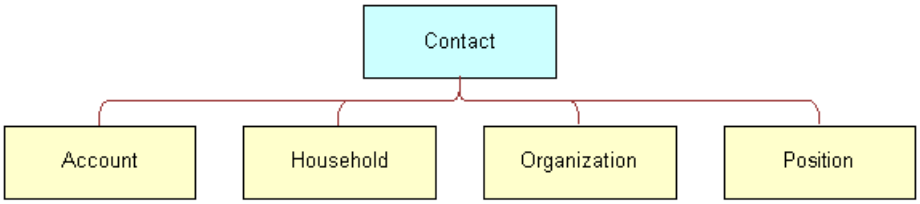

Figure 16. UCM Contact Relationship Structure

Table 49 lists the Contact Relationship Query Integration objects.

Table 49. Contact Relationship Query Integration Objects

Integration Object	Business Component	XML Tag	User Keys
CIFContactRelationship	Contact	LookUpContactRelationship	UK1:PartyUId UK2: Id
CIFContactRelationshipRs	Contact	LookUpContactRelationshipResponse	UK1:PartyUId UK2: Id

Table 50 lists the Contact Relationship Query by Name Integration objects.

Table 50. Contact Relationship Query by Name Integration Objects

Integration Object	Business Component	XML Tag	User Keys
CIFContactRelationshipByName	Contact	LookUpContactRelationshipBy Name	UK1:PartyUId UK2: Id
CIFContactRelationshipByNameRs	Contact	LookUpContactRelationshipBy NameResponse	UK1:PartyUId UK2: Id

Table 51 lists the Contact Relationship Update Integration objects.

Table 51. Contact Relationship Update Integration Objects

Integration Object	Business Component	XML Tag	User Keys
CIFContactRelationshipUpdate	Contact	UpdateContactRelationship	UK1:PartyUId UK2: Id
CIFContactRelationshipUpdateRs	Contact	UpdateContactRelationshipResponse	UK1:PartyUId UK2: Id

Table 52 lists the Contact Relationship Insert Integration objects.

Table 52. Contact Relationship Insert Integration Objects

Integration Object	Business Component	XML Tag	User Keys
CIFContactRelationshipInsert	Contact	InsertContactRelationship	UK1:PartyUId UK2: Id
CIFContactRelationshipInsertRs	Contact	InsertContactRelationshipResponse	UK1:PartyUId UK2: Id

Table 53 lists the Contact Relationship Delete Integration objects.

Table 53. Contact Relationship Delete Integration Objects

Integration Object	Business Component	XML Tag	User Keys
CIFContactRelationshipDelete	Contact	DeleteContactRelationship	UK1:PartyUId UK2: Id
CIFContactRelationshipDeleteRs	Contact	DeleteContactRelationshipResponse	UK1:PartyUId UK2: Id

Table 54 lists the Contact Relationship Integration Component and Integration Component Fields.

Table 54. Contact Relationship Integration Component and Integration Component Fields

Name	XML Tag	Data Type	Length	Req	Note
IC: Contact					
Assistant	Assistant	DTYPE_TEXT	50		
Assistant Phone #	AssistantPhone	DTYPE_TEXT	40		
Cellular Phone #	CellularPhone	DTYPE_TEXT	40		
Email Address	EmailAddress	DTYPE_TEXT	50		
Fax Phone #	FaxPhone	DTYPE_TEXT	40		
First Name	FirstName	DTYPE_TEXT	50	✔	
Home Phone #	HomePhone	DTYPE_TEXT	40		
Id	Id	DTYPE_ID	15	✔	This tag is required for Id cross-referencing. Tag stores incoming external Id.
Integration Id	IntegrationId	DTYPE_TEXT	30		
Job Title	JobTitle	DTYPE_TEXT	75		
Last Name	LastName	DTYPE_TEXT	50	✔	
M/F	MF	DTYPE_TEXT	30		Bound by SEX_MF
M/M	MM	DTYPE_TEXT	15		Bound by MR_MS
Manager First Name	ManagerFirstName	DTYPE_TEXT	50		
Manager Id	ManagerId	DTYPE_ID			
Manager Integration Id	ManagerIntegrationId	DTYPE_TEXT	30		
Manager Last Name	ManagerLastName	DTYPE_TEXT	50		Used for picking a related entity.
Manager Middle Name	ManagerMiddleName	DTYPE_TEXT	50		

Table 54. Contact Relationship Integration Component and Integration Component Fields

Name	XML Tag	Data Type	Length	Req	Note
Manager Primary Organization	ManagerPrimaryOrganization	DTYPE_TEXT	100		
Manager Primary Organization Id	ManagerPrimaryOrganizationId	DTYPE_ID	15		
Manager UId	ManagerUId	DTYPE_TEXT	100		
Middle Name	MiddleName	DTYPE_TEXT	50		
Party UId	PartyUId	DTYPE_TEXT	100		
Person UId	PersonUId	DTYPE_TEXT	100		
Preferred Communications	PreferredCommunications	DTYPE_TEXT	30		Bound by OFFER_MEDIA
Status	Status	DTYPE_TEXT	30		Bound by CONTACT_STATUS
Work Phone #	WorkPhone	DTYPE_TEXT	40		
Work Phone Extension	WorkPhoneExtension	DTYPE_TEXT	10		
operation	operation	DTYPE_TEXT	30		System field
IC: Account					
Account	Account	DTYPE_TEXT	100	✔	
Account Location	AccountLocation	DTYPE_TEXT	50		
Account Organization	AccountOrganization	DTYPE_TEXT	100		
Account Party UId	AccountPartyUId	DTYPE_TEXT	100		
Account Row Id	AccountRowId	DTYPE_ID	15		
Account Status	AccountStatus	DTYPE_TEXT	30		Bound by ACCOUNT_STATUS
IsPrimaryMVG	IsPrimaryMVG	DTYPE_TEXT	1		
Related Account Integration Id	RelatedAccountIntegrationId	DTYPE_TEXT	30		
operation	operation	DTYPE_TEXT	30		System field

Table 54. Contact Relationship Integration Component and Integration Component Fields

Name	XML Tag	Data Type	Length	Req	Note
Household					
Household Id	HouseholdId	DTYPE_ID	30		
Household Integration Id	HouseholdIntegrationId	DTYPE_TEXT	30		
Household Name	HouseholdName	DTYPE_TEXT	50		
Household Number	HouseholdNumber	DTYPE_TEXT	100		
Household Party UId	HouseholdPartyUId	DTYPE_TEXT	100		
Household Status	HouseholdStatus	DTYPE_TEXT	30		Bound by HOUSEHOLD_STATUS
IsPrimaryMVG	IsPrimaryMVG	DTYPE_TEXT	1		
operation	operation	DTYPE_TEXT	30		System field
Organization					
IsPrimaryMVG	IsPrimaryMVG	DTYPE_TEXT	1		
Organization	Organization	DTYPE_TEXT	50		
Organization Id	OrganizationId	DTYPE_ID	30		
Organization Integration Id	OrganizationIntegrationId	DTYPE_TEXT	30		
operation	operation	DTYPE_TEXT	30		System field
IC: Position					
Division	Division	DTYPE_TEXT	255		
IsPrimaryMVG	IsPrimaryMVG	DTYPE_TEXT	1		
Position	Position	DTYPE_TEXT	50		
Position Id	PositionId	DTYPE_ID			
Position Integration Id	PositionIntegrationId	DTYPE_TEXT	30		
Sales Rep	SalesRep	DTYPE_TEXT	50		
operation	operation	DTYPE_TEXT	30		System field

Contact

UCM Contact messages are used in insert, query, update, and delete operations to maintain Contact information. LookUpContact and LookUpContactByName are used for query operations. UpdateContact, InsertContact, and DeleteContact are message pairs for request and response operations. Figure 17 describes the structure of this message.

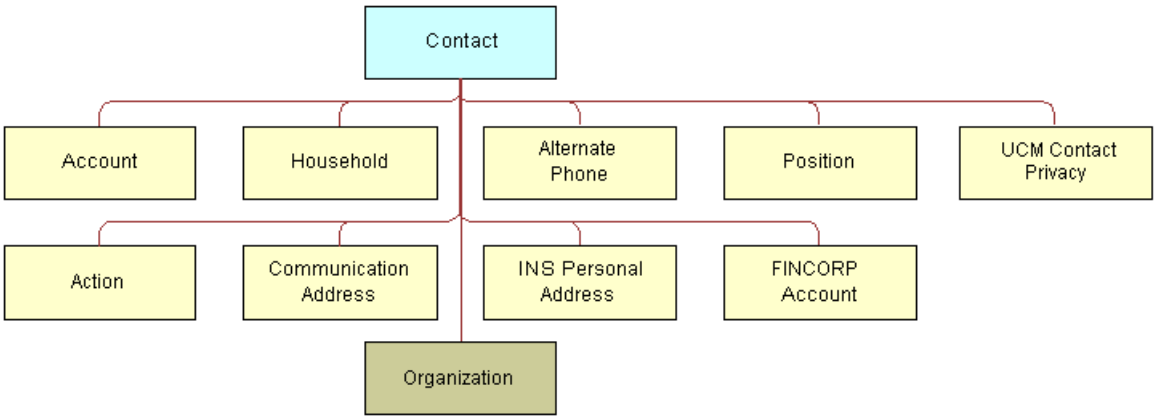

Figure 17. UCM Contact Structure

Table 55 lists the Contact Query Integration objects.

Table 55. Contact Query Integration Objects

Integration Object	Business Component	XML Tag	User Keys
CIFContact	Contact	LookUpContact	UK1:PartyUId UK2: Id
CIFContactRs	Contact	LookUpContactResponse	UK1:PartyUId UK2: Id

Table 56 lists the Contact Query by Name Integration objects.

Table 56. Contact Query by Name Integration Objects

Integration Object	Business Component	XML Tag	User Keys
CIFContactByName	Contact	LookUpContactByName	1:PartyUId2: Id
CIFContactByNameRs	Contact	LookUpContactByNameResponse	1:PartyUId2: Id

Table 57 lists the Contact Update Integration objects.

Table 57. Contact Update Integration Objects

Integration Object	Business Component	XML Tag	User Keys
CIFContactUpdate	Contact	UpdateContact	1UK1:PartyUId UK2: Id
CIFContactUpdateRs	Contact	UpdateContactResponse	UK1:PartyUId UK2: Id

Table 58 lists the Contact Insert Integration objects.

Table 58. Contact Insert Integration Objects

Integration Object	Business Component	XML Tag	User Keys
CIFContactInsert	Contact	InsertContact	UK1:PartyUId UK2: Id
CIFContactInsertRs	Contact	InsertContactResponse	1:UK1:PartyUId UK2: Id: Id

Table 59 lists the Contact Delete Integration objects.

Table 59. Contact Delete Integration Objects

Integration Object	Business Component	XML Tag	User Keys
CIFContactDelete	Contact	DeleteContact	UK1:PartyUId UK2: Id
CIFContactDeleteRs	Contact	DeleteContactResponse	UK1:PartyUId UK2: Id

Table 60 lists the Contact Integration Component and Integration Component Fields.

Table 60. Contact Integration Component and Integration Component Fields

Name	XML Tag	Data Type	Length	Req	Note
IC: Contact					
Alias	Alias	DTYPE_TEXT	50		

Table 60. Contact Integration Component and Integration Component Fields

Name	XML Tag	Data Type	Length	Req	Note
Assistant	Assistant	DTYPE_TEXT	50		
Assistant Phone #	AssistantPhone	DTYPE_TEXT	40		
Best Call Time	BestCallTime	DTYPE_TEXT	30		Bound by FINCORP_CONTACT_CALL_TIME
Cellular Phone #	CellularPhone	DTYPE_TEXT	40		
Customer Value	CustomerValue	DTYPE_TEXT	30		Bound by FIN_CON_VALUE_TYPE
Date of Birth	DateofBirth	DTYPE_DATE	7		
Email Address	EmailAddress	DTYPE_TEXT	50		
Email Address 2	EmailAddress2	DTYPE_TEXT	50		
Fax Phone #	FaxPhone	DTYPE_TEXT	40		
First Name	FirstName	DTYPE_TEXT	50	✔	
Home Phone #	HomePhone	DTYPE_TEXT	40		
Id	Id	DTYPE_ID	15	✔	This tag is required for Id cross-referencing. Tag stores incoming external Id.
Income	Income	DTYPE_CURRENCY	22		
Income Currency Code	IncomeCurrencyCode	DTYPE_TEXT	20		Used for picking a related entity.
Income Exchange Date	IncomeExchangeDate	DTYPE_DATE	7		
Integration Id	IntegrationId	DTYPE_TEXT	30		
Job Title	JobTitle	DTYPE_TEXT	75		
Last Name	LastName	DTYPE_TEXT	50	✔	
M/F	MF	DTYPE_TEXT	30		Bound by SEX_MF

Table 60. Contact Integration Component and Integration Component Fields

Name	XML Tag	Data Type	Length	Req	Note
M/M	MM	DTYPE_TEXT	15		Bound by MR_MS
Manager First Name	ManagerFirstName	DTYPE_TEXT	50		
Manager Id	ManagerId	DTYPE_ID	15		
Manager Integration Id	ManagerIntegrationId	DTYPE_TEXT	30		
Manager Last Name	ManagerLastName	DTYPE_TEXT	50		Used for picking a related entity
Manager Middle Name	ManagerMiddleName	DTYPE_TEXT	50		
Manager Primary Organization	ManagerPrimaryOrganization	DTYPE_TEXT	100		
Manager Primary Organization Id	ManagerPrimaryOrganization Id	DTYPE_ID	15		
Manager UId	ManagerUId	DTYPE_TEXT	100		
Marital Status	MaritalStatus	DTYPE_TEXT	30		Bound by MARITAL_STAT US
Middle Name	MiddleName	DTYPE_TEXT	50		
Mother Maiden Name	MotherMaidenName	DTYPE_TEXT	50		
Party UId	PartyUId	DTYPE_TEXT	100		
Person UId	PersonUId	DTYPE_TEXT	100		
Preferred Communications	PreferredCommunications	DTYPE_TEXT	30		Bound by OFFER_MEDIA
Preferred Language Code	PreferredLanguageCode	DTYPE_TEXT	20		
Privacy Code	PrivacyCode	DTYPE_TEXT	30		Bound by PRIVACY_CODE
Social Security Number	SocialSecurityNumber	DTYPE_TEXT	11		
Status	Status	DTYPE_TEXT	30		Bound by CONTACT_STAT US

Table 60. Contact Integration Component and Integration Component Fields

Name	XML Tag	Data Type	Length	Req	Note
Suppress All Calls	SuppressAllCalls	DTYPE_TEXT	1		
Suppress All Emails	SuppressAllEmails	DTYPE_TEXT	1		
Suppress All Faxes	SuppressAllFaxes	DTYPE_TEXT	1		
Suppress All Mailings	SuppressAllMailings	DTYPE_TEXT	1		
Time Zone Id	TimeZoneId	DTYPE_ID	15		
Time Zone Name	TimeZoneName	DTYPE_TEXT	100		Used for picking a related entity.
Work Phone #	WorkPhone	DTYPE_TEXT	40		
Work Phone Extension	WorkPhoneExtension	DTYPE_TEXT	10		
operation	operation	DTYPE_TEXT	30		System field
IC: Account					
Account	Account	DTYPE_TEXT	100	✔	
Account Location	AccountLocation	DTYPE_TEXT	50		
Account Organization	AccountOrganization	DTYPE_TEXT	100		
Account Party UId	AccountPartyUId	DTYPE_TEXT	100		
Account Row Id	AccountRowId	DTYPE_ID	15		
Account Status	AccountStatus	DTYPE_TEXT	24		Bound by ACCOUNT_STATUS
IsPrimaryMVG	IsPrimaryMVG	DTYPE_TEXT	1		
Related Account Integration Id	RelatedAccountIntegrationId	DTYPE_TEXT	30		
operation	operation	DTYPE_TEXT	30		
Alternate Phone					
Alternate Phone #	AlternatePhone	DTYPE_TEXT	40	✔	
Alternate Phone Id	AlternatePhoneId	DTYPE_ID	15		

Table 60. Contact Integration Component and Integration Component Fields

Name	XML Tag	Data Type	Length	Req	Note
Alternate Phone Integration Id	AlternatePhoneIntegrationId	DTYPE_TEXT	30		
Alternate Phone Medium Type	AlternatePhoneMediumType	DTYPE_TEXT	30		
Alternate Phone Name	AlternatePhoneName	DTYPE_TEXT	50		
IsPrimaryMVG	IsPrimaryMVG	DTYPE_TEXT	1		
operation	operation	DTYPE_TEXT	30		System field
IC: Communication Address					
Alternate Email Address	AlternateEmailAddress	DTYPE_TEXT	100	✔	
Communication Address Id	CommunicationAddressId	DTYPE_ID	15		
Communication Address Integration Id	CommunicationAddressIntegrationId	DTYPE_TEXT	30		
Communication Address Medium Type	CommunicationAddressMediumType	DTYPE_TEXT	30		
Communication Address Name	CommunicationAddressName	DTYPE_TEXT	50	✔	
IsPrimaryMVG	IsPrimaryMVG	DTYPE_TEXT	1		
operation	operation	DTYPE_TEXT	30		System field
IC: Household					
Household Id	HouseholdId	DTYPE_ID	15		
Household Integration Id	HouseholdIntegrationId	DTYPE_TEXT	30		
Household Name	HouseholdName	DTYPE_TEXT	50		
Household Number	HouseholdNumber	DTYPE_TEXT	100		
Household Party UId	HouseholdPartyUId	DTYPE_TEXT	100		
Household Status	HouseholdStatus	DTYPE_TEXT	30		Bound by HOUSEHOLD_STATUS
IsPrimaryMVG	IsPrimaryMVG	DTYPE_TEXT	1		

Table 60. Contact Integration Component and Integration Component Fields

Name	XML Tag	Data Type	Length	Req	Note
operation	operation	DTYPE_TEXT	30		
INS Personal Address					
INS Personal Address Id	INSPersonalAddressId	DTYPE_ID	15		
INS Personal Address Name	INSPersonalAddressName	DTYPE_TEXT	100		
INS Personal Address Type	INSPersonalAddressType	DTYPE_TEXT	30		
INS Personal City	INSPersonalCity	DTYPE_TEXT	50	✔	
INS Personal Country	INSPersonalCountry	DTYPE_TEXT	30		Bound by COUNTRY
INS Personal County	INSPersonalCounty	DTYPE_TEXT	50		
INS Personal Integration Id	INSPersonalIntegrationId	DTYPE_TEXT	30		
INS Personal Postal Code	INSPersonalPostalCode	DTYPE_TEXT	30		
INS Personal State	INSPersonalState	DTYPE_TEXT	10		Bound by STATE_ABBREV
INS Personal Street Address	INSPersonalStreetAddress	DTYPE_TEXT	200	✔	
INS Personal Street Address 2	INSPersonalStreetAddress2	DTYPE_TEXT	100		
IsPrimaryMVG	IsPrimaryMVG	DTYPE_TEXT	1		
operation	operation	DTYPE_TEXT	30		System field
IC: Organization					
Division	Division	DTYPE_TEXT	255		
IsPrimaryMVG	IsPrimaryMVG	DTYPE_TEXT	1		
Position	Position	DTYPE_TEXT	50		
Position Id	PositionId	DTYPE_ID	15		
Position Integration Id	PositionIntegrationId	DTYPE_TEXT	30		
Sales Rep	SalesRep	DTYPE_TEXT	50		
operation	operation	DTYPE_TEXT	30		System field

Table 60. Contact Integration Component and Integration Component Fields

Name	XML Tag	Data Type	Length	Req	Note
IC: Position					
Division	Division	DTYPE_TEXT	255		
IsPrimaryMVG	IsPrimaryMVG	DTYPE_TEXT	1		
operation	operation	DTYPE_TEXT	30		System field
Position	Position	DTYPE_TEXT	50		
Position Id	PositionId	DTYPE_ID	15		
Position Integration Id	PositionIntegrationId	DTYPE_TEXT	30		
Sales Rep	SalesRep	DTYPE_TEXT	50		
IC: Action					
Account Id	AccountId	DTYPE_ID	15		
Account Location	AccountLocation	DTYPE_TEXT	50		
Account Name	AccountName	DTYPE_TEXT	100		Used for picking a related entity.
Audience	Audience	DTYPE_TEXT	30		Bound by ACTIVITY_AUDIENCE_TYPE
Category	Category	DTYPE_TEXT	30		Bound by TODO_TYPE
Description	Description	DTYPE_TEXT	100		
Done	Done	DTYPE_UTC DATETIME	7		
Done Flag	DoneFlag	DTYPE_TEXT	1		
Due	Due	DTYPE_UTC DATETIME	7		
Duration Minutes	DurationMinutes	DTYPE_INTEGER	22		Bound by APPOINTMENT_DURATION
Id	Id	DTYPE_ID	15		
Integration Id	IntegrationId	DTYPE_TEXT	30		
MeetingLocation	MeetingLocation	DTYPE_TEXT	100		
No Sooner Than Date	NoSoonerThanDate	DTYPE_UTC DATETIME	7		
operation	operation	DTYPE_TEXT	30		System field

Table 60. Contact Integration Component and Integration Component Fields

Name	XML Tag	Data Type	Length	Req	Note
Planned	Planned	DTYPE_UTC DATETIME	7		
Started	Started	DTYPE_UTC DATETIME	7		
Status	Status	DTYPE_TEXT	30		Bound by EVENT_STATUS
Type	Type	DTYPE_TEXT	30	✔	Bound by TODO_TYPE
IC: FINCORP Account					
Account Branch ABA	AccountBranchABA	DTYPE_TEXT	50		Used for picking a related entity.
Account Branch Id	AccountBranchId	DTYPE_ID	15		
Account Holder	AccountHolder	DTYPE_TEXT	100		Used for picking a related entity.
Account Name	AccountName	DTYPE_TEXT	100		
Account Number	AccountNumber3	DTYPE_TEXT	100	✔	
Account Status	AccountStatus2	DTYPE_TEXT	30		Bound by FINS_CRPST_LFLN_IPFS_PLC_MLOV
Account Type	AccountType	DTYPE_TEXT	30		Bound by FINCORP_PROD_ADMIN_CLASS_MLOV
Acct Gen - Branch Id	AcctGen-BranchId	DTYPE_ID	15		
Available Balance	AvailableBalance	DTYPE_CURRENCY	22		
Billing Option	BillingOption	DTYPE_TEXT	30		Bound by FINCORP_ACCOUNT_BILL_OPTION
Branch	Branch	DTYPE_TEXT	50		
Current Balance	CurrentBalance	DTYPE_CURRENCY	22		
Id	Id	DTYPE_ID	15		

Table 60. Contact Integration Component and Integration Component Fields

Name	XML Tag	Data Type	Length	Req	Note
Integration Id	IntegrationId	DTYPE_TEXT	30		
Name	Name	DTYPE_TEXT	100	✔	Used for picking a related entity
operation	operation	DTYPE_TEXT	30		System field
Organization Id	OrganizationId2	DTYPE_ID	15		
Product Id	ProductId	DTYPE_ID	15	✔	
Relationship Limit	RelationshipLimit	DTYPE_CURRENCY	22		
Start Date	StartDate	DTYPE_DATETIME	7		
IC: UCM Contact Privacy					
Affiliate Flag	AffiliateFlag	DTYPE_TEXT	30		
Affiliate Method	AffiliateMethod	DTYPE_TEXT	30		
Affiliate Source	AffiliateSource	DTYPE_TEXT	30		
Affiliate System	AffiliateSystem	DTYPE_TEXT	30		
Channel Flag	ChannelFlag	DTYPE_TEXT	30		
Channel Method	ChannelMethod	DTYPE_TEXT	30		
Channel Source	ChannelSource	DTYPE_TEXT	30		
Channel System	ChannelSystem	DTYPE_TEXT	30		
Contact Id	ContactId	DTYPE_ID			
Cover Letter Type	CoverLetterType	DTYPE_TEXT	30		
Cover Letter Type II	CoverLetterTypeII	DTYPE_TEXT	30		
Id	Id	DTYPE_ID	30		
Integration Id	IntegrationId	DTYPE_TEXT	30		
Nonaffiliate Flag	NonaffiliateFlag	DTYPE_TEXT	30		
Nonaffiliate Method	NonaffiliateMethod	DTYPE_TEXT	30		
Nonaffiliate Source	NonaffiliateSource	DTYPE_TEXT	30		
Nonaffiliate System	NonaffiliateSystem	DTYPE_TEXT	30		

Table 60. Contact Integration Component and Integration Component Fields

Name	XML Tag	Data Type	Length	Req	Note
Notice Send Flag	NoticeSendFlag	DTYPE_TEXT	30		
Notice Sent Date	NoticeSentDate	DTYPE_UTC DATETIME			
Notice Sent Reason	NoticeSentReason	DTYPE_TEXT	100		
Notice Sent Type	NoticeSentType	DTYPE_TEXT	30		
OFAC Flag	OFACFlag	DTYPE_BOO L			
PEND End Date	PENDEndDate	DTYPE_UTC DATETIME			
PEND Start Date	PENDStartDate	DTYPE_UTC DATETIME			
Privacy Parent Type	PrivacyParentType	DTYPE_TEXT	30		
Privacy State Code	PrivacyStateCode	DTYPE_TEXT	30		
Receive Annual Notice Flag	ReceiveAnnualNoticeFlag	DTYPE_TEXT	30		
Receive Annual Notice Method	ReceiveAnnualNoticeMethod	DTYPE_TEXT	30		
Receive Annual Notice Source	ReceiveAnnualNoticeSource	DTYPE_TEXT	30		
Receive Annual Notice System	ReceiveAnnualNoticeSystem	DTYPE_TEXT	30		
State PEND Period	StatePENDPeriod	DTYPE_INTE GER			
Telemarketing Flag	TelemarketingFlag	DTYPE_TEXT	30		
Telemarketing Method	TelemarketingMethod	DTYPE_TEXT	30		
Telemarketing Source	TelemarketingSource	DTYPE_TEXT	30		
Telemarketing System	TelemarketingSystem	DTYPE_TEXT	30		

Table 60. Contact Integration Component and Integration Component Fields

Name	XML Tag	Data Type	Length	Req	Note
Updated	Updated	DTYPE_DATETIME	30		
Updated By	UpdatedBy	DTYPE_ID	30		

Household

UCM Household messages are used in insert, query, update, and delete operations to maintain Household information. LookUpHousehold and LookUpHouseholdByName are used for query operations. UpdateHousehold, InsertHoushold, and DeleteHousehold are message pairs for request and response operations. Figure 18 describes the structure of this message.

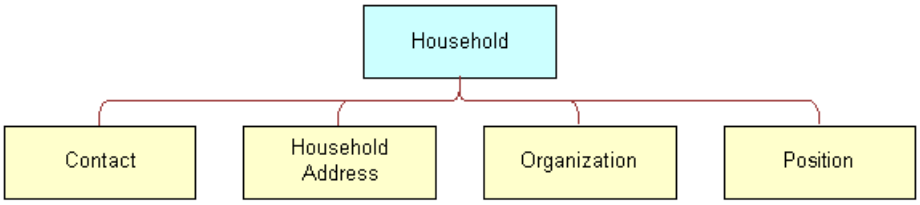

Figure 18. UCM Household Structure

Table 61 lists the Household Query Integration objects.

Table 61. Household Query Integration Objects

Integration Object	Business Component	XML Tag	User Keys
CIFHousehold	Household	LookUpHousehold	UK1:PartyUId UK2: Id
CIFHouseholdRs	Household	LookUpHouseholdResponse	UK1:PartyUId UK2: Id

Table 62 lists the Household Query by Name Integration objects.

Table 62. Household Query by Name Integration Objects

Integration Object	Business Component	XML Tag	User Keys
CIFHouseholdByName	Household	LookUpHouseholdByName	UK1:PartyUId UK2: Id
CIFHouseholdByNameRs	Household	LookUpHouseholdByNameResponse	UK1:PartyUId UK2: Id

Table 63 lists the Household Update Integration objects.

Table 63. Household Update Integration Objects

Integration Object	Business Component	XML Tag	User Keys
CIFHouseholdUpdate	Household	UpdateHousehold	UK1:PartyUId UK2: Id
CIFHouseholdUpdateRs	Household	UpdateHouseholdResponse	UK1:PartyUId UK2: Id

Table 64 lists the Household Insert Integration objects.

Table 64. Household Insert Integration Objects

Integration Object	Business Component	XML Tag	User Keys
CIFHouseholdInsert	Household	InsertHousehold	UK1:PartyUId UK2: Id
CIFHouseholdInsertRs	Household	InsertHouseholdResponse	UK1:PartyUId UK2: Id

Table 65 lists the Household Delete Integration objects.

Table 65. Household Delete Integration Objects

Integration Object	Business Component	XML Tag	User Keys
CIFHouseholdDelete	Household	DeleteHousehold	UK1:PartyUId UK2: Id
CIFHouseholdDeleteRs	Household	DeleteHouseholdResponse	UK1:PartyUId UK2: Id

Table 66 lists the Household Integration Component and Integration Component Fields.

Table 66. Household Integration Component and Integration Component Fields

Name	XML Tag	Data Type	Length	Req	Note
IC: Household					
Alias	Alias	DTYPE_TEXT	50		
Category	Category	DTYPE_TEXT	30		Bound by HOUSEHOLD_CATEGORIES
Currency Code	CurrencyCode	DTYPE_TEXT	20		Used for picking a related entity.
Fax Number	FaxNumber	DTYPE_TEXT	40		
Household Name	HouseholdName	DTYPE_TEXT	50		
Household Number	HouseholdNumber	DTYPE_TEXT	100		
Household Size	HouseholdSize	DTYPE_INTEGER	22		
Household Wealth	HouseholdWealth	DTYPE_CURRENCY	22		
Id	Id	DTYPE_ID	30	Y	This tag is required for Id cross-referencing. Tag stores incoming external Id.
Income	Income	DTYPE_CURRENCY	7		

Table 66. Household Integration Component and Integration Component Fields

Name	XML Tag	Data Type	Length	Req	Note
Integration Id	IntegrationId	DTYPE_TEXT	30		
Party UId	PartyUId	DTYPE_TEXT	100		
Phone Number	PhoneNumber	DTYPE_TEXT	40		
Privacy Code	PrivacyCode	DTYPE_TEXT	30		Bound by PRIVACY_CODE
Revenue	Revenue	DTYPE_CURRENCY	22		
Segment	Segment	DTYPE_TEXT	30		Bound by HOUSEHOLD_SEGMENT
Status	Status	DTYPE_TEXT	30		Bound by HOUSEHOLD_STATUS
Suppress All Calls	SuppressAllCalls	DTYPE_TEXT	1		
Suppress All Mailings	SuppressAllMailings	DTYPE_TEXT	1		
Type	Type	DTYPE_TEXT	30		Bound by HOUSEHOLD_TYPE
operation	operation	DTYPE_TEXT	30		
IC: Contact					
Contact Integration Id	ContactIntegrationId	DTYPE_TEXT	30		
Contact Party UId	ContactPartyUId	DTYPE_TEXT	100		
DOB	DOB	DTYPE_DATE	7		
First Name	FirstName	DTYPE_TEXT	50	✔	
IsPrimaryMVG	IsPrimaryMVG	DTYPE_TEXT	1		
Last Name	LastName	DTYPE_TEXT	50	✔	
M/M	MM	DTYPE_TEXT	15		Bound by MR_MS
Middle Name	MiddleName	DTYPE_TEXT	50		
Person UId	PersonUId	DTYPE_TEXT	100		

Table 66. Household Integration Component and Integration Component Fields

Name	XML Tag	Data Type	Length	Req	Note
Related Contact Id	RelatedContactId	DTYPE_ID	15		
Related Date Entered Household	RelatedDateEnteredHousehold	DTYPE_UTCDATETIME	7		
Related Date Exited Household	RelatedDateExitedHousehold	DTYPE_UTCDATETIME	7		
Related Primary Organization Id	RelatedPrimaryOrganizationId	DTYPE_ID	15		
Relation to Household	RelationtoHousehold	DTYPE_TEXT	30		Bound by REL_TO_HOUSEHOLD
SSN	SSN	DTYPE_TEXT	11		
operation	operation	DTYPE_TEXT	30		System field
IC: Household Address					
Address Integration Id	AddressIntegrationId	DTYPE_TEXT	30		
Address Name	AddressName	DTYPE_TEXT	100		
Address Row Id	AddressRowId	DTYPE_ID	15		
City	City	DTYPE_TEXT	50	✔	
Country	Country	DTYPE_TEXT	30		Bound by COUNTRY
IsPrimaryMVG	IsPrimaryMVG	DTYPE_TEXT	1		
operation	operation	DTYPE_TEXT	30		System field
Postal code	Postalcode	DTYPE_TEXT	30		
State	State	DTYPE_TEXT	10		Bound by STATE_ABBREV
Street Address	StreetAddress	DTYPE_TEXT	200	✔	
Street Address 2	StreetAddress2	DTYPE_TEXT	100		
IC: Organization					
IsPrimaryMVG	IsPrimaryMVG	DTYPE_TEXT	1		

Table 66. Household Integration Component and Integration Component Fields

Name	XML Tag	Data Type	Length	Req	Note
operation	operation	DTYPE_TEXT	30		System field
Organization	Organization	DTYPE_TEXT	50		
Organization Id	OrganizationId	DTYPE_ID	30		
Organization Integration Id	OrganizationIntegrationId	DTYPE_TEXT	30		
IC: Position					
IsPrimaryMVG	IsPrimaryMVG	DTYPE_TEXT	1		
operation	operation	DTYPE_TEXT	30		System field
Position	Position	DTYPE_TEXT	50		
Position Division	PositionDivision	DTYPE_TEXT	255		
Position Id	PositionId	DTYPE_ID	15		
Position Integration Id	PositionIntegrationId	DTYPE_TEXT	30		
Team	Team	DTYPE_TEXT	50		

Personal Address

UCM Personal Address messages are used for requesting Personal Address information. They are a message pair for request and response operations. Figure 19 describes the structure of this message.

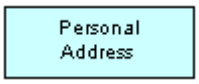

Figure 19. UCM Personal Address Structure

Table 67 lists the Personal Address Query Integration objects.

Table 67. Personal Address Query Integration Objects

Integration Object	Business Component	XML Tag	User Keys
CIFPersonalAddress	Personal Address	LookUpPersonalAddress	1: Street Address, City,State, Postal Code
CIFPersonalAddressRs	Personal Address	LookUpPersonalAddressResponse	1: Street Address, City,State, Postal Code

Table 68 lists the Personal Address Integration Component and Integration Component Fields.

Table 68. Personal Address Integration Component and Integration Component Fields

Name	XML Tag	Data Type	Length	Req	Note
IC: Personal Address					
Address Description	AddressDescription	DTYPE_TEXT	30		Bound by FIN_CON_ADDRESS_DESC
Address Name	AddressName	DTYPE_TEXT	100		
Address Type	AddressType	DTYPE_TEXT	30		Bound by FIN_CON_ADDRESS_TYPE
Apartment Number	ApartmentNumber	DTYPE_TEXT	5		
City	City	DTYPE_TEXT	50	✔	
Country	Country	DTYPE_TEXT	30		Bound by COUNTRY
County	County	DTYPE_TEXT	50		
Integration Id	IntegrationId	DTYPE_TEXT	30		
operation	operation	DTYPE_TEXT	30		System field
Postal Code	PostalCode	DTYPE_TEXT	30		
State	State	DTYPE_TEXT	10		Bound by STATE_ABBREV
Street Address	StreetAddress	DTYPE_TEXT	200	✔	
Street Address 2	StreetAddress2	DTYPE_TEXT	100		
Zip Code	ZipCode	DTYPE_TEXT	30		

FINCORP Account

UCM FINCORP Account messages are used for requesting FINCORP Account information. They are a message pair for request and response operations. Figure 20 describes the structure of this message.

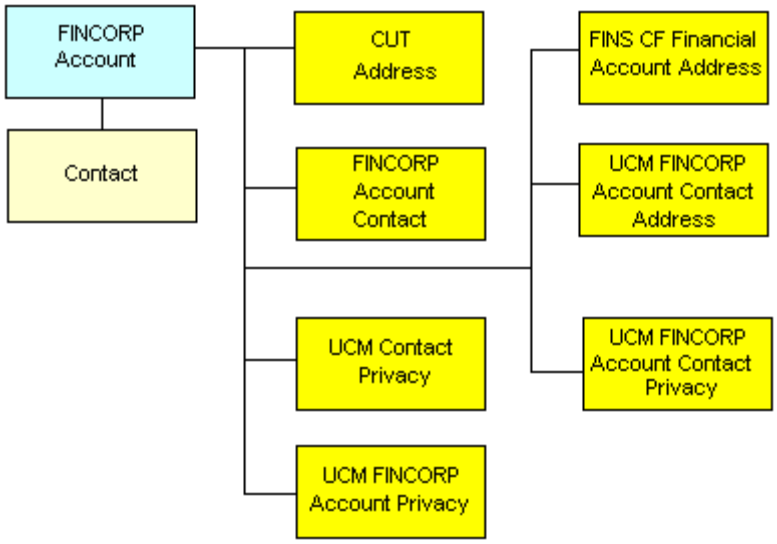

Figure 20. UCM FINCORP Account Structure

Table 69 lists the FINCORP Account Query Integration objects.

Table 69. FINCORP Account Query Integration Objects

Integration Object	Business Component	XML Tag	User Keys
CIFFINCORPAccount	FINCORP Account	LookUpFINCORPAccount	UK1: Account Number
CIFFINCORPAccountRs	FINCORP Account	LookUpFINCORPAccountResponse	UK1: Account Number

Table 70 lists the FINCORP Account Integration Component and Integration Component Fields.

Table 70. FINCORP Account Integration Component and Integration Component Fields

Name	XML Tag	Data Type	Length	Req	Notes
IC: FINCORP Account					
Account Branch ABA	AccountBranchABA	DTYPE_TEXT	50	N	

Table 70. FINCORP Account Integration Component and Integration Component Fields

Name	XML Tag	Data Type	Length	Req	Notes
Account Branch Id	AccountBranchId	DTYPE_ID		N	
Account Holder	AccountHolder	DTYPE_TEXT		N	
Account Name	AccountName	DTYPE_TEXT	100	N	
Account Number	AccountNumber	DTYPE_TEXT	100	Y	
Account Status	AccountStatus	DTYPE_TEXT	30	N	
Account Type	AccountType	DTYPE_TEXT	30	N	
Acct Gen - Branch Id	AcctGen-BranchId	DTYPE_ID		N	
Acct Gen - Primary Contact Id	AcctGen-PrimaryContactId	DTYPE_ID		N	
Available Balance	AvailableBalance	DTYPE_TEXT		N	
Billing Option	BillingOption	DTYPE_TEXT	30	N	
Branch	Branch	DTYPE_TEXT	50	N	
CA Next Statement Date	CANextStatementDate	DTYPE_DATETIME		N	
Commitment Amount	CommitmentAmount	DTYPE_TEXT		N	
Conflict Id	ConflictId	DTYPE_ID	30	N	
Created	Created	DTYPE_DATETIME	30	N	
Current Balance	CurrentBalance	DTYPE_TEXT		N	
Id	Id	DTYPE_ID	30	N	
Integration Id	IntegrationId	DTYPE_TEXT	30	N	
Mod Id	ModId	DTYPE_ID	30	N	
Name	Name	DTYPE_TEXT	100	N	
operation	operation	DTYPE_TEXT	30	N	
Organization Id	OrganizationId	DTYPE_ID		N	
Owner Type	OwnerType	DTYPE_TEXT	30	N	
Parent ABA Number	ParentABANumber	DTYPE_TEXT	50	N	

Table 70. FINCORP Account Integration Component and Integration Component Fields

Name	XML Tag	Data Type	Length	Req	Notes
Parent Asset ID	ParentAssetID	DTYPE_ID		N	
Parent Asset Number	ParentAssetNumber	DTYPE_TEXT	100	N	
Primary Account Id	PrimaryAccountId	DTYPE_ID		N	
Product Id	ProductId	DTYPE_ID		Y	
Relationship	Relationship	DTYPE_TEXT	30	N	
Relationship Limit	RelationshipLimit	DTYPE_TEXT		N	
searchspec	searchspec	DTYPE_TEXT	250	N	
Start Date	StartDate	DTYPE_DATETIME		N	
Sub Type	SubType	DTYPE_TEXT	30	N	
Type	Type	DTYPE_TEXT	30	N	
Updated	Updated	DTYPE_UTCDATETIME	30	N	
Updated By	UpdatedBy	DTYPE_ID	30	N	
IC: Contact					
Account Id	AccountId	DTYPE_ID		N	
Account Integration Id	AccountIntegrationId	DTYPE_TEXT	30	N	
Assistant	Assistant	DTYPE_TEXT	50	N	
Assistant Phone #	AssistantPhone	DTYPE_TEXT		N	
Cellular Phone #	CellularPhone	DTYPE_TEXT		N	
Fax Phone #	FaxPhone	DTYPE_TEXT		N	
First Name	FirstName	DTYPE_TEXT	50	N	
Home Phone #	HomePhone	DTYPE_TEXT		N	
Id	Id	DTYPE_ID	100	N	
Integration Id	IntegrationId	DTYPE_TEXT	30	N	
Job Title	JobTitle	DTYPE_TEXT	75	N	
Last Name	LastName	DTYPE_TEXT	50	N	

Table 70. FINCORP Account Integration Component and Integration Component Fields

Name	XML Tag	Data Type	Length	Req	Notes
M/M	MM	DTYPE_TEXT	15	N	
Market Cap Pref	MarketCapPref	DTYPE_TEXT	30	N	
Middle Name	MiddleName	DTYPE_TEXT	50	N	
Mod Id	ModId	DTYPE_ID	30	N	
operation	operation	DTYPE_TEXT	30	N	
Party UId	PartyUId	DTYPE_TEXT	30	N	
Person UId	PersonUId	DTYPE_ID	30	N	
Preferred Communication Method	PreferredCommunication Method	DTYPE_TEXT	20	N	
Primary Personal Address Id	PrimaryPersonalAddressI d	DTYPE_ID		N	
Privacy Code	PrivacyCode	DTYPE_TEXT	30	N	
searchspec	searchspec	DTYPE_TEXT	250	N	
Suppress All Calls	SuppressAllCalls	DTYPE_BOOL		N	
Suppress All Emails	SuppressAllEmails2	DTYPE_BOOL		N	
Suppress All Faxes	SuppressAllFaxes	DTYPE_BOOL		N	
Suppress All Mailings	SuppressAllMailings	DTYPE_BOOL		N	
Updated	Updated	DTYPE_DATETI ME	30	N	
Updated By	UpdatedBy	DTYPE_ID	30	N	
Work Phone #	WorkPhone	DTYPE_TEXT		N	
IC:FINCORP Account Contact					
Asset Id	AssetId	DTYPE_ID		N	
Contact Id	ContactId	DTYPE_ID		N	
Id	Id	DTYPE_ID	30	N	
Status	Status	DTYPE_TEXT	30	N	
Updated	Updated	DTYPE_DATETI ME	30	N	

Table 70. FINCORP Account Integration Component and Integration Component Fields

Name	XML Tag	Data Type	Length	Req	Notes
Updated By	UpdatedBy	DTYPE_ID	30	N	
IC: CUT Address					
Address Name	AddressName	DTYPE_TEXT	100	Y	
City	City	DTYPE_TEXT	50	Y	
Country	Country	DTYPE_TEXT	30	N	
County	County	DTYPE_TEXT	50	N	
Id	Id	DTYPE_ID	30	N	
Integration Id	IntegrationId	DTYPE_TEXT	30	N	
operation	operation	DTYPE_TEXT	30	N	
Postal Code	PostalCode	DTYPE_TEXT	30	N	
searchspec	searchspec	DTYPE_TEXT	250	N	
State	State	DTYPE_TEXT	10	N	
Street Address	StreetAddress	DTYPE_TEXT	200	Y	
Street Address 2	StreetAddress2	DTYPE_TEXT	100	N	
Updated	Updated	DTYPE_DATETIME	30	N	
Updated By	UpdatedBy	DTYPE_ID	30	N	
IC: FINS CF Financial Account Address					
Active Flag	ActiveFlag	DTYPE_BOOL		N	
Addr Id	AddrId	DTYPE_ID		N	
City	City	DTYPE_TEXT	50	N	
Contact Id	ContactId	DTYPE_ID		N	
Country	Country	DTYPE_TEXT	30	N	
Fin Acct Id	FinAcctId	DTYPE_ID		N	
First Name	FirstName	DTYPE_TEXT	50	N	
Id	Id	DTYPE_ID	30	N	
Last Name	LastName	DTYPE_TEXT	50	N	
operation	operation	DTYPE_TEXT	30	N	
searchspec	searchspec	DTYPE_TEXT	250	N	
State	State	DTYPE_TEXT	10	N	

Table 70. FINCORP Account Integration Component and Integration Component Fields

Name	XML Tag	Data Type	Length	Req	Notes
Street	Street	DTYPE_TEXT	200	Y	
Updated	Updated	DTYPE_DATETIME	30	N	
Updated By	UpdatedBy	DTYPE_ID	30	N	
Use Type	UseType	DTYPE_TEXT	30	Y	
Zipcode	Zipcode2	DTYPE_TEXT	30	N	
IC: UCM Contact Privacy					
Affiliate Flag	AffiliateFlag	DTYPE_TEXT	30	N	
Affiliate Method	AffiliateMethod	DTYPE_TEXT	30	N	
Affiliate Source	AffiliateSource	DTYPE_TEXT	30	N	
Affiliate System	AffiliateSystem	DTYPE_TEXT	30	N	
Channel Flag	ChannelFlag	DTYPE_TEXT	30	N	
Channel Method	ChannelMethod	DTYPE_TEXT	30	N	
Channel Source	ChannelSource	DTYPE_TEXT	30	N	
Channel System	ChannelSystem	DTYPE_TEXT	30	N	
Contact Id	ContactId	DTYPE_ID		N	
Cover Letter Type	CoverLetterType	DTYPE_TEXT	30	N	
Cover Letter Type II	CoverLetterTypeII	DTYPE_TEXT	30	N	
Id	Id	DTYPE_ID	30	N	
Integration Id	IntegrationId	DTYPE_TEXT	30	N	
Nonaffiliate Flag	NonaffiliateFlag	DTYPE_TEXT	30	N	
Nonaffiliate Method	NonaffiliateMethod	DTYPE_TEXT	30	N	
Nonaffiliate Source	NonaffiliateSource	DTYPE_TEXT	30	N	
Nonaffiliate System	NonaffiliateSystem	DTYPE_TEXT	30	N	
Notice Send Flag	NoticeSendFlag	DTYPE_TEXT	30	N	
Notice Sent Date	NoticeSentDate	DTYPE_UTCDATETIME		N	

Table 70. FINCORP Account Integration Component and Integration Component Fields

Name	XML Tag	Data Type	Length	Req	Notes
Notice Sent Reason	NoticeSentReason	DTYPE_TEXT	100	N	
Notice Sent Type	NoticeSentType	DTYPE_TEXT	30	N	
OFAC Flag	OFACFlag	DTYPE_BOOL		N	
PEND End Date	PENDEndDate	DTYPE_UTCDATETIME		N	
PEND Start Date	PENDStartDate	DTYPE_UTCDATETIME		N	
Privacy Parent Type	PrivacyParentType	DTYPE_TEXT	30	Y	
Privacy State Code	PrivacyStateCode	DTYPE_TEXT	30	N	
Receive Annual Notice Flag	ReceiveAnnualNoticeFlag	DTYPE_TEXT	30	N	
Receive Annual Notice Method	ReceiveAnnualNoticeMethod	DTYPE_TEXT	30	N	
Receive Annual Notice Source	ReceiveAnnualNoticeSource	DTYPE_TEXT	30	N	
Receive Annual Notice System	ReceiveAnnualNoticeSystem	DTYPE_TEXT	30	N	
State PEND Period	StatePENDPeriod	DTYPE_INTEGER		N	
Telemarketing Flag	TelemarketingFlag	DTYPE_TEXT	30	N	
Telemarketing Method	TelemarketingMethod	DTYPE_TEXT	30	N	
Telemarketing Source	TelemarketingSource	DTYPE_TEXT	30	N	
Telemarketing System	TelemarketingSystem	DTYPE_TEXT	30	N	
Updated	Updated	DTYPE_DATETIME	30	N	
Updated By	UpdatedBy	DTYPE_ID	30	N	
IC:UCM FINCORP Account Contact Address					
Address Type	AddressType	DTYPE_TEXT	30	N	

Table 70. FINCORP Account Integration Component and Integration Component Fields

Name	XML Tag	Data Type	Length	Req	Notes
City	City	DTYPE_TEXT	50	N	
Comments	Comments	DTYPE_TEXT	200	N	
Country	Country	DTYPE_TEXT	30	N	
Financial Account Contact Id	FinancialAccountContactId	DTYPE_ID		N	
Id	Id	DTYPE_ID	30	N	
operation	operation	DTYPE_TEXT	30	N	
Personal Address Id	PersonalAddressId	DTYPE_ID		Y	
Postal Code	PostalCode	DTYPE_TEXT	30	N	
searchspec	searchspec	DTYPE_TEXT	250	N	
State	State	DTYPE_TEXT	10	N	
Street Address	StreetAddress	DTYPE_TEXT	200	N	
Street Address 2	StreetAddress2	DTYPE_TEXT	200	N	
Updated	Updated	DTYPE_DATETIME	30	N	
Updated By	UpdatedBy	DTYPE_ID	30	N	
IC: UCM FINCORP Account Contact Privacy					
Affiliate Flag	AffiliateFlag	DTYPE_TEXT	30	N	
Affiliate Method	AffiliateMethod	DTYPE_TEXT	30	N	
Affiliate Source	AffiliateSource	DTYPE_TEXT	30	N	
Affiliate System	AffiliateSystem	DTYPE_TEXT	30	N	
Channel Flag	ChannelFlag	DTYPE_TEXT	30	N	
Channel Method	ChannelMethod	DTYPE_TEXT	30	N	
Channel Source	ChannelSource	DTYPE_TEXT	30	N	
Channel System	ChannelSystem	DTYPE_TEXT	30	N	
Cover Letter Type	CoverLetterType	DTYPE_TEXT	30	N	
Cover Letter Type II	CoverLetterTypeII	DTYPE_TEXT	30	N	

Table 70. FINCORP Account Integration Component and Integration Component Fields

Name	XML Tag	Data Type	Length	Req	Notes
Financial Account Contact Id	FinancialAccountContactId	DTYPE_ID		N	
Financial Account Id	FinancialAccountId	DTYPE_ID		N	
Id	Id	DTYPE_ID	30	N	
Integration Id	IntegrationId	DTYPE_TEXT	30	N	
Nonaffiliate Flag	NonaffiliateFlag	DTYPE_TEXT	30	N	
Nonaffiliate Method	NonaffiliateMethod	DTYPE_TEXT	30	N	
Nonaffiliate Source	NonaffiliateSource	DTYPE_TEXT	30	N	
Nonaffiliate System	NonaffiliateSystem	DTYPE_TEXT	30	N	
Notice Send Flag	NoticeSendFlag	DTYPE_TEXT	30	N	
Notice Sent Date	NoticeSentDate	DTYPE_UTCDATETIME		N	
Notice Sent Reason	NoticeSentReason	DTYPE_TEXT	100	N	
Notice Sent Type	NoticeSentType	DTYPE_TEXT	30	N	
OFAC Flag	OFACFlag	DTYPE_BOOL		N	
operation	operation	DTYPE_TEXT	30	N	
PEND Flag	PENDFlag	DTYPE_BOOL		N	
Privacy Parent Type	PrivacyParentType	DTYPE_TEXT	30	Y	
Privacy State Code	PrivacyStateCode	DTYPE_TEXT	30	N	
Receive Annual Notice Flag	ReceiveAnnualNoticeFlag	DTYPE_TEXT	30	N	
Receive Annual Notice Method	ReceiveAnnualNoticeMethod	DTYPE_TEXT	30	N	
Receive Annual Notice Source	ReceiveAnnualNoticeSource	DTYPE_TEXT	30	N	

Table 70. FINCORP Account Integration Component and Integration Component Fields

Name	XML Tag	Data Type	Length	Req	Notes
Receive Annual Notice System	ReceiveAnnualNoticeSystem	DTYPE_TEXT	30	N	
searchspec	searchspec	DTYPE_TEXT	250	N	
Telemarketing Flag	TelemarketingFlag	DTYPE_TEXT	30	N	
Telemarketing Method	TelemarketingMethod	DTYPE_TEXT	30	N	
Telemarketing Source	TelemarketingSource	DTYPE_TEXT	30	N	
Telemarketing System	TelemarketingSystem	DTYPE_TEXT	30	N	
Updated	Updated	DTYPE_DATETIME	30	N	
Updated By	UpdatedBy	DTYPE_ID	30	N	
IC: UCM FINCORP Account Privacy					
Affiliate Flag	AffiliateFlag	DTYPE_TEXT	30	N	
Affiliate Method	AffiliateMethod	DTYPE_TEXT	30	N	
Affiliate Source	AffiliateSource	DTYPE_TEXT	30	N	
Affiliate System	AffiliateSystem	DTYPE_TEXT	30	N	
Channel Flag	ChannelFlag	DTYPE_TEXT	30	N	
Channel Method	ChannelMethod	DTYPE_TEXT	30	N	
Channel Source	ChannelSource	DTYPE_TEXT	30	N	
Channel System	ChannelSystem	DTYPE_TEXT	30	N	
Cover Letter Type	CoverLetterType	DTYPE_TEXT	30	N	
Cover Letter Type II	CoverLetterTypeII	DTYPE_TEXT	30	N	
Financial Account Id	FinancialAccountId	DTYPE_ID		N	
Id	Id	DTYPE_ID	30	N	
Integration Id	IntegrationId	DTYPE_TEXT	30	N	
Nonaffiliate Flag	NonaffiliateFlag	DTYPE_TEXT	30	N	
Nonaffiliate Method	NonaffiliateMethod	DTYPE_TEXT	30	N	

Table 70. FINCORP Account Integration Component and Integration Component Fields

Name	XML Tag	Data Type	Length	Req	Notes
Nonaffiliate Source	NonaffiliateSource	DTYPE_TEXT	30	N	
Nonaffiliate System	NonaffiliateSystem	DTYPE_TEXT	30	N	
Notice Send Flag	NoticeSendFlag	DTYPE_TEXT	30	N	
Notice Sent Date	NoticeSentDate	DTYPE_UTCDATETIME		N	
Notice Sent Reason	NoticeSentReason	DTYPE_TEXT	100	N	
Notice Sent Type	NoticeSentType	DTYPE_TEXT	30	N	
OFAC Flag	OFACFlag	DTYPE_BOOL		N	
operation	operation	DTYPE_TEXT	30	N	
PEND Start Date	PENDStartDate	DTYPE_UTCDATETIME		N	
Privacy Parent Type	PrivacyParentType	DTYPE_TEXT	30	Y	
Privacy State Code	PrivacyStateCode	DTYPE_TEXT	30	N	
Receive Annual Notice Flag	ReceiveAnnualNoticeFlag	DTYPE_TEXT	30	N	
Receive Annual Notice Method	ReceiveAnnualNoticeMethod	DTYPE_TEXT	30	N	
Receive Annual Notice Source	ReceiveAnnualNoticeSource	DTYPE_TEXT	30	N	
Receive Annual Notice System	ReceiveAnnualNoticeSystem	DTYPE_TEXT	30	N	
searchspec	searchspec	DTYPE_TEXT	250	N	
State PEND Period	StatePENDPeriod	DTYPE_INTEGER		N	
Telemarketing Flag	TelemarketingFlag	DTYPE_TEXT	30	N	
Telemarketing Method	TelemarketingMethod	DTYPE_TEXT	30	N	

Table 70. FINCORP Account Integration Component and Integration Component Fields

Name	XML Tag	Data Type	Length	Req	Notes
Telemarketing Source	TelemarketingSource	DTYPE_TEXT	30	N	
Telemarketing System	TelemarketingSystem	DTYPE_TEXT	30	N	
Updated	Updated	DTYPE_DATETIME	30	N	
Updated By	UpdatedBy	DTYPE_ID	30	N	

Account Profile

UCM Account Profile messages are used in insert, query, update, and delete operations to maintain Account Profile Information. LookUpAccount Profile and LookUptProfileByName are used for query operations. UpdateAccountProfile, InsertAccountProfile, and DeleteAccountProfile are message pairs for request and response operations. Figure 21 describes the structure of this message.

Figure 21. UCM Account Profile Structure

Table 71 lists the Account Profile Query Integration objects.

Table 71. Account Profile Query Integration Objects

Integration Object	Business Component	XML Tag	User Keys
CIFAccountProfile	Account	LookUpAccountProfile	UK1:PartyUId UK2: Id
CIFAccountProfileRs	Account	LookUpAccountProfileResponse	UK1:PartyUId UK2: Id

Table 72 lists the Account Profile Query by Name Integration objects.

Table 72. Account Profile Query by Name Integration Objects

Integration Object	Business Component	XML Tag	User Keys
CIFAccountProfileByName	Account	LookUpAccountProfileByName	UK1:PartyUId UK2: Id
CIFAccountProfileByNameRs	Account	LookUpAccountProfileByNameResponse	UK1:PartyUId UK2: Id

Table 73 lists the Account Profile Update Integration objects.

Table 73. Account Profile Update Integration Objects

Integration Object	Business Component	XML Tag	User Keys
CIFAccountProfileUpdate	Account	UpdateAccountProfile	UK1:PartyUId UK2: Id
CIFAccountProfileUpdateRs	Account	UpdateAccountProfileResponse	UK1:PartyUId UK2: Id

Table 74 lists the Account Profile Insert Integration objects.

Table 74. Account Profile Insert Integration Objects

Integration Object	Business Component	XML Tag	User Keys
CIFAccountProfileInsert	Account	InsertAccountProfile	UK1:PartyUId UK2: Id
CIFAccountProfileInsertRs	Account	InsertAccountProfileResponse	UK1:PartyUId UK2: Id

Table 75 lists the Account Profile Delete Integration objects.

Table 75. Account Profile Delete Integration Objects

Integration Object	Business Component	XML Tag	User Keys
CIFAccountProfileInsert	Account	DeleteAccountProfile	UK1:PartyUId UK2: Id
CIFAccountProfileInsertRs	Account	DeleteAccountProfileResponse	UK1:PartyUId UK2: Id

Table 76 lists the Account Profile Integration Component and Integration Component Fields.

Table 76. Account Profile Integration Component and Integration Component Fields

Name	XML Tag	Data Type	Length	Req	Notes
Account Region	AccountRegion	DTYPE_TEXT	30		Use for picking a related entity
Account Status	AccountStatus	DTYPE_TEXT	24		Bound by ACCOUNT_STATUS
Account Type Code	AccountTypeCode	DTYPE_TEXT	30	✔	Bound by CUT_ACCOUNT_TYPE
Account Value	AccountValue	DTYPE_TEXT	30		Bound by FIN_CON_VALUE_TYPE
Alias	Alias	DTYPE_TEXT	50		
Annual Revenue	AnnualRevenue	DTYPE_TEXT	22		
Client Flag	ClientFlag	DTYPE_BOOL	1		
Competitor	Competitor	DTYPE_BOOL	1		
Credit Agency Number	CreditAgencyNumber	DTYPE_TEXT	30		
Cumulative Sales	CumulativeSales	DTYPE_NUMBER	22		
Currency Code	CurrencyCode	DTYPE_TEXT	15	✔	
Current Volume	CurrentVolume	DTYPE_CURRENCY	22		
Dont Call	DontCall	DTYPE_BOOL	1		

Table 76. Account Profile Integration Component and Integration Component Fields

Name	XML Tag	Data Type	Length	Req	Notes
Drug Spent Currency Code	DrugSpentCurrencyCode	DTYPE_TEXT	30	✔	Use for picking a related entity
Employees	Employees	DTYPE_NUMBER	22		
Facility Flag	FacilityFlag	DTYPE_BOOL	1		
Federal Tax Number	FederalTaxNumber	DTYPE_TEXT	15		
Financing Method	FinancingMethod	DTYPE_TEXT	30		Bound by FINCORP_CLIENT_FINANCING
Gross Potential Value	GrossPotentialValue	DTYPE_CURRENCY	22		
Group Type Code	GroupTypeCode	DTYPE_TEXT	30	✔	Bound by GROUP_TYPE_CD
Home Page	HomePage	DTYPE_TEXT	100		
Id	Id	DTYPE_ID	15		Use for cross-referencing
Integration Id	IntegrationId	DTYPE_TEXT	30		
Internal Org Flag	InternalOrgFlag	DTYPE_BOOL	1	✔	
Investor Flag	InvestorFlag	DTYPE_BOOL	1		
Last Year Sales	LastYearSales	DTYPE_CURRENCY	22		
Legal Status	LegalStatus	DTYPE_TEXT	30		Bound by UT_LEGAL_STATUS
Location	Location	DTYPE_TEXT	50		
Main Fax Number	MainFaxNumber	DTYPE_PHONE	40		
Main Phone Number	MainPhoneNumber	DTYPE_PHONE	40		
Master Account Id	MasterAccountId	DTYPE_ID	15		
Name	Name	DTYPE_TEXT	100	✔	
operation	operation	DTYPE_TEXT	30		
Partner Flag	PartnerFlag	DTYPE_BOOL	1		
Party UId	PartyUId	DTYPE_TEXT	100	✔	

Table 76. Account Profile Integration Component and Integration Component Fields

Name	XML Tag	Data Type	Length	Req	Notes
Privacy Code	PrivacyCode	DTYPE_TEXT	30	✔	Bound by PRIVACY_CODE
Reference Flag	ReferenceFlag	DTYPE_BOOL	1		
Referenciable	Referenciable	DTYPE_BOOL	1		
Revenue Growth	RevenueGrowth	DTYPE_NUMBER	22		
Service Provider Flag	ServiceProviderFlag	DTYPE_TEXT	1		
Status	Status	DTYPE_TEXT	30		Bound by FINS_COMPETITOR _STATUS_MLOV
Total # Meters	TotalMeters	DTYPE_NUMBER	22		
Total Potential Volume	TotalPotentialVolume	DTYPE_CURRENCY	22		
Type	Type	DTYPE_TEXT	30	✔	Bound by ACCOUNT_TYPE
Unionized	Unionized	DTYPE_BOOL	1		
YTD Net Contribution	YTDNetContribution	DTYPE_CURRENCY	22		

Account Address

UCM Account Address messages are used in insert, query, update, and delete operations to maintain Account Address Information. LookUpAccount Address and LookUpAccountAddressByName are used for query operations. UpdateAccountAddress, InsertAccountAddress, and DeleteAccountAddress are message pairs for request and response operations. Figure 22 describes the structure of this message.

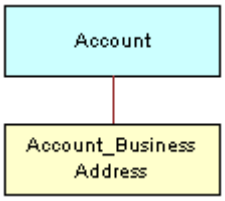

Figure 22. UCM Account Address Structure

Table 77 lists the Account Address Query Integration objects.

Table 77. Account Address Query Integration Objects

Integration Object	Business Component	XML Tag	User Keys
CIFAccountAddress	Account	LookUpAccountAddress	UK1:PartyUId UK2: Id
CIFAccountAddressRs	Account	LookUpAccountAddressResponse	UK1:PartyUId UK2: Id

Table 78 lists the Account Address Query by Name Integration objects.

Table 78. Account Address Query by Name Integration Objects

Integration Object	Business Component	XML Tag	User Keys
CIFAccountAddressByName	Account	LookUpAccountAddressByName	UK1:PartyUId UK2: Id
CIFAccountAddressByNameRs	Account	LookUpAccountAddressByNameResponse	UK1:PartyUId UK2: Id

Table 79 lists the Account Address Update Integration objects.

Table 79. Account Address Update Integration Objects

Integration Object	Business Component	XML Tag	User Keys
CIFAccountAddressUpdate	Account	UpdateAccountAddress	UK1:PartyUId UK2: Id
CIFAccountAddressUpdateRs	Account	UpdateAccountAddressResponse	UK1:PartyUId UK2: Id

Table 80 lists the Account Address Insert Integration objects.

Table 80. Account Address Insert Integration Objects

Integration Object	Business Component	XML Tag	User Keys
CIFAccountAddressInsert	Account	InsertAccountAddress	UK1:PartyUId UK2: Id
CIFAccountAddressInsertRs	Account	InsertAccountAddressResponse	UK1:PartyUId UK2: Id

Table 81 lists the Account Address Delete Integration objects.

Table 81. Account Address Delete Integration Objects

Integration Object	Business Component	XML Tag	User Keys
CIFAccountAddressDelete	Account	DeleteAccountAddress	UK1:PartyUId UK2: Id
CIFAccountAddressDeleteRs	Account	DeleteAccountAddressResponse	UK1:PartyUId UK2: Id

Table 82 lists the Account Address Integration Component and Integration Component Fields.

Table 82. Account Address Integration Component and Integration Component Fields

Name	XML Tag	Data Type	Length	Req	Notes
IC: Account					
Account Region	AccountRegion	DTYPE_TEXT	30		Use for picking a related entity
Account Status	AccountStatus	DTYPE_TEXT	24		Bound by ACCOUNT_STATUS
Account Type Code	AccountTypeCode	DTYPE_TEXT	30	✔	Bound by CUT_ACCOUNT_TYPE
Account Value	AccountValue	DTYPE_TEXT	30		Bound by FIN_CON_VALUE_TYPE
Alias	Alias	DTYPE_TEXT	50		

Table 82. Account Address Integration Component and Integration Component Fields

Name	XML Tag	Data Type	Length	Req	Notes
Annual Revenue	AnnualRevenue	DTYPE_TEXT	22		
Client Flag	ClientFlag	DTYPE_BOOL	1		
Competitor	Competitor	DTYPE_BOOL	1		
Credit Agency Number	CreditAgencyNumber	DTYPE_TEXT	30		
Cumulative Sales	CumulativeSales	DTYPE_NUMBER	22		
Currency Code	CurrencyCode	DTYPE_TEXT	15	✔	
Current Volume	CurrentVolume	DTYPE_CURRENCY	22		
Dont Call	DontCall	DTYPE_BOOL	1		
Drug Spent Currency Code	DrugSpentCurrencyCode	DTYPE_TEXT	30	✔	Use for picking a related entity
Employees	Employees	DTYPE_NUMBER	22		
Facility Flag	FacilityFlag	DTYPE_BOOL	1		
Federal Tax Number	FederalTaxNumber	DTYPE_TEXT	15		
Financing Method	FinancingMethod	DTYPE_TEXT	30		Bound by FINCORP_CLIENT_FINANCING
Gross Potential Value	GrossPotentialValue	DTYPE_CURRENCY	22		
Group Type Code	GroupTypeCode	DTYPE_TEXT	30	✔	Bound by GROUP_TYPE_CD
Home Page	HomePage	DTYPE_TEXT	100		
Integration Id	IntegrationId	DTYPE_TEXT	30		
Internal Org Flag	InternalOrgFlag	DTYPE_BOOL	1	✔	
Investor Flag	InvestorFlag	DTYPE_BOOL	1		
Last Year Sales	LastYearSales	DTYPE_CURRENCY	22		
Legal Status	LegalStatus	DTYPE_TEXT	30		Bound by UT_LEGAL_STATUS

Table 82. Account Address Integration Component and Integration Component Fields

Name	XML Tag	Data Type	Length	Req	Notes
Location	Location	DTYPE_TEXT	50		
Main Fax Number	MainFaxNumber	DTYPE_PHONE	40		
Main Phone Number	MainPhoneNumber	DTYPE_PHONE	40		
Master Account Id	MasterAccountId	DTYPE_ID	15		
Name	Name	DTYPE_TEXT	100	✔	
Partner Flag	PartnerFlag	DTYPE_BOOL	1		
Party UId	PartyUId	DTYPE_TEXT	100	✔	
Privacy Code	PrivacyCode	DTYPE_TEXT	30	✔	Bound by PRIVACY_CODE
Reference Flag	ReferenceFlag	DTYPE_BOOL	1		
Referenciable	Referenciable	DTYPE_BOOL	1		
Revenue Growth	RevenueGrowth	DTYPE_NUMBER	22		
Service Provider Flag	ServiceProviderFlag	DTYPE_TEXT	1		
Status	Status	DTYPE_TEXT	30		Bound by FINS_COMPETITOR_STATUS_MLOV
Total # Meters	TotalMeters	DTYPE_NUMBER	22		
Total Potential Volume	TotalPotentialVolume	DTYPE_CURRENCY	22		
Type	Type	DTYPE_TEXT	30	✔	Bound by ACCOUNT_TYPE
Unionized	Unionized	DTYPE_BOOL	1		
YTD Net Contribution	YTDNetContribution	DTYPE_CURRENCY	22		
operation	operation	DTYPE_TEXT	30		
Id	Id	DTYPE_ID	15		For cross-referencing
IC: Account_Business Address					
Address Id	AddressId	DTYPE_ID	30	✔	

Table 82. Account Address Integration Component and Integration Component Fields

Name	XML Tag	Data Type	Length	Req	Notes
Address Integration Id	AddressIntegrationId	DTYPE_TEXT	30		
Bill Address Flag	BillAddressFlag	DTYPE_TEXT	1		
City	City	DTYPE_TEXT	50		
Country	Country	DTYPE_TEXT	50		Use for pick a related entity
Email Address	EmailAddress	DTYPE_TEXT	50		
IsPrimaryMVG	IsPrimaryMVG	DTYPE_TEXT	1		
operation	operation	DTYPE_TEXT	30		
Postal Code	PostalCode	DTYPE_TEXT	30		
searchspec	searchspec	DTYPE_TEXT	250		
Ship Address Flag	ShipAddressFlag	DTYPE_TEXT	1		
State	State	DTYPE_TEXT	10		Bound by STATE_ABBREV
Street Address	StreetAddress	DTYPE_TEXT	200		

Account Activity

UCM Account Activity messages are used in insert, query, update, and delete operations to maintain Account Activity Information. LookUpAccountActivity and LookUpAccountActivityByName are used for query operations. UpdateAccountActivity, InsertAccountActivity, and DeleteAccountActivity are message pairs for request and response operations. Figure 23 describes the structure of this message.

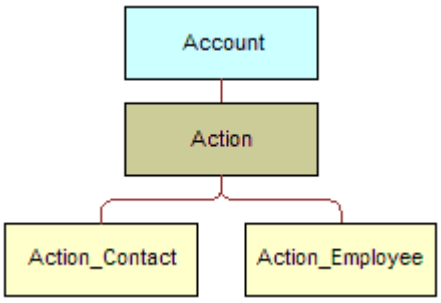

Figure 23. UCM Account Activity Structure

Table 83 lists the Account Activity Query Integration objects.

Table 83. Account Activity Query Integration Objects

Integration Object	Business Component	XML Tag	User Keys
CIFAccountActivity	Account	LookUpAccountActivity	UK1:PartyUId UK2: Id
CIFAccountActivityRs	Account	LookUpAccountActivityResponse	UK1:PartyUId UK2: Id

Table 84 lists the Account Activity Query by Name Integration objects.

Table 84. Account Activity Query by Name Integration Objects

Integration Object	Business Component	XML Tag	User Keys
CIFAccountActivityByName	Account	LookUpAccountActivityByNames	UK1:PartyUId UK2: Id
CIFAccountActivityByNameRs	Account	LookUpAccountActivityByName Response	UK1:PartyUId UK2: Id

Table 85 lists the Account Activity Update Integration objects.

Table 85. Account Activity Update Integration Objects

Integration Object	Business Component	XML Tag	User Keys
CIFAccountActivityUpdate	Account	UpdateAccountActivitys	UK1:PartyUId UK2: Id
CIFAccountActivityUpdateRs	Account	UpdateAccountActivityResponse	UK1:PartyUId UK2: Id

Table 86 lists the Account Activity Insert Integration objects.

Table 86. Account Activity Insert Integration Objects

Integration Object	Business Component	XML Tag	User Keys
CIFAccountActivityInsert	Account	InsertAccountActivity	UK1:PartyUId UK2: Id
CIFAccountActivityInsertRs	Account	InsertAccountActivityResponse	UK1:PartyUId UK2: Id

Table 87 lists the Account Activity Delete Integration objects.

Table 87. Account Activity Delete Integration Objects

Integration Object	Business Component	XML Tag	User Keys
CIFAccountActivityDelete	Account	DeleteAccountActivity	UK1:PartyUId UK2: Id
CIFAccountActivityDeleteRs	Account	DeleteAccountActivityResponse	UK1:PartyUId UK2: Id

Table 88 lists the Account Activity Integration Component and Integration Component Fields.

Table 88. Account Activity Integration Component and Integration Component Fields

Name	XML Tag	Type	Length	Req	Notes
IC: Account					
Account Region	AccountRegion	DTYPE_TEXT	30		Use for picking a related entity
Account Status	AccountStatus	DTYPE_TEXT	24		Bound by ACCOUNT_STATUS
Account Type Code	AccountTypeCode	DTYPE_TEXT	30	✔	Bound by CUT_ACCOUNT_TYPE
Account Value	AccountValue	DTYPE_TEXT	30		Bound by FIN_CON_VALUE_TYPE
Alias	Alias	DTYPE_TEXT	50		

Table 88. Account Activity Integration Component and Integration Component Fields

Name	XML Tag	Type	Length	Req	Notes
Annual Revenue	AnnualRevenue	DTYPE_TEXT	22		
Client Flag	ClientFlag	DTYPE_BOOL	1		
Competitor	Competitor	DTYPE_BOOL	1		
Credit Agency Number	CreditAgencyNumber	DTYPE_TEXT	30		
Cumulative Sales	CumulativeSales	DTYPE_NUMBER	22		
Currency Code	CurrencyCode	DTYPE_TEXT	15	✔	
Current Volume	CurrentVolume	DTYPE_CURRENCY	22		
Dont Call	DontCall	DTYPE_BOOL	1		
Drug Spent Currency Code	DrugSpentCurrencyCode	DTYPE_TEXT	30	✔	Use for picking a related entity
Employees	Employees	DTYPE_NUMBER	22		
Facility Flag	FacilityFlag	DTYPE_BOOL	1		
Federal Tax Number	FederalTaxNumber	DTYPE_TEXT	15		
Financing Method	FinancingMethod	DTYPE_TEXT	30		Bound by FINCORP_CLIENT_FINANCING
Gross Potential Value	GrossPotentialValue	DTYPE_CURRENCY	22		
Group Type Code	GroupTypeCode	DTYPE_TEXT	30	✔	Bound by GROUP_TYPE_CD
Home Page	HomePage	DTYPE_TEXT	100		
Integration Id	IntegrationId	DTYPE_TEXT	30		
Internal Org Flag	InternalOrgFlag	DTYPE_BOOL	1	✔	
Investor Flag	InvestorFlag	DTYPE_BOOL	1		
Last Year Sales	LastYearSales	DTYPE_CURRENCY	22		
Legal Status	LegalStatus	DTYPE_TEXT	30		Bound by UT_LEGAL_STATUS

Table 88. Account Activity Integration Component and Integration Component Fields

Name	XML Tag	Type	Length	Req	Notes
Location	Location	DTYPE_TEXT	50		
Main Fax Number	MainFaxNumber	DTYPE_PHONE	40		
Main Phone Number	MainPhoneNumber	DTYPE_PHONE	40		
Master Account Id	MasterAccountId	DTYPE_ID	15		
Name	Name	DTYPE_TEXT	100	✔	
Partner Flag	PartnerFlag	DTYPE_BOOL	1		
Party UId	PartyUId	DTYPE_TEXT	100	✔	
Privacy Code	PrivacyCode	DTYPE_TEXT	30	✔	Bound by PRIVACY_CODE
Reference Flag	ReferenceFlag	DTYPE_BOOL	1		
Referenciable	Referenciable	DTYPE_BOOL	1		
Revenue Growth	RevenueGrowth	DTYPE_NUMBER	22		
Service Provider Flag	ServiceProviderFlag	DTYPE_TEXT	1		
Status	Status	DTYPE_TEXT	30		Bound by FINS_COMPETITOR_STATUS_MLOV
Total # Meters	TotalMeters	DTYPE_NUMBER	22		
Total Potential Volume	TotalPotentialVolume	DTYPE_CURRENCY	22		
Type	Type	DTYPE_TEXT	30	✔	Bound by ACCOUNT_TYPE
Unionized	Unionized	DTYPE_BOOL	1		
YTD Net Contribution	YTDNetContribution	DTYPE_CURRENCY	22		
operation	operation	DTYPE_TEXT	30		
Id	Id	DTYPE_ID	15		For cross-referencing
IC: Action					
Alarm	Alarm	DTYPE_BOOL	1		

Table 88. Account Activity Integration Component and Integration Component Fields

Name	XML Tag	Type	Length	Req	Notes
Assignment Excluded	AssignmentExcluded	DTYPE_BOOL	1		
Associated Cost	AssociatedCost	DTYPE_CURRENCY	22		
Billable Flag	BillableFlag	DTYPE_BOOL	1		
Call Id	CallId	DTYPE_TEXT	30		
Category	Category	DTYPE_TEXT	30	✔	Use for picking a related entity
Class	Class	DTYPE_TEXT	30		Bound by FS_ACTIVITY_CLASS
Comment	Comment	DTYPE_TEXT	“1,500”		
Description	Description	DTYPE_TEXT	100		
Display	Display	DTYPE_TEXT	30		
Done	Done	DTYPE_UTCDATETIME	7		
Done Flag	DoneFlag	DTYPE_BOOL	1		
Due	Due	DTYPE_UTCDATETIME	7		
Duration Minutes	DurationMinutes	DTYPE_INTEGER	22		Bound by APPOINTMENT_DURATION
Expense Related Flag	ExpenseRelatedFlag	DTYPE_BOOL	1		
Id	Id	DTYPE_TEXT	100	✔	
Integration Id	IntegrationId	DTYPE_TEXT	30		
No Sooner Than Date	NoSoonerThanDate	DTYPE_UTCDATETIME	7		
Objective	Objective	DTYPE_TEXT	30		
operation	operation	DTYPE_TEXT	30		
Owned By	OwnedBy	DTYPE_TEXT	30	✔	Use for pick a related entity
Owned By Id	OwnedById	DTYPE_TEXT	30	✔	
Primary Owned By	PrimaryOwnedBy	DTYPE_TEXT	50		Use for pick a related entity

Table 88. Account Activity Integration Component and Integration Component Fields

Name	XML Tag	Type	Length	Req	Notes
Primary Owner Id	PrimaryOwnerId	DTYPE_ID	30		
Priority	Priority	DTYPE_TEXT	30		Bound by ACTIVITY_PRIORITY
Private	Private	DTYPE_BOOL	1		
Repeating	Repeating	DTYPE_BOOL	1		
Repeating Expires	RepeatingExpires	DTYPE_DATETIME	7		
searchspec	searchspec	DTYPE_TEXT	250		
Status	Status	DTYPE_TEXT	30		Bound by EVENT_STATUS
Type	Type2	DTYPE_TEXT	30	✔	Bound by TODO_TYPE
IC: Action_Contact					
Company Identifier	CompanyIdentifier	DTYPE_TEXT	15		
Contact First Name	ContactFirstName	DTYPE_TEXT	50	✔	
Contact Last Name	ContactLastName	DTYPE_TEXT	50	✔	
Employee Number	EmployeeNumber	DTYPE_TEXT	30		
IsPrimaryMVG	IsPrimaryMVG	DTYPE_TEXT	1		
operation	operation	DTYPE_TEXT	30		
Personal Contact	PersonalContact	DTYPE_BOOL	1		
searchspec	searchspec	DTYPE_TEXT	250		
IC: Action_Employee					
Emp #	Emp	DTYPE_TEXT	30		
Employee First Name (MM)	EmployeeFirstNameMM	DTYPE_TEXT	50	✔	
Employee Job Title (MM)	EmployeeJobTitleMM	DTYPE_TEXT	75		

Table 88. Account Activity Integration Component and Integration Component Fields

Name	XML Tag	Type	Length	Req	Notes
Employee Last Name (MM)	EmployeeLastNameMM	DTYPE_TEXT	50	✔	
Employee Login Name (MM)	EmployeeLoginNameMM	DTYPE_TEXT	50	✔	
IsPrimaryMVG	IsPrimaryMVG	DTYPE_TEXT	1		
operation	operation	DTYPE_TEXT	30		
Private	Private	DTYPE_BOOL	1	✔	
searchspec	searchspec	DTYPE_TEXT	250		

Account Product

UCM Account Products messages are used in insert, query, update, and delete operations to maintain Account Product (Financial Account) Information. LookUpAccountProduct and LookUpAccountProductByName are used for query operations. UpdateAccountProduct, InsertAccountProduct, and DeleteAccountProduct are message pairs for request and response operations. Figure 24 describes the structure of this message.

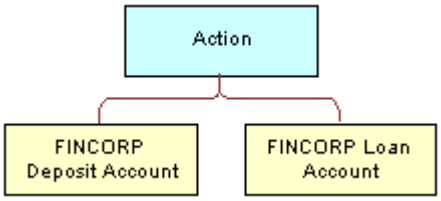

Figure 24. UCM Account Products Structure

Table 89 lists the Account Product Query Integration objects.

Table 89. Account Product Query Integration Objects

Integration Object	Business Component	XML Tag	User Keys
CIFAccountProduct	Account	LookUpAccountProduct	UK1:PartyUId UK2: Id
CIFAccountProductRs	Account	LookUpAccountProductResponse	UK1:PartyUId UK2: Id

Table 90 lists the Account Product Query by Name Integration objects.

Table 90. Account Product Query by Name Integration Objects

Integration Object	Business Component	XML Tag	User Keys
CIFAccountProductByName	Account	LookUpAccountProductByName	UK1:PartyUId UK2: Id
CIFAccountProductByNameRs	Account	LookUpAccountProductByNameResponse	UK1:PartyUId UK2: Id

Table 91 lists the Account Product Update Integration objects.

Table 91. Account Product Update Integration Objects

Integration Object	Business Component	XML Tag	User Keys
CIFAccountProductUpdate	Account	UpdateAccountProduct	UK1:PartyUId UK2: Id
CIFAccountProductUpdateRs	Account	UpdateAccountProductResponse	UK1:PartyUId UK2: Id

Table 92 lists the Account Product Insert Integration objects.

Table 92. Account Product Insert Integration Objects

Integration Object	Business Component	XML Tag	User Keys
CIFAccountProductInsert	Account	InsertAccountProduct	UK1:PartyUId UK2: Id
CIFAccountProductInsertRs	Account	InsertAccountProductResponse	UK1:PartyUId UK2: Id

Table 93 lists the Account Product Delete Integration objects.

Table 93. Account Product Delete Integration Objects

Integration Object	Business Component	XML Tag	User Keys
CIFAccountProductDelete	Account	DeleteAccountProduct	UK1:PartyUId UK2: Id
CIFAccountProductDeleteRs	Account	DeleteAccountProductResponse	UK1:PartyUId UK2: Id

Table 94 lists the Account Product Integration Component and Integration Component Fields.

Table 94. Account Product Integration Component and Integration Component Fields

Name	XML Tag	Data Type	Length	Req	Notes
IC: Account					
Account Region	AccountRegion	DTYPE_TEXT	30		Use for picking a related entity
Account Status	AccountStatus	DTYPE_TEXT	24		Bound by ACCOUNT_STATUS
Account Type Code	AccountTypeCode	DTYPE_TEXT	30	✔	Bound by CUT_ACCOUNT_TYPE
Account Value	AccountValue	DTYPE_TEXT	30		Bound by FIN_CON_VALUE_TYPE
Alias	Alias	DTYPE_TEXT	50		
Annual Revenue	AnnualRevenue	DTYPE_TEXT	22		
Client Flag	ClientFlag	DTYPE_BOOL	1		
Competitor	Competitor	DTYPE_BOOL	1		
Credit Agency Number	CreditAgencyNumber	DTYPE_TEXT	30		
Cumulative Sales	CumulativeSales	DTYPE_NUMBER	22		
Currency Code	CurrencyCode	DTYPE_TEXT	15	✔	
Current Volume	CurrentVolume	DTYPE_CURRENCY	22		

Table 94. Account Product Integration Component and Integration Component Fields

Name	XML Tag	Data Type	Length	Req	Notes
Dont Call	DontCall	DTYPE_BOOL	1		
Drug Spent Currency Code	DrugSpentCurrencyCode	DTYPE_TEXT	30	✔	Use for picking a related entity
Employees	Employees	DTYPE_NUMBER	22		
Facility Flag	FacilityFlag	DTYPE_BOOL	1		
Federal Tax Number	FederalTaxNumber	DTYPE_TEXT	15		
Financing Method	FinancingMethod	DTYPE_TEXT	30		Bound by FINCORP_CLIENT_FINANCING
Gross Potential Value	GrossPotentialValue	DTYPE_CURRENCY	22		
Group Type Code	GroupTypeCode	DTYPE_TEXT	30	✔	Bound by GROUP_TYPE_CD
Home Page	HomePage	DTYPE_TEXT	100		
Id	Id	DTYPE_ID	15		For cross-referencing
Integration Id	IntegrationId	DTYPE_TEXT	30		
Internal Org Flag	InternalOrgFlag	DTYPE_BOOL	1	✔	
Investor Flag	InvestorFlag	DTYPE_BOOL	1		
Last Year Sales	LastYearSales	DTYPE_CURRENCY	22		
Legal Status	LegalStatus	DTYPE_TEXT	30		Bound by UT_LEGAL_STATUS
Location	Location	DTYPE_TEXT	50		
Main Fax Number	MainFaxNumber	DTYPE_PHONE	40		
Main Phone Number	MainPhoneNumber	DTYPE_PHONE	40		
Master Account Id	MasterAccountId	DTYPE_ID	15		
Name	Name	DTYPE_TEXT	100		
operation	operation	DTYPE_TEXT	30		

Table 94. Account Product Integration Component and Integration Component Fields

Name	XML Tag	Data Type	Length	Req	Notes
Partner Flag	PartnerFlag	DTYPE_BOOL	1		
Party UId	PartyUId	DTYPE_TEXT	100	✔	
Privacy Code	PrivacyCode	DTYPE_TEXT	30	✔	Bound by PRIVACY_CODE
Reference Flag	ReferenceFlag	DTYPE_BOOL	1		
Referenciable	Referenciable	DTYPE_BOOL	1		
Revenue Growth	RevenueGrowth	DTYPE_NUMBER	22		
Service Provider Flag	ServiceProviderFlag	DTYPE_TEXT	1		
Status	Status	DTYPE_TEXT	30		Bound by FINS_COMPETITOR_STATUS_MLOV
Total # Meters	TotalMeters	DTYPE_NUMBER	22		
Total Potential Volume	TotalPotentialVolume	DTYPE_CURRENCY	22		
Type	Type	DTYPE_TEXT	30		Bound by ACCOUNT_TYPE
Unionized	Unionized	DTYPE_BOOL	1		
YTD Net Contribution	YTDNetContribution	DTYPE_CURRENCY	22		
IC: FINCORP Deposit Account					
Account Branch Number	AccountBranchNumber	DTYPE_TEXT	100		
Account Category	AccountCategory	DTYPE_TEXT	30		Bound by FINS_ASSET_PROD_CLASS_MLOV
Account Number	AccountNumber	DTYPE_TEXT	100	✔	
Account Status	AccountStatus	DTYPE_TEXT	30		Bound by FINS_CRPST_LFLN_IPFS_PLC_MLOV
Branch	Branch	DTYPE_TEXT	50		

Table 94. Account Product Integration Component and Integration Component Fields

Name	XML Tag	Data Type	Length	Req	Notes
Current Balance	CurrentBalance	DTYPE_CURRENCY	22		
Id	Id	DTYPE_TEXT	100		
Integration Id	Integration Id	DTYPE_TEXT	30		
Name	Name	DTYPE_TEXT	100	✔	Use for pick a related entity
operation	operation	DTYPE_TEXT	30		
Product Id	ProductId	DTYPE_ID	100	✔	
searchspec	searchspec	DTYPE_TEXT	250		
Type	Type	DTYPE_TEXT	30		Bound by FINS_OWNERSHIP_TYPE_MLOV
IC: FINCORP Loan Account					
Account Branch Number	AccountBranchNumber	DTYPE_TEXT	30		
Account Category	AccountCategory	DTYPE_TEXT	30		Bound by FINS_ASSET_PROD_CLASS_MLOV
Account Number	AccountNumber	DTYPE_TEXT	100	✔	
Account Status	AccountStatus	DTYPE_TEXT	30		Bound by FINS_CRPST_LFLN_IPFS_PLC_MLOV
Current Balance	CurrentBalance	DTYPE_CURRENCY	22		
Id	Id	DTYPE_TEXT	30		
Integration Id	IntegrationId	DTYPE_TEXT	30		
Name	Name	DTYPE_TEXT	100	✔	Use for pick a related entity
operation	operation	DTYPE_TEXT	30		
Product Id	ProductId	DTYPE_ID	100	✔	Use for pick a related entity
searchspec	searchspec	DTYPE_TEXT	250		
Type	Type	DTYPE_TEXT	30		FINS_OWNERSHIP_TYPE_MLOV

Account Relationship

UCM Account Relationship messages are used in insert, query, update, and delete operations to maintain Account Relationships. LookUpAccountRelationship and LookUpAccountRelationshipByName are used for query operations. UpdateAccountRelationship, InsertAccountRelationship, and DeleteAccountRelationship are message pairs for request and response operations. Figure 25 describes the structure of this message.

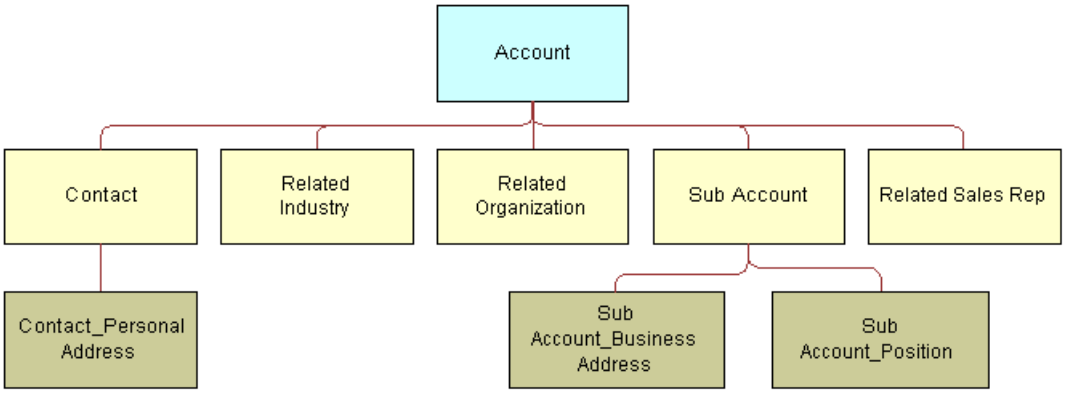

Figure 25. UCM Account Relationship Structure

Table 95 lists the Account Relationship Query Integration objects.

Table 95. Account Relationship Query Integration Objects

Integration Object	Business Component	XML Tag	User Keys
CIFAccountRelationship	Account	LookUpAccountRelationship	UK1:PartyUId UK2: Id
CIFAccountRelationshipRs	Account	LookUpAccountRelationshipResponse	UK1:PartyUId UK2: Id

Table 96 lists the Account Relationship Query by Name Integration objects.

Table 96. Account Relationship Query by Name Integration Objects

Integration Object	Business Component	XML Tag	User Keys
CIFAccountRelationshipByName	Account	LookUpAccountRelationshipByName	UK1:PartyUId UK2: Id
CIFAccountRelationshipByNameRs	Account	LookUpAccountRelationshipByName Response	UK1:PartyUId UK2: Id

Table 97 lists the Account Relationship Update Integration objects.

Table 97. Account Relationship Update Integration Objects

Integration Object	Business Component	XML Tag	User Keys
CIFAccountRelationshipUpdate	Account	UpdateAccountRelationship	UK1:PartyUId UK2: Id
CIFAccountRelationshipUpdateRs	Account	UpdateAccountRelationshipResponse	UK1:PartyUId UK2: Id

Table 98 lists the Account Relationship Insert Integration objects.

Table 98. Account Relationship Insert Integration Objects

Integration Object	Business Component	XML Tag	User Keys
CIFAccountRelationshipInsert	Account	InsertAccountRelationship	UK1:PartyUId UK2: Id
CIFAccountRelationshipInsertRs	Account	InsertAccountRelationshipResponse	UK1:PartyUId UK2: Id

Table 99 lists the Account Relationship Delete Integration objects.

Table 99. Account Relationship Delete Integration Objects

Integration Object	Business Component	XML Tag	User Keys
CIFAccountRelationshipDelete	Account	DeleteAccountRelationship	UK1:PartyUId UK2: Id
CIFAccountRelationshipDeleteRs	Account	DeleteAccountRelationshipResponse	UK1:PartyUId UK2: Id

Table 100 lists the Account Relationship Integration Component and Integration Component Fields.

Table 100. Account Relationship Integration Component and Integration Component Fields

Name	XML Tag	Data Type	Length	Req	Notes
IC: Account					
Account Region	AccountRegion	DTYPE_TEXT	30		Use for picking a related entity
Account Status	AccountStatus	DTYPE_TEXT	24		Bound by ACCOUNT_STATUS
Account Type Code	AccountTypeCode	DTYPE_TEXT	30	✔	Bound by CUT_ACCOUNT_TYPE
Account Value	AccountValue	DTYPE_TEXT	30		Bound by FIN_CON_VALUE_TYPE
Alias	Alias	DTYPE_TEXT	50		
Annual Revenue	AnnualRevenue	DTYPE_TEXT	22		
Client Flag	ClientFlag	DTYPE_BOOL	1		
Competitor	Competitor	DTYPE_BOOL	1		
Credit Agency Number	CreditAgencyNumber	DTYPE_TEXT	30		
Cumulative Sales	CumulativeSales	DTYPE_NUMBER	22		
Currency Code	CurrencyCode	DTYPE_TEXT	15	✔	
Current Volume	CurrentVolume	DTYPE_CURRENCY	22		
Dont Call	DontCall	DTYPE_BOOL	1		
Drug Spent Currency Code	DrugSpentCurrencyCode	DTYPE_TEXT	30	✔	Use for picking a related entity
Employees	Employees	DTYPE_NUMBER	22		
Facility Flag	FacilityFlag	DTYPE_BOOL	1		
Federal Tax Number	FederalTaxNumber	DTYPE_TEXT	15		
Financing Method	FinancingMethod	DTYPE_TEXT	30		Bound by FINCORP_CLIENT_FINANCING

Table 100. Account Relationship Integration Component and Integration Component Fields

Name	XML Tag	Data Type	Length	Req	Notes
Gross Potential Value	GrossPotentialValue	DTYPE_CURRENCY	22		
Group Type Code	GroupTypeCode	DTYPE_TEXT	30	✔	Bound by GROUP_TYPE_CD
Home Page	HomePage	DTYPE_TEXT	100		
Id	Id	DTYPE_ID	15		For cross-referencing
Integration Id	IntegrationId	DTYPE_TEXT	30		
Internal Org Flag	InternalOrgFlag	DTYPE_BOOL	1	✔	
Investor Flag	InvestorFlag	DTYPE_BOOL	1		
Last Year Sales	LastYearSales	DTYPE_CURRENCY	22		
Legal Status	LegalStatus	DTYPE_TEXT	30		Bound by UT_LEGAL_STATUS
Location	Location	DTYPE_TEXT	50		
Main Fax Number	MainFaxNumber	DTYPE_PHONE	40		
Main Phone Number	MainPhoneNumber	DTYPE_PHONE	40		
Master Account Id	MasterAccountId	DTYPE_ID	15		
Name	Name	DTYPE_TEXT	100	✔	
operation	operation	DTYPE_TEXT	30		
Partner Flag	PartnerFlag	DTYPE_BOOL	1		
Party UId	PartyUId	DTYPE_TEXT	100	✔	
Privacy Code	PrivacyCode	DTYPE_TEXT	30	✔	Bound by PRIVACY_CODE
Reference Flag	ReferenceFlag	DTYPE_BOOL	1		
Referenciable	Referenciable	DTYPE_BOOL	1		
Revenue Growth	RevenueGrowth	DTYPE_NUMBER	22		
Service Provider Flag	ServiceProviderFlag	DTYPE_TEXT	1		

Table 100. Account Relationship Integration Component and Integration Component Fields

Name	XML Tag	Data Type	Length	Req	Notes
Status	Status	DTYPE_TEXT	30		Bound by FINS_COMPETITOR_STATUS_MLOV
Total # Meters	TotalMeters	DTYPE_NUMBER	22		
Total Potential Volume	TotalPotentialVolume	DTYPE_CURRENCY	22		
Type	Type	DTYPE_TEXT	30		Bound by ACCOUNT_TYPE
Unionized	Unionized	DTYPE_BOOL	1		
YTD Net Contribution	YTDNetContribution	DTYPE_CURRENCY	22		
IC: Contact					
Account Id	AccountId	DTYPE_ID	30		
Account Integration Id	AccountIntegrationId	DTYPE_TEXT	30		
Assistant	Assistant	DTYPE_TEXT	50		
Assistant Phone #	AssistantPhone	DTYPE_PHONE	40		
Cellular Phone #	CellularPhone	DTYPE_PHONE	40		
Fax Phone #	FaxPhone	DTYPE_PHONE	40		
First Name	FirstName	DTYPE_TEXT	50	✔	
Home Phone #	HomePhone	DTYPE_TEXT	40		
Job Title	JobTitle	DTYPE_TEXT	75		
Last Name	LastName	DTYPE_TEXT	50	✔	
M/M	MM	DTYPE_TEXT	15		
Middle Name	MiddleName	DTYPE_TEXT	50		
operation	operation	DTYPE_TEXT	30		
Party UId	PartyUId	DTYPE_TEXT	100	✔	
Person UId	PersonUId	DTYPE_ID	30		
Preferred Communication Method	PreferredCommunicationMethod	DTYPE_TEXT	20		Bound by COMM_METHOD

Table 100. Account Relationship Integration Component and Integration Component Fields

Name	XML Tag	Data Type	Length	Req	Notes
Privacy Code	PrivacyCode	DTYPE_TEXT	30	✔	Bound by PRIVACY_CODE
Row Id	RowId	DTYPE_ID	100		
searchspec	searchspec	DTYPE_TEXT	250		
Work Phone #	WorkPhone	DTYPE_PHONE	45		
IC: Contact_Personal Address					
IsPrimaryMVG	IsPrimaryMVG	DTYPE_TEXT	1		
operation	operation	DTYPE_TEXT	30		
Personal Address Id	PersonalAddressId	DTYPE_ID	100	✔	
Personal Address Integration Id	PersonalAddressIntegrationId	DTYPE_TEXT	30		
Personal City	PersonalCity	DTYPE_TEXT	50	✔	
Personal Country	PersonalCountry	DTYPE_TEXT	50		Bound by COUNTRY
Personal Postal Code	PersonalPostalCode	DTYPE_TEXT	30		
Personal State	PersonalState	DTYPE_TEXT	10		Bound by STATE_ABBREV
Personal Street Address	PersonalStreetAddress	DTYPE_TEXT	200	✔	
searchspec	searchspec	DTYPE_TEXT	250		
IC: Related Industry					
Industry	Industry	DTYPE_TEXT	50	✔	
Industry Id	IndustryId	DTYPE_ID	30	✔	
Industry Integration Id	IndustryIntegrationId	DTYPE_TEXT	30		
IsPrimaryMVG	IsPrimaryMVG	DTYPE_TEXT	1		
SIC Code	SICCode	DTYPE_TEXT	30	✔	
IC: Related Organization					
IsPrimaryMVG	IsPrimaryMVG	DTYPE_TEXT	1		
Organization	Organization	DTYPE_TEXT	100	✔	
Organization Id	OrganizationId	DTYPE_ID	100	✔	

Table 100. Account Relationship Integration Component and Integration Component Fields

Name	XML Tag	Data Type	Length	Req	Notes
Organization Integration Id	OrganizationIntegrationId	DTYPE_TEXT	30		
IC: Related Sales Reps					
Division	Division	DTYPE_TEXT	255	✔	Use to pick a related entity
IsPrimaryMVG	IsPrimaryMVG	DTYPE_TEXT	1		
Login	Login	DTYPE_ID	15		
Position	Position	DTYPE_TEXT	50		
Position Id	PositionId	DTYPE_ID	30		
Position Integration Id	PositionIntegrationId	DTYPE_TEXT	30		
IC: Sub Account					
Account Status	AccountStatus2	DTYPE_TEXT	30		Bound by ACCOUNT_STATUS
Id	Id	DTYPE_ID	100	✔	
Location	Location	DTYPE_TEXT	50		
Main Phone Number	MainPhoneNumber	DTYPE_PHONE	40		
Name	Name	DTYPE_TEXT	100	✔	
operation	operation	DTYPE_TEXT	30		
operation	operation	DTYPE_TEXT	30		
Party UId	PartyUId	DTYPE_TEXT	30	✔	
Party UId	PartyUId	DTYPE_TEXT	30		
searchspec	searchspec	DTYPE_TEXT	250		
searchspec	searchspec	DTYPE_TEXT	250		
Type	Type	DTYPE_TEXT	255	✔	Bound by ACCOUNT_TYPE
Type	Type	DTYPE_TEXT	255		
IC: Sub Account_Business Address					
Address Id	AddressId	DTYPE_ID	30	✔	
Address Name	AddressName	DTYPE_TEXT	100		
Integration Id	IntegrationId	DTYPE_TEXT	30		
IsPrimaryMVG	IsPrimaryMVG	DTYPE_TEXT	1		

Table 100. Account Relationship Integration Component and Integration Component Fields

Name	XML Tag	Data Type	Length	Req	Notes
operation	operation	DTYPE_TEXT	30		
searchspec	searchspec	DTYPE_TEXT	250		
Street Address	StreetAddress	DTYPE_TEXT	200	✔	
IC: Sub Account_Position					
Division	Division	DTYPE_TEXT	255	✔	Use to pick the related entity
Division Integration Id	DivisionIntegrationId	DTYPE_TEXT	30		
IsPrimaryMVG	IsPrimaryMVG	DTYPE_TEXT	1		
operation	operation	DTYPE_TEXT	30		
Position	Position	DTYPE_TEXT	50		Bound by POSITION_TYPE
Position Id	PositionId	DTYPE_ID	30		
searchspec	searchspec	DTYPE_TEXT	250		

Account

UCM Account messages are used in insert, query, update, and delete operations to maintain Account Profile Information. LookUpAccount Profile and LookUptProfileByName are used for query operations. UpdateAccountProfile, InsertAccountProfile, and DeleteAccountProfile are message pairs for request and response operations. Figure 26 describes the structure of this message.

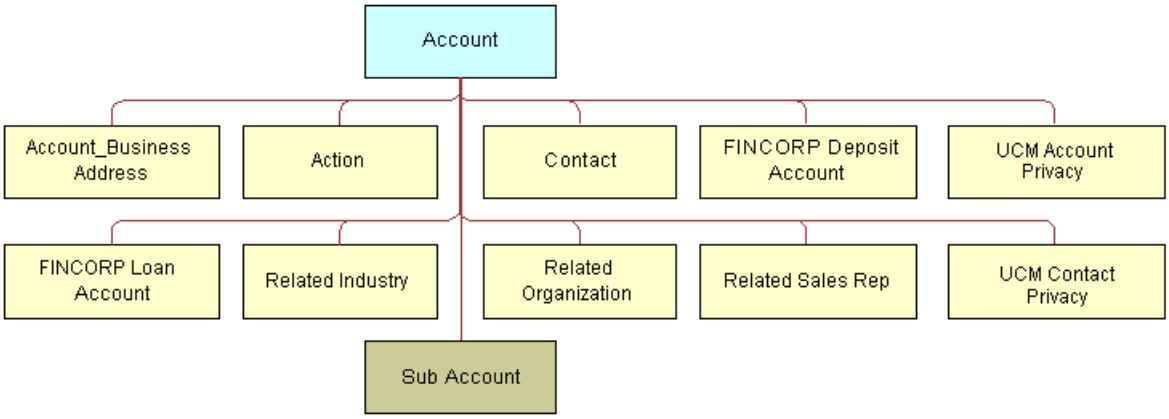

Figure 26. UCM Account Structure

Table 101 lists the Account Query Integration objects.

Table 101. Account Query Integration Objects

Integration Object	Business Component	XML Tag	User Keys
CIFAccount	Account	LookUpAccount	UK1:PartyUId UK2: Id
CIFAccountRs	Account	LookUpAccountResponse	UK1:PartyUId UK2: Id

Table 102 lists the Account Query by Name Integration objects.

Table 102. Account Query by Name Integration Objects

Integration Object	Business Component	XML Tag	User Keys
CIFAccountUpdate	Account	UpdateAccount	UK1:PartyUId UK2: Id
CIFAccountUpdateRs	Account	UpdateAccountResponse	UK1:PartyUId UK2: Id

Table 103 lists the Account Update Integration objects.

Table 103. Account Update Integration Objects

Integration Object	Business Component	XML Tag	User Keys
CIFAccountProductDelete	Account	UpdateAccountProduct	UK1:PartyUId UK2: Id
CIFAccountProductDeleteRs	Account	UpdateAccountProductResponse	UK1:PartyUId UK2: Id

Table 104 lists the Account Insert Integration objects.

Table 104. Account Insert Integration Objects

Integration Object	Business Component	XML Tag	User Keys
CIFAccountInsert	Account	InsertAccount	UK1:PartyUId UK2: Id
CIFAccountInsertRs	Account	InsertAccountResponse	UK1:PartyUId UK2: Id

Table 105 lists the Account Delete Integration objects.

Table 105. Account Delete Integration Objects

Integration Object	Business Component	XML Tag	User Keys
CIFAccountDelete	Account	DeleteAccount	UK1:PartyUId UK2: Id
CIFAccountDeleteRs	Account	DeleteAccountResponse	UK1:PartyUId UK2: Id

Table 106 lists the Account Integration Component and Integration Component Fields.

Table 106. Account Integration Component and Integration Component Fields

Name	XML Tag	Data Type	Length	Req	Notes
IC: Account					
Account Region	AccountRegion	DTYPE_TEXT	30		Use for picking a related entity
Account Status	AccountStatus	DTYPE_TEXT	24		Bound by ACCOUNT_STATUS
Account Type Code	AccountTypeCode	DTYPE_TEXT	30	✔	Bound by CUT_ACCOUNT_TYPE
Account Value	AccountValue	DTYPE_TEXT	30		Bound by FIN_CON_VALUE_TYPE
Alias	Alias	DTYPE_TEXT	50		

Table 106. Account Integration Component and Integration Component Fields

Name	XML Tag	Data Type	Length	Req	Notes
Annual Revenue	AnnualRevenue	DTYPE_TEXT	22		
Client Flag	ClientFlag	DTYPE_BOOL	1		
Competitor	Competitor	DTYPE_BOOL	1		
Credit Agency Number	CreditAgencyNumber	DTYPE_TEXT	30		
Cumulative Sales	CumulativeSales	DTYPE_NUMBER	22		
Currency Code	CurrencyCode	DTYPE_TEXT	15	✔	
Current Volume	CurrentVolume	DTYPE_CURRENCY	22		
Dont Call	DontCall	DTYPE_BOOL	1		
Drug Spent Currency Code	DrugSpentCurrencyCode	DTYPE_TEXT	30	✔	Use for picking a related entity
Employees	Employees	DTYPE_NUMBER	22		
Facility Flag	FacilityFlag	DTYPE_BOOL	1		
Federal Tax Number	FederalTaxNumber	DTYPE_TEXT	15		
Financing Method	FinancingMethod	DTYPE_TEXT	30		Bound by FINCORP_CLIENT_FINANCING
Gross Potential Value	GrossPotentialValue	DTYPE_CURRENCY	22		
Group Type Code	GroupTypeCode	DTYPE_TEXT	30	✔	Bound by GROUP_TYPE_CD
Home Page	HomePage	DTYPE_TEXT	100		
Id	Id	DTYPE_ID	150		For cross-referencing
Integration Id	IntegrationId	DTYPE_TEXT	30		
Internal Org Flag	InternalOrgFlag	DTYPE_BOOL	1	✔	
Investor Flag	InvestorFlag	DTYPE_BOOL	1		
Last Year Sales	LastYearSales	DTYPE_CURRENCY	22		

Table 106. Account Integration Component and Integration Component Fields

Name	XML Tag	Data Type	Length	Req	Notes
Legal Status	LegalStatus	DTYPE_TEXT	30		Bound by UT_LEGAL_STATUS
Location	Location	DTYPE_TEXT	50		
Main Fax Number	MainFaxNumber	DTYPE_PHONE	40		
Main Phone Number	MainPhoneNumber	DTYPE_PHONE	40		
Master Account Id	MasterAccountId	DTYPE_ID	15		
Name	Name	DTYPE_TEXT	100	✔	
operation	operation	DTYPE_TEXT	30		
Partner Flag	PartnerFlag	DTYPE_BOOL	1		
Party UId	PartyUId	DTYPE_TEXT	100	✔	
Privacy Code	PrivacyCode	DTYPE_TEXT	30	✔	Bound by PRIVACY_CODE
Reference Flag	ReferenceFlag	DTYPE_BOOL	1		
Referenciable	Referenciable	DTYPE_BOOL	1		
Revenue Growth	RevenueGrowth	DTYPE_NUMBER	22		
Service Provider Flag	ServiceProviderFlag	DTYPE_TEXT	1		
Status	Status	DTYPE_TEXT	30		Bound by FINS_COMPETITOR_STATUS_MLOV
Total # Meters	TotalMeters	DTYPE_NUMBER	22		
Total Potential Volume	TotalPotentialVolume	DTYPE_CURRENCY	22		
Type	Type	DTYPE_TEXT	30	✔	Bound by ACCOUNT_TYPE
Unionized	Unionized	DTYPE_BOOL	1		
YTD Net Contribution	YTDNetContribution	DTYPE_CURRENCY	22		
IC: Account_Business Address					
Address Id	AddressId	DTYPE_ID	30	✔	

Table 106. Account Integration Component and Integration Component Fields

Name	XML Tag	Data Type	Length	Req	Notes
Address Integration Id	AddressIntegrationId	DTYPE_TEXT	30		
Bill Address Flag	BillAddressFlag	DTYPE_TEXT	1		
City	City	DTYPE_TEXT	50		
Country	Country	DTYPE_TEXT	50		Use for pick a related entity
Email Address	EmailAddress	DTYPE_TEXT	50		
IsPrimaryMVG	IsPrimaryMVG	DTYPE_TEXT	1		
operation	operation	DTYPE_TEXT	30		
Postal Code	PostalCode	DTYPE_TEXT	30		
searchspec	searchspec	DTYPE_TEXT	250		
Ship Address Flag	ShipAddressFlag	DTYPE_TEXT	1		
State	State	DTYPE_TEXT	10		Bound by STATE_ABBREV
Street Address	StreetAddress	DTYPE_TEXT	200		
IC: Action					
Alarm	Alarm	DTYPE_BOOL	1		
Assignment Excluded	AssignmentExcluded	DTYPE_BOOL	1		
Associated Cost	AssociatedCost	DTYPE_CURRENCY	22		
Billable Flag	BillableFlag	DTYPE_BOOL	1		
Call Id	CallId	DTYPE_TEXT	30		
Category	Category	DTYPE_TEXT	30	✔	Use for picking a related entity
Class	Class	DTYPE_TEXT	30		Bound by FS_ACTIVITY_CLASS
Comment	Comment	DTYPE_TEXT	“1,500”		
Description	Description	DTYPE_TEXT	100		
Display	Display	DTYPE_TEXT	30		

Table 106. Account Integration Component and Integration Component Fields

Name	XML Tag	Data Type	Length	Req	Notes
Done	Done	DTYPE_UTCDATETIME	7		
Done Flag	DoneFlag	DTYPE_BOOL	1		
Due	Due	DTYPE_UTCDATETIME	7		
Duration Minutes	DurationMinutes	DTYPE_INTEGER	22		Bound by APPOINTMENT_DURATION
Expense Related Flag	ExpenseRelatedFlag	DTYPE_BOOL	1		
Id	Id	DTYPE_TEXT	100	✔	
Integration Id	IntegrationId	DTYPE_TEXT	30		
No Sooner Than Date	NoSoonerThanDate	DTYPE_UTCDATETIME	7		
Objective	Objective	DTYPE_TEXT	30		
operation	operation	DTYPE_TEXT	30		
Owned By	OwnedBy	DTYPE_TEXT	30	✔	Use for pick a related entity
Owned By Id	OwnedById	DTYPE_TEXT	30	✔	
Primary Owned By	PrimaryOwnedBy	DTYPE_TEXT	50		Use for pick a related entity
Primary Owner Id	PrimaryOwnerId	DTYPE_ID	30		
Priority	Priority	DTYPE_TEXT	30		Bound by ACTIVITY_PRIORITY
Private	Private	DTYPE_BOOL	1		
Repeating	Repeating	DTYPE_BOOL	1		
Repeating Expires	RepeatingExpires	DTYPE_DATETIME	7		
searchspec	searchspec	DTYPE_TEXT	250		
Status	Status	DTYPE_TEXT	30		Bound by EVENT_STATUS
Type	Type2	DTYPE_TEXT	30	✔	Bound by TODO_TYPE

Table 106. Account Integration Component and Integration Component Fields

Name	XML Tag	Data Type	Length	Req	Notes
IC: Contact					
Account Id	AccountId	DTYPE_ID	30		
Account Integration Id	AccountIntegrationId	DTYPE_TEXT	30		
Assistant	Assistant	DTYPE_TEXT	50		
Assistant Phone #	AssistantPhone	DTYPE_PHONE	40		
Cellular Phone #	CellularPhone	DTYPE_PHONE	40		
Fax Phone #	FaxPhone	DTYPE_PHONE	40		
First Name	FirstName	DTYPE_TEXT	50	✔	
Home Phone #	HomePhone	DTYPE_TEXT	40		
Job Title	JobTitle	DTYPE_TEXT	75		
Last Name	LastName	DTYPE_TEXT	50	✔	
M/M	MM	DTYPE_TEXT	15		
Middle Name	MiddleName	DTYPE_TEXT	50		
operation	operation	DTYPE_TEXT	30		
Party UId	PartyUId	DTYPE_TEXT	100	✔	
Person UId	PersonUId	DTYPE_ID	30	✔	
Preferred Communication Method	PreferredCommunication Method	DTYPE_TEXT	20		Bound by COMM_METHOD
Privacy Code	PrivacyCode	DTYPE_TEXT	30	✔	Bound by PRIVACY_CODE
Row Id	RowId	DTYPE_ID	100		
searchspec	searchspec	DTYPE_TEXT	250		
Work Phone #	WorkPhone	DTYPE_PHONE	45		
IC: FINCORP Deposit Account					
Account Branch Number	AccountBranchNumber	DTYPE_TEXT	100		
Account Category	AccountCategory	DTYPE_TEXT	30		Bound by FINS_ASSET_PROD _CLASS_MLOV

Table 106. Account Integration Component and Integration Component Fields

Name	XML Tag	Data Type	Length	Req	Notes
Account Number	AccountNumber	DTYPE_TEXT	100	✔	
Account Status	AccountStatus	DTYPE_TEXT	30		Bound by FINS_CRPST_LFLN_IPFS_PLC_MLOV
Branch	Branch	DTYPE_TEXT	50		
Current Balance	CurrentBalance	DTYPE_CURRENCY	22		
Id	Id	DTYPE_TEXT	100		
Integration Id	Integration Id	DTYPE_TEXT	30		
Name	Name	DTYPE_TEXT	100	✔	Use for pick a related entity
operation	operation	DTYPE_TEXT	30		
Product Id	ProductId	DTYPE_ID	100	✔	
searchspec	searchspec	DTYPE_TEXT	250		
Type	Type	DTYPE_TEXT	30		Bound by FINS_OWNERSHIP_TYPE_MLOV
IC: FINCORP Loan Account					
Account Branch Number	AccountBranchNumber	DTYPE_TEXT	30		
Account Category	AccountCategory	DTYPE_TEXT	30		Bound by FINS_ASSET_PROD_CLASS_MLOV
Account Number	AccountNumber	DTYPE_TEXT	100	✔	
Account Status	AccountStatus	DTYPE_TEXT	30		Bound by FINS_CRPST_LFLN_IPFS_PLC_MLOV
Current Balance	CurrentBalance	DTYPE_CURRENCY	22		
Id	Id	DTYPE_TEXT	30		
Integration Id	IntegrationId	DTYPE_TEXT	30		
Name	Name	DTYPE_TEXT	100	✔	Use for pick a related entity

Table 106. Account Integration Component and Integration Component Fields

Name	XML Tag	Data Type	Length	Req	Notes
operation	operation	DTYPE_TEXT	30		
Product Id	ProductId	DTYPE_ID	100	✔	Use for pick a related entity
searchspec	searchspec	DTYPE_TEXT	250		
Type	Type	DTYPE_TEXT	30		FINS_OWNERSHIP _TYPE_MLOV
IC: Related Industry					
Industry	Industry	DTYPE_TEXT	50	✔	
Industry Id	IndustryId	DTYPE_ID	30	✔	
Industry Integration Id	IndustryIntegrationId	DTYPE_TEXT	30		
IsPrimaryMVG	IsPrimaryMVG	DTYPE_TEXT	1		
IsPrimaryMVG	IsPrimaryMVG	DTYPE_TEXT	1		
Organization	Organization	DTYPE_TEXT	100	✔	
Organization Id	OrganizationId	DTYPE_ID	100	✔	
Organization Integration Id	OrganizationIntegrationId	DTYPE_TEXT	30		
Related Organization					
SIC Code	SICCode	DTYPE_TEXT	30	✔	
IC: Related Sales Reps					
Division	Division	DTYPE_TEXT	255	✔	Use to pick a related entity
IsPrimaryMVG	IsPrimaryMVG	DTYPE_TEXT	1		
Login	Login	DTYPE_ID	15		
Position	Position	DTYPE_TEXT	50		
Position Id	PositionId	DTYPE_ID	30		
Position Integration Id	PositionIntegrationId	DTYPE_TEXT	30		
IC: Sub Account					
Account Status	AccountStatus2	DTYPE_TEXT	30		Bound by ACCOUNT_STATUS
Id	Id	DTYPE_ID	100	✔	

Table 106. Account Integration Component and Integration Component Fields

Name	XML Tag	Data Type	Length	Req	Notes
Location	Location	DTYPE_TEXT	50		
Main Phone Number	MainPhoneNumber	DTYPE_PHONE	40		
Name	Name	DTYPE_TEXT	100	✔	
operation	operation	DTYPE_TEXT	30		
Party UId	PartyUId	DTYPE_TEXT	30	✔	
searchspec	searchspec	DTYPE_TEXT	250		
Type	Type	DTYPE_TEXT	255	✔	Bound by ACCOUNT_TYPE
IC: UCM Account Privacy					
Account Id	AccountId	DTYPE_ID			
Affiliate Flag	AffiliateFlag	DTYPE_TEXT	30		
Affiliate Flag	AffiliateFlag	DTYPE_TEXT	30		
Affiliate Method	AffiliateMethod	DTYPE_TEXT	30		
Affiliate Method	AffiliateMethod	DTYPE_TEXT	30		
Affiliate Source	AffiliateSource	DTYPE_TEXT	30		
Affiliate Source	AffiliateSource	DTYPE_TEXT	30		
Affiliate System	AffiliateSystem	DTYPE_TEXT	30		
Affiliate System	AffiliateSystem	DTYPE_TEXT	30		
Channel Flag	ChannelFlag	DTYPE_TEXT	30		
Channel Flag	ChannelFlag	DTYPE_TEXT	30		
Channel Method	ChannelMethod	DTYPE_TEXT	30		
Channel Method	ChannelMethod	DTYPE_TEXT	30		
Channel Source	ChannelSource	DTYPE_TEXT	30		
Channel Source	ChannelSource	DTYPE_TEXT	30		

Table 106. Account Integration Component and Integration Component Fields

Name	XML Tag	Data Type	Length	Req	Notes
Channel System	ChannelSystem	DTYPE_TEXT	30		
Channel System	ChannelSystem	DTYPE_TEXT	30		
Contact Id	ContactId	DTYPE_ID			
Cover Letter Type	CoverLetterType	DTYPE_TEXT	30		
Cover Letter Type	CoverLetterType	DTYPE_TEXT	30		
Cover Letter Type II	CoverLetterTypeII	DTYPE_TEXT	30		
Cover Letter Type II	CoverLetterTypeII	DTYPE_TEXT	30		
IC: UCM Contact Privacy					
Id	Id	DTYPE_ID	30		
Id	Id	DTYPE_ID	30		
Integration Id	IntegrationId	DTYPE_TEXT	30		
Integration Id	IntegrationId	DTYPE_TEXT	30		
Nonaffiliate Flag	NonaffiliateFlag	DTYPE_TEXT	30		
Nonaffiliate Flag	NonaffiliateFlag	DTYPE_TEXT	30		
Nonaffiliate Method	NonaffiliateMethod	DTYPE_TEXT	30		
Nonaffiliate Method	NonaffiliateMethod	DTYPE_TEXT	30		
Nonaffiliate Source	NonaffiliateSource	DTYPE_TEXT	30		
Nonaffiliate Source	NonaffiliateSource	DTYPE_TEXT	30		
Nonaffiliate System	NonaffiliateSystem	DTYPE_TEXT	30		
Nonaffiliate System	NonaffiliateSystem	DTYPE_TEXT	30		
Notice Send Flag	NoticeSendFlag	DTYPE_TEXT	30		

Table 106. Account Integration Component and Integration Component Fields

Name	XML Tag	Data Type	Length	Req	Notes
Notice Send Flag	NoticeSendFlag	DTYPE_TEXT	30		
Notice Sent Date	NoticeSentDate	DTYPE_UTCDATETIME			
Notice Sent Date	NoticeSentDate	DTYPE_UTCDATETIME			
Notice Sent Reason	NoticeSentReason	DTYPE_TEXT	100		
Notice Sent Reason	NoticeSentReason	DTYPE_TEXT	100		
Notice Sent Type	NoticeSentType	DTYPE_TEXT	30		
Notice Sent Type	NoticeSentType	DTYPE_TEXT	30		
OFAC Flag	OFACFlag	DTYPE_BOOL			
OFAC Flag	OFACFlag	DTYPE_BOOL			
PEND End Date	PENDEndDate	DTYPE_UTCDATETIME			
PEND End Date	PENDEndDate	DTYPE_UTCDATETIME			
PEND Start Date	PENDStartDate	DTYPE_UTCDATETIME			
PEND Start Date	PENDStartDate	DTYPE_UTCDATETIME			
Privacy Parent Type	PrivacyParentType	DTYPE_TEXT	30		='Account'
Privacy Parent Type	PrivacyParentType	DTYPE_TEXT	30		
Privacy State Code	PrivacyStateCode	DTYPE_TEXT	30		
Privacy State Code	PrivacyStateCode	DTYPE_TEXT	30		
Receive Annual Notice Flag	ReceiveAnnualNoticeFlag	DTYPE_TEXT	30		
Receive Annual Notice Flag	ReceiveAnnualNoticeFlag	DTYPE_TEXT	30		

Table 106. Account Integration Component and Integration Component Fields

Name	XML Tag	Data Type	Length	Req	Notes
Receive Annual Notice Method	ReceiveAnnualNoticeMethod	DTYPE_TEXT	30		
Receive Annual Notice Method	ReceiveAnnualNoticeMethod	DTYPE_TEXT	30		
Receive Annual Notice Source	ReceiveAnnualNoticeSource	DTYPE_TEXT	30		
Receive Annual Notice Source	ReceiveAnnualNoticeSource	DTYPE_TEXT	30		
Receive Annual Notice System	ReceiveAnnualNoticeSystem	DTYPE_TEXT	30		
Receive Annual Notice System	ReceiveAnnualNoticeSystem	DTYPE_TEXT	30		
State PEND Period	StatePENDPeriod	DTYPE_INTEGER			
State PEND Period	StatePENDPeriod	DTYPE_INTEGER			
Telemarketing Flag	TelemarketingFlag	DTYPE_TEXT	30		
Telemarketing Flag	TelemarketingFlag	DTYPE_TEXT	30		
Telemarketing Method	TelemarketingMethod	DTYPE_TEXT	30		
Telemarketing Method	TelemarketingMethod	DTYPE_TEXT	30		
Telemarketing Source	TelemarketingSource	DTYPE_TEXT	30		
Telemarketing Source	TelemarketingSource	DTYPE_TEXT	30		
Telemarketing System	TelemarketingSystem	DTYPE_TEXT	30		
Telemarketing System	TelemarketingSystem	DTYPE_TEXT	30		
Updated	Updated	DTYPE_DATETIME	30		
Updated	Updated	DTYPE_DATETIME	30		

Table 106. Account Integration Component and Integration Component Fields

Name	XML Tag	Data Type	Length	Req	Notes
Updated By	UpdatedBy	DTYPE_ID	30		
Updated By	UpdatedBy	DTYPE_ID	30		

Cross Reference

UCM Cross Reference messages are used in query and response operations for account, contact and household cross references. There is a pair of request and response messages for each object. Figure 27 describes the structure of these messages.

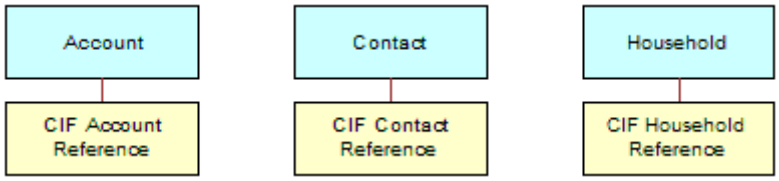

Figure 27. UCM Cross Reference Structure

Table 107 lists the Account Cross Reference Integration Objects.

Table 107. Account Cross Reference Integration Objects

Integration Object	Business Component	XML Tag	User Keys
UCMAccountExternalID	Contact	LookUpAccountExternalID	UK1:PartyUId UK2: Id
UCMAccountExternalIDRs	Contact	LookUpAccountExternalIDResponse	UK1:PartyUId UK2: Id

Table 108 lists the Contact Cross Reference Integration Objects.

Table 108. Contact Cross Reference Integration Objects

Integration Object	Business Component	XML Tag	User Keys
UCMContactExternalID	Contact	LookUpContactExternalID	UK1:PartyUId UK2: Id
UCMContactExternalIDRs	Contact	LookUpContactExternalIDResponse	UK1:PartyUId UK2: Id

Table 109 lists the Household Cross Reference Integration Objects.

Table 109. Household Cross Reference Integration Objects

Integration Object	Business Component	XML Age	User Keys
UCMHouseholdExternalID	Household	LookUpHouseholdExternalID	UK1:Id UK2:IntegrationId UK3:PartyUId
UCMHouseholdExternalIDRs	Household	LookUpHouseholdExternalID Response	UK1:Id UK2:IntegrationId UK3:PartyUId

Table 110 lists the Account Cross Reference Integration Component and Integration Component Fields.

Table 110. Account Cross Reference Integration Component and Integration Component Fields

Name	XML Tag	Data Type	Length	Required	Note
IC: Account					
Account Status	AccountStatus	DTYPE_TEXT	24		
Account Type Code	AccountTypeCode	DTYPE_TEXT	30		
Annual Revenue	AnnualRevenue	DTYPE_TEXT			
Client Flag	ClientFlag	DTYPE_TEXT			
Competitor	Competitor	DTYPE_TEXT			
Conflict Id	ConflictId	DTYPE_ID	30		
Created	Created	DTYPE_DATETIME	30		
Credit Agency Number	CreditAgencyNumber	DTYPE_TEXT	30		
Currency Code	CurrencyCode	DTYPE_TEXT	15		
Facility Flag	FacilityFlag	DTYPE_TEXT			
Federal Tax Number	FederalTaxNumber	DTYPE_TEXT	15		
Home Page	HomePage	DTYPE_TEXT	100		
Id	Id	DTYPE_ID	30		
Integration Id	IntegrationId	DTYPE_TEXT	30		
Internal Org Flag	InternalOrgFlag	DTYPE_TEXT			
Investor Flag	InvestorFlag	DTYPE_TEXT			
Legal Status	LegalStatus	DTYPE_TEXT	30		

Table 110. Account Cross Reference Integration Component and Integration Component Fields

Name	XML Tag	Data Type	Length	Required	Note
Location	Location	DTYPE_TEXT	50		
Main Fax Number	MainFaxNumber	DTYPE_TEXT			
Main Phone Number	MainPhoneNumber	DTYPE_TEXT			
Master Account Id	MasterAccountId	DTYPE_ID			
Mod Id	ModId	DTYPE_ID	30		
Name	Name	DTYPE_TEXT	100		
operation	operation	DTYPE_TEXT	30		
Partner Flag	PartnerFlag	DTYPE_TEXT			
Party UId	PartyUId	DTYPE_TEXT	100		
searchspec	searchspec	DTYPE_TEXT	250		
Service Provider Flag	ServiceProviderFlag	DTYPE_TEXT			
Status	Status	DTYPE_TEXT	30		
Type	Type	DTYPE_TEXT	30		
Updated	Updated	DTYPE_DATETIME	30		
IC: CIF Account Reference					
Account Id	AccountId	DTYPE_ID		✔	
CIF System Id	CIFSystemId	DTYPE_ID		✔	
Comment	Comment	DTYPE_TEXT	250		
Conflict Id	ConflictId	DTYPE_ID	30		
Created	Created	DTYPE_DATETIME	30		
External Id1	ExternalId1	DTYPE_TEXT	100		
External Id2	ExternalId2	DTYPE_TEXT	100		
External Id3	ExternalId3	DTYPE_TEXT	100		
Id	Id	DTYPE_ID	30		
Mod Id	ModId	DTYPE_ID	30		
operation	operation	DTYPE_TEXT	30		
searchspec	searchspec	DTYPE_TEXT	250		
System Name	SystemName	DTYPE_TEXT	100		
System Number	SystemNumber	DTYPE_TEXT	30		
Updated	Updated	DTYPE_DATETIME	30		

Table 111 lists the Contact Cross Reference Integration Component and Integration Component Fields.

Table 111. Contact Cross Reference Integration Component and Integration Component Fields

Name	XML Tag	Data Type	Length	Required	Note
IC: Contact					
Cellular Phone #	CellularPhone	DTYPE_TEXT			
Conflict Id	ConflictId	DTYPE_ID	30		
Created	Created	DTYPE_DATETIME	30		
Date of Birth	DateofBirth	DTYPE_DATETIME			
Email Address	EmailAddress	DTYPE_TEXT	50		
Fax Phone #	FaxPhone	DTYPE_TEXT			
First Name	FirstName	DTYPE_TEXT	50		
Home Phone #	HomePhone	DTYPE_TEXT			
Id	Id	DTYPE_ID	30		
Integration Id	IntegrationId	DTYPE_TEXT	30		
Job Title	JobTitle	DTYPE_TEXT	75		
Last Name	LastName	DTYPE_TEXT	50		
M/F	MF	DTYPE_TEXT	30		
M/M	MM	DTYPE_TEXT	15		
Marital Status	MaritalStatus	DTYPE_TEXT	30		
Middle Name	MiddleName	DTYPE_TEXT	50		
Mod Id	ModId	DTYPE_ID	30		
Mother Maiden Name	MotherMaidenName	DTYPE_TEXT	50		
operation	operation	DTYPE_TEXT	30		
Party UId	PartyUId	DTYPE_TEXT	100		
Person UId	PersonUId	DTYPE_TEXT	100		
searchspec	searchspec	DTYPE_TEXT	250		
Social Security Number	SocialSecurityNumber	DTYPE_TEXT			
Status	Status	DTYPE_TEXT	30		
Updated	Updated	DTYPE_DATETIME	30		
Work Phone #	WorkPhone	DTYPE_TEXT			
Work Phone Extension	WorkPhoneExtension	DTYPE_TEXT	10		

Table 111. Contact Cross Reference Integration Component and Integration Component Fields

Name	XML Tag	Data Type	Length	Required	Note
IC: CIF Contact Reference					
CIF System Id	CIFSystemId	DTYPE_ID		✔	
Comment	Comment	DTYPE_TEXT	250		
Conflict Id	ConflictId	DTYPE_ID	30		
Contact Id	ContactId	DTYPE_ID		✔	
Created	Created	DTYPE_DATETIME	30		
External Id1	ExternalId1	DTYPE_TEXT	100		
External Id2	ExternalId2	DTYPE_TEXT	100		
External Id3	ExternalId3	DTYPE_TEXT	100		
Id	Id	DTYPE_ID	30		
Mod Id	ModId	DTYPE_ID	30		
operation	operation	DTYPE_TEXT	30		
searchspec	searchspec	DTYPE_TEXT	250		
System Name	SystemName	DTYPE_TEXT	100		
System Number	SystemNumber	DTYPE_TEXT	30		
Updated	Updated	DTYPE_DATETIME	30		

Table 112 lists the Household Cross Reference Integration Component and Integration Component Fields.

Table 112. Household Cross Reference Integration Component and Integration Component Fields

Name	XML Tag	Data Type	Length	Required	Note
IC: Household					
Category	Category	DTYPE_TEXT	30		
Conflict Id	ConflictId	DTYPE_ID	30		
Created	Created	DTYPE_DATETIME	30		
Currency Code	CurrencyCode	DTYPE_TEXT	20		
Fax Number	FaxNumber	DTYPE_TEXT			
Household Name	HouseholdName	DTYPE_TEXT	50		
Household Number	HouseholdNumber	DTYPE_TEXT	100		
Id	Id	DTYPE_ID	30		

Table 112. Household Cross Reference Integration Component and Integration Component Fields

Name	XML Tag	Data Type	Length	Required	Note
Integration Id	IntegrationId	DTYPE_TEXT	30		
Mod Id	ModId	DTYPE_ID	30		
operation	operation	DTYPE_TEXT	30		
Party UId	PartyUId	DTYPE_TEXT	100		
Phone Number	PhoneNumber	DTYPE_TEXT			
searchspec	searchspec	DTYPE_TEXT	250		
Status	Status	DTYPE_TEXT	30		
Type	Type	DTYPE_TEXT	30		
Updated	Updated	DTYPE_DATETIME	30		
IC: CIF Household Reference					
CIF System Id	CIFSystemId	DTYPE_ID		✔	
Comment	Comment	DTYPE_TEXT	250		
Conflict Id	ConflictId	DTYPE_ID	30		
Created	Created	DTYPE_DATETIME	30		
External Id1	ExternalId1	DTYPE_TEXT	100		
External Id2	ExternalId2	DTYPE_TEXT	100		
External Id3	ExternalId3	DTYPE_TEXT	100		
Household Id	ContactId	DTYPE_ID		✔	
Id	Id	DTYPE_ID	30		
Mod Id	ModId	DTYPE_ID	30		
operation	operation	DTYPE_TEXT	30		
searchspec	searchspec	DTYPE_TEXT	250		
System Name	SystemName	DTYPE_TEXT	100		
System Number	SystemNumber	DTYPE_TEXT	30		
Updated	Updated	DTYPE_DATETIME	30		

8 Configuring and Administering Siebel Universal Customer Master Privacy Management

This chapter describes the configuration requirements and basic administrative tasks for Siebel Universal Customer Master (UCM) Privacy Management. It is to be used in reference with *Siebel Business Rules Administration Guide*.

This chapter includes the following topics:

- "About Siebel Universal Customer Master Privacy Management" on page 221
- "Process of Configuring Siebel Universal Customer Master Privacy Management" on page 222
- "Administering Siebel Universal Customer Master Privacy Management" on page 228

About Siebel Universal Customer Master Privacy Management

Universal Customer Master (UCM) Privacy Management provides an infrastructure that allows companies to implement privacy solutions to their customer data. These solutions assist companies to protect personal customer data and to comply with government privacy regulations.

The UCM Privacy Management implementation requires three major components:

- Universal Customer Master (UCM), which maintains the privacy preferences.
- An embedded rules engine, which allows rules creation and processing. For background information on the rules engine, see *Siebel Business Rules Administration Guide*.

About Siebel Universal Customer Master Privacy Management Features and Capabilities

The UCM Privacy Management infrastructure consists of the following features and capabilities:

- **Privacy Data Model.** Provides preconfigured privacy data definitions to capture customer data sharing consent; solicitation choices; election to receive privacy notices, and other necessary data to generate compliance reports and revision history.
- **Privacy Views.** Provide privacy data status and history views at the account, contact, financial account, and financial account contact level. The history views allow for compliance monitoring of updates to the customer's privacy preferences.
- **Financial Account Views.** Support a single view of financial accounts. Financial accounts include a data synchronization business service, UCM FINCORP Account Source Data and History Service, which stores data (as integration object instances) into the database.

- **UCM Privacy Workflows.** Provided to administer privacy changes associated with contacts on a financial account. For further detail on the UCM Privacy Workflows, see "About Siebel Universal Customer Master Privacy Workflow Processes" on page 223.
- **UCM Privacy Management Vocabulary.** Preconfigured privacy rule building blocks, also known as privacy vocabularies, and a material set of sample privacy rules. The vocabulary has modeled UCM Privacy entities and attributes and is built in the context of various privacy regulations. The sample rules can be used to create custom rules specific to a company's privacy policies. For more information, see Chapter 9, "Privacy Management Solution".
- **Financial Account Integration Web Services.** Provides Web services to interact with systems across the enterprise. These Web services are enhanced to include privacy integration objects to query, update, and insert customers' privacy data in UCM.

About Siebel Universal Customer Master Privacy Management Configuration

The Universal Customer Master (UCM) Privacy Management infrastructure requires configurations to make sure the privacy features are operational in the Siebel UCM application. Configurations are necessary on the Siebel UCM application, Siebel Business Rules Developer, and integration technologies. For complete details on the tasks necessary to configure UCM Privacy Management, see "Process of Configuring Siebel Universal Customer Master Privacy Management" on page 222.

About Siebel Universal Customer Master Privacy Management Administration

Siebel Universal Customer Master (UCM) Privacy Management is administered from the Siebel UCM user interface as well as the Siebel Business Rules Developer, which is based on the HaleyAuthority product from Haley Systems, Inc.,

Privacy rule authoring and administration occurs solely in the Siebel Business Rules Developer. For complete information on creating privacy rules, see Chapter 9, "Privacy Management Solution" and *Siebel Business Rules Administration Guide*.

Process of Configuring Siebel Universal Customer Master Privacy Management

Perform the tasks in the following bulleted list to configure Siebel Universal Customer Master (UCM) Privacy Management functionality. For background information on UCM Privacy Management, see "About Siebel Universal Customer Master Privacy Management" on page 221.

- Make sure preliminary software requirements are met. For information on this task, see "Preliminary Requirements Before Configuring Siebel Universal Customer Master Privacy Management" on page 223.

- Make sure UCM Privacy Workflow processes are available or deployed. For information on this task, see"Deploying Siebel Universal Customer Master Privacy Workflow Processes" on page 225.
- Make sure all applicable user properties are set. For more information, see "Setting Privacy-Related User Properties" on page 225.

Preliminary Requirements Before Configuring Siebel Universal Customer Master Privacy Management

This task is a step in "Process of Configuring Siebel Universal Customer Master Privacy Management" on page 222.

Before configuring Siebel Universal Customer Master (UCM) Privacy Management, make sure the following preliminary software requirements are met:

- UCM SIA installed
- Siebel Business Rules Developer installed (part of the Siebel Tools installation)

The Siebel Business Rules Developer is required for design time of privacy rules but is not required in run time.

About UCM Rule Engine Helper

The Universal Customer Master (UCM) Rule Engine Helper is a prebuilt helper business service for master data application to help preparing input for the rule engine and processing output from the rule engine.

To review specific details on business services, including user properties, methods, and method arguments, use Siebel Tools. For information on using Siebel Tools, see *Using Siebel Tools* on the *Siebel Bookshelf*.

About Siebel Universal Customer Master Privacy Workflow Processes

Siebel Universal Customer Master (UCM) Privacy Management consists of the following preconfigured UCM Privacy workflow processes. For background information on workflow processes, see *Siebel Business Process Framework: Workflow Guide* on the *Siebel Bookshelf*.

The following workflows and the corresponding sample privacy rules are provided to administer privacy changes associated with Contacts on a Financial Account. Similar workflows and privacy rules can be created in UCM and Siebel Business Rules Developer corresponding to privacy changes associated with prospects, consumers, companies, organizations, or households. Run-time events are used to trigger these workflows.

The UCM Privacy Workflow processes are:

- "UCM Privacy Process" on page 224
- "UCM Privacy Process—Pending Expired" on page 224
- "UCM Privacy Process—Reevaluate Privacy Batch" on page 225

UCM Privacy Process

This workflow process is invoked by run-time events through the business component user property Named Method and can be configured to support additional events. This workflow handles the following events:

- **Financial account type changes.** This workflow is triggered, by way of run-time events, when a financial account type changes, for example, a car loan has changed from a consumer car loan to a business car loan because the car is now used as a company vehicle. Some privacy regulations mandate different compliance requirements for consumers than those for businesses. The sample rules provide examples where business financial accounts, by default, opt-in data sharing compared with consumer financial accounts, by default, have more restrictive data sharing.
- **New financial account contact has been added.** This workflow is triggered, by way of run-time events, typically during a financial account new booking process where a new financial account with contact information is submitted to UCM from an external system such as a financial loan origination system. This workflow calls the rules engine to set the initial privacy preference values for each contact and for the financial account to which the contact belongs. This workflow is also triggered if a new financial contact is added. The sample privacy rules written in Siebel Business Rules Developer determine the contact privacy preferences based on each contact's primary garage address state. You must change the default privacy preference values stored in the privacy rule based on privacy regulations and the company's data sharing practice.
- **Financial account contact privacy setting has been updated.** This workflow is triggered, by way of run-time events, when a financial account contact’s privacy setting is changed.
- **Financial account contact status has been set to Inactive.** This workflow is triggered, by way of run-time events, when a contact changes its status from active to inactive. When a contact’s status changes, its privacy sharing rights may change, too. The sample privacy rules provide the logic to set updated privacy preference values for the inactive contact, as well as re-evaluate the privacy status of the financial account to which the inactive contact associates.
- **Financial account contact garage address has been changed.** This workflow is triggered, by way of run-time events, when a contact on a financial account changes his or her address. The current U.S. privacy regulation allows states to pass their own privacy regulation and therefore different privacy rules and business logic may be applied depending on the state where the financial account belongs. For example, California Financial Information Privacy Act (also known as CA SB1) requires that financial institutions provide California-based residents with a special disclosure and allow a 45-day initial waiting period before sharing occurs. This initial waiting period may be different for other states. The sample Haley Privacy Knowledge Base provides examples of such logic.

UCM Privacy Process—Pending Expired

Different states in the U.S. can mandate different initial waiting periods during which a Contact may elect to opt-in or opt-out sharing of his or her personal information. This workflow process can be invoked using a workflow process batch manager server component. For background information on how to run workflow process batch manager server component, see *Siebel System Administration Guide*. It can be run in background with search specification to pick up all the financial accounts where the pending period have expired. The length of the pending period is specified as rules and is set in UCM when the financial account is created or updated. The pending period is preconfigured to 50 days.

UCM Privacy Process—Reevaluate Privacy Batch

This workflow process can be invoked using a workflow process manager server component. For background information on how to run workflow process manager server component, see *Siebel System Administration Guide*. It is a self-looping workflow process to avoid workflow batch process manager's query overhead. This workflow is triggered when there is a change to the privacy global settings, for example, some states' law changes the default privacy settings for telemarketing sharing. The sample rules provide some basic logic to re-evaluate market sharing for the financial account and the contact.

Deploying Siebel Universal Customer Master Privacy Workflow Processes

This task is a step in "Process of Configuring Siebel Universal Customer Master Privacy Management" on page 222.

Deploying Siebel Universal Customer Master (UCM) Privacy workflow processes is a necessary task to configure the UCM Privacy feature. To deploy a workflow process, see the task provided in *Siebel Business Process Framework: Workflow Guide* on the *Siebel Bookshelf*.

For background information on the UCM Privacy workflow processes, see "About Siebel Universal Customer Master Privacy Workflow Processes" on page 223.

Setting Privacy-Related User Properties

This task is a step in "Process of Configuring Siebel Universal Customer Master Privacy Management" on page 222.

Using Siebel Tools, set the user properties in the following business components:

- "UCM FINCORP Account Contact Privacy" on page 226
- "FINCORP Account" on page 227
- "UCM FINCORP Account Contact Address" on page 227
- "UCM FINCORP Account Contact Address Source Data and History" on page 227
- "FINCORP Account Contact" on page 228

UCM FINCORP Account Contact Privacy

Table 113 lists the user properties that need to be set for the UCM FINCORP Account Contact Privacy business component.

Table 113. User Properties in the UCM FINCORP Account Contact Privacy Business Component

Name	Value	Comments
Named Method 1	"AccountContactPrivacyChanged", "INVOKESVC", "UCM FINCORP Account Contact Privacy", "Workflow Process Manager", "RunProcess", "'ProcessName'", "'UCM Privacy Process'", "'RowId'", "[Financial Account Id]", "'Module Names'", "'Input interpretation,Initialization,Reevaluate privacy,Core Privacy Rules,Generate Output'"	Invoke the privacy workflow process to re-evaluate the privacy changes.
Named Method 2	"GetAddress", "INVOKE", "UCM FINCORP Account Contact Privacy", "RefreshRecord"	Fresh the current record. Mainly for UI
Named Method 3	"AccountContactPrivacyNew", "INVOKESVC", "UCM FINCORP Account Contact Privacy", "Workflow Process Manager", "RunProcess", "'ProcessName'", "'UCM Privacy Process'", "'RowId'", "[Financial Account Id]", "'Module Names'", "'Input interpretation,Initialization,Reevaluate privacy,New Financial Account_New Contact Event,Core Privacy Rules,Generate Output'"	Invoke the privacy workflow process when a new contact privacy has been created.
PEND Date Rollup Parent Buscomp Name	UCM FINCORP Account Privacy	The parent business component name for rolling up the PEND end date.
PEND Date Rollup Parent Field Name	Financial Account Id	The parent business component field which serves as the parent Id field in PEND end date rollup.
PEND End Required Fields 1	Nonaffiliate Flag	A “must check” field when rolling up the PEND end date. If this field is still PEND, pend does not end. Only when all the “must check” fields are not PEND, the pend ends.
PEND End Required Fields 2	Affiliate Flag	
PEND End Required Fields 3	Channel Flag	
PEND End Required Fields 4	Telemarketing Flag	
PEND End Required Fields 5	Receive Annual Notice Flag	

FINCORP Account

Table 114 lists the user properties that need to be set for the FINCORP Account business component.

Table 114. User Properties in the Fincorp Account Business Component

Name	Value	Comments
Named Method 1	"AccountTypeChanged", "INVOKESVC", "FINCORP Account", "Workflow Process Manager", "RunProcess", "'ProcessName'", "'UCM Privacy Process'", "'RowId'", "[Id]", "'Module Names'", "'Input interpretation,Initialization,Account Type Change Event,Core Privacy Rules,Generate Output'"	Invoke the privacy workflow process when account type has been changed.

UCM FINCORP Account Contact Address

Table 115 lists the user properties that need to be set for the UCM FINCORP Account Contact Address business component.

Table 115. User Properties in the UCM FINCORP Account Contact Address Business Component

Name	Value	Comments
Named Method 1	"GetPrivacy", "INVOKE", "UCM FINCORP Account Contact Address", "RefreshRecord"	Fresh the current record. Mainly for UI

UCM FINCORP Account Contact Address Source Data and History

Table 116 lists the user properties that need to be set for the UCM FINCORP Account Contact Address Source Data and History business component.

Table 116. User Properties in the UCM FINCORP Account Contact Address Source Data and History Business Component

Name	Value	Comments
Named Method 1	"GetFinancialAccount", "INVOKE", "UCM FINCORP Account Contact Address Source Data and History", "RefreshRecord"	Fresh the current record. Mainly for UI
Named Method 2	"AccountContactAddressChanged", "INVOKESVC", "UCM FINCORP Account Contact Address Source Data and History", "Workflow Process Manager", "RunProcess", "'ProcessName'", "'UCM Privacy Process'", "'RowId'", "[Financial Account Id]", "'Module Names'", "'Input interpretation,Initialization,Consumer Address Change Event,Core Privacy Rules,Generate Output'"	Invoke the privacy workflow process when garage address has been changed.

FINCORP Account Contact

Table 117 lists the user properties that need to be set for the FINCORP Account Contact business component.

Table 117. User Properties in the FINCORP Account Contact Business Component

Name	Value	Comments
Named Method 2	"ContactInactivated", "INVOKESVC", "FINCORP Account Contact", "Workflow Process Manager", "RunProcess", "'ProcessName'", "'UCM Privacy Process'", "'RowId'", "[Asset Id]", "'Module Names'", "'Contact Inactivated Event'"	Invoke the privacy workflow process when a financial account has been inactivated.

Administering Siebel Universal Customer Master Privacy Management

Data stewards or administrators can monitor privacy options for Contacts, Accounts, and Financial Accounts in the Siebel UCM user interface. The primary administrative tasks of authoring privacy rules for UCM Privacy Management are undertaken in Siebel Business Rules Developer.

Managing Privacy Data

Siebel Universal Customer Master (UCM) Privacy Management provides preconfigured privacy tabs to the Siebel UCM application screens. From these tabs, administrators or data stewards can monitor the privacy elections and history of individual contacts, accounts, financial accounts, or financial account contacts. For example, they can see if a customer has opted out of a marketing campaign.

For an example of the Contacts, Accounts, or Financial Accounts Privacy tab, see Figure 28.

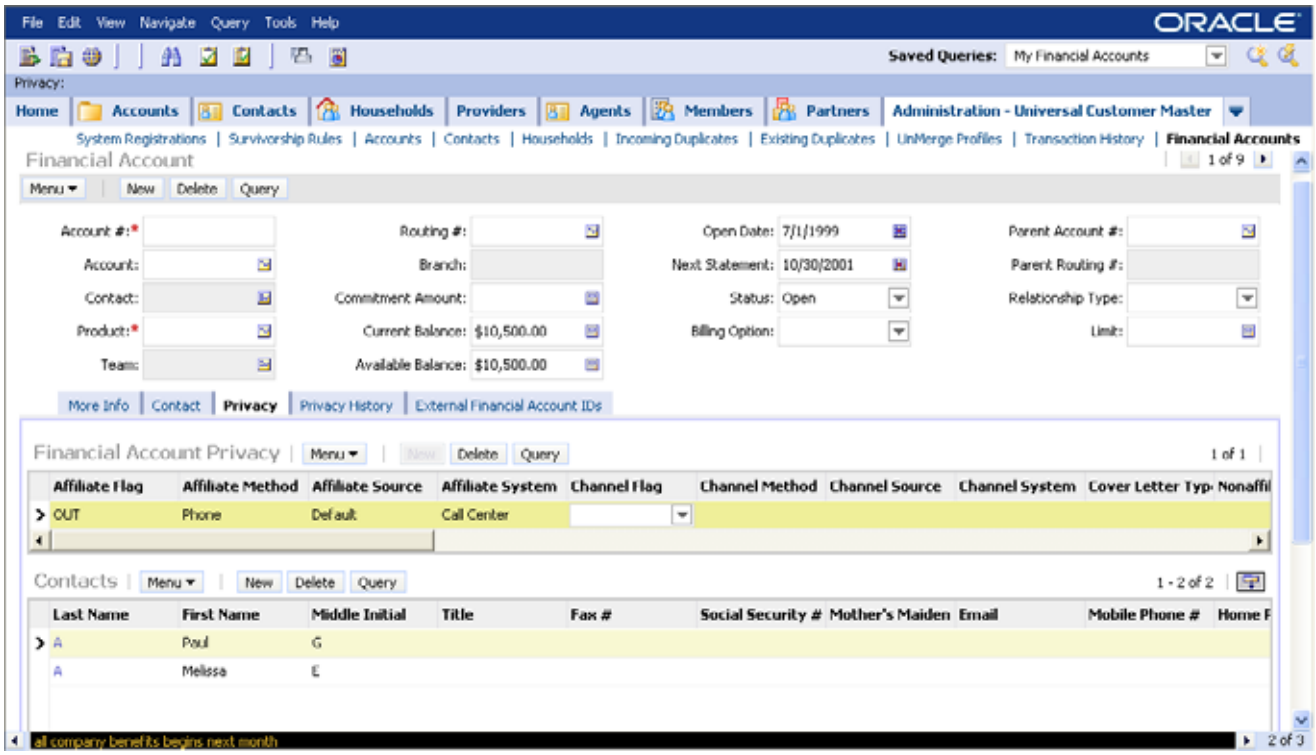

Figure 28. Financial Account, Financial Account Contacts Privacy Tab

Manage Privacy Rule Modules

UCM provides seed privacy rule modules. You can display these rule modules in the Administration - Rule Modules screen. For an example, see Figure 29.

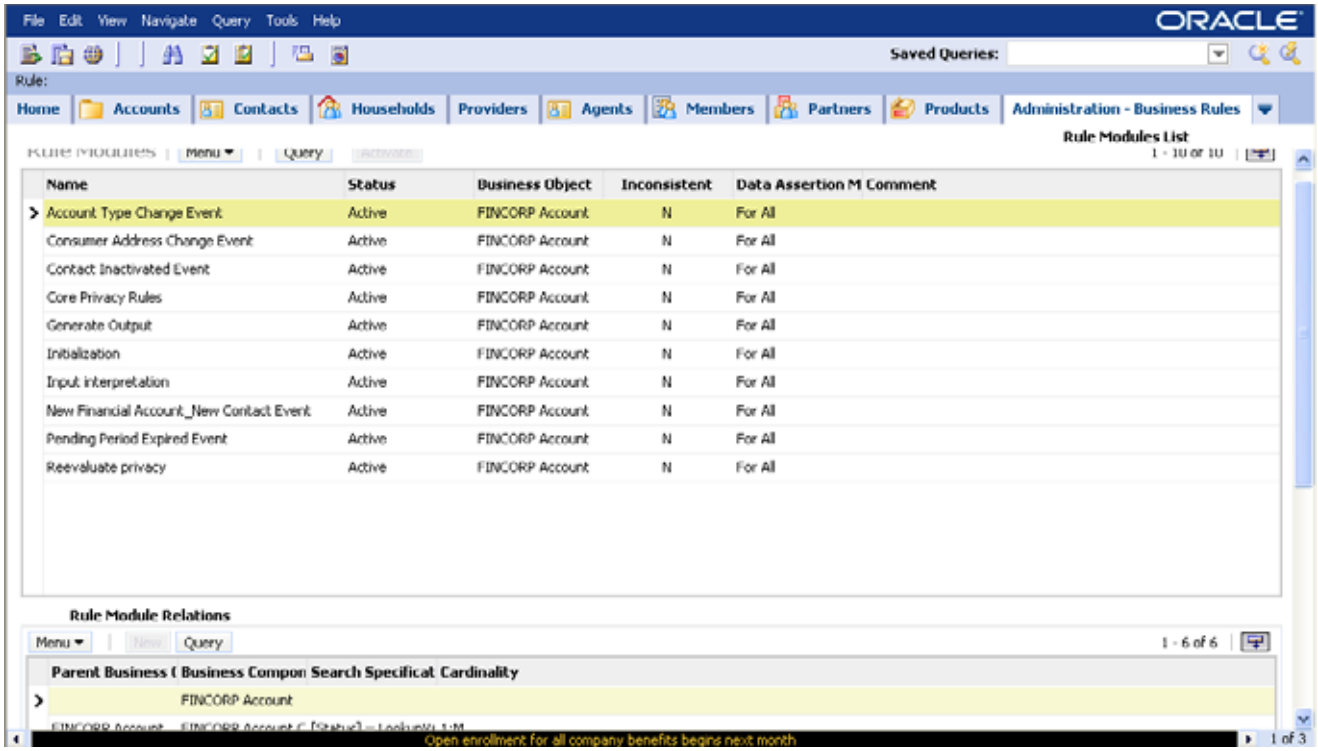

Figure 29. Administration - Rule Modules View with Privacy Rule Module Seed Data

Managing Privacy Runtime Events

UCM privacy workflow process can be triggered through runtime events. Privacy related run-time events can be viewed in the Admin - Runtime Events screen. Query on:

```
Object = Business Component, Action Set = UCM*
```

Run-time events in the business component level are used in privacy management to invoke a workflow process which communicates with the rule engine to get privacy settings. Run-time events do the condition checking over different business components (entities) and trigger the desired actions when conditions are met.

The following tables provide details of run-time events used in the default privacy management settings.

Table 118 lists Retrieve Initial Privacy Setting run-time events.

Table 118. Run-time Events

Sequence	Object Name	Event	Conditional Expression	Action Set Name
1	UCM FINCORP Account Contact Privacy	WriteRecordNew	(GetProfileAttr('Haley Message') IS NULL OR GetProfileAttr('Haley Message')<>'Y')	UCM Set FACPrivacyNew To Y And Refresh Record
2	UCM FINCORP Account Contact Privacy	WriteRecordNew	GetProfileAttr('FACPrivacyNew')='Y' AND [Address Id] IS NOT NULL	UCM New Financial Account/Financial Account Contact

Table 119 lists initial privacy settings where Action Set = “UCM Set FACPrivacyNew To Y And Refresh Record” with Actions.

Table 119. Action Set = “UCM Set FACPrivacyNew To Y And Refresh Record”

Name	Action Type	Sequence	More Info
UCM Set Profile Attribute	Attribute Set	1	Profile Attribute=FACPrivacyNew; Set Operator=Set; Value="Y"
UCM Refresh Record	Invoke Method	2	Method Name=GetAddress

Table 120 lists initial privacy settings where Action Set = “UCM New Financial Account/Financial Account Contact” with Actions:

Table 120. Action Set = “UCM New Financial Account/Financial Account Contact”

Name	Action Type	Sequence	More Info
UCM Set Profile Attribute	Attribute Set	1	Profile Attribute=FACPrivacyNew; Set Operator=Set; Value="N"
UCM New Financial Account/Financial Account Contact	Invoke Method	2	Method Name=AccountContactPrivacyNew
Refresh	Invoke Method	3	Method Name=GetAddress

Table 121 lists Re-evaluate privacy setting, run-time event.

Table 121. Re-evaluate Privacy Setting, Run-time Event

Sequence	Object Name	Event	Conditional Expression	Action Set Name
1	UCM FINCORP Account Contact Privacy	WriteRecordUpdated	GetProfileAttr('Haley Message') IS NULL OR GetProfileAttr('Haley Message')<>'Y	'UCM Update Financial Account Contact Privacy

Table 122 lists Re-evaluate privacy setting, where Action Set = “UCM Update Financial Account Contact Privacy” with Actions.

Table 122. Action Set = “UCM Update Financial Account Contact Privacy”

Name	Action Type	Sequence	More Info
UCM Update Financial Account Contact Privacy	Invoke Method	1	Method Name=AccountContactPrivacyChanged
Refresh	Invoke Method	2	Method Name = GetAddress

Table 123 lists Update Account Type, Run-time Event.

Table 123. Update Account Type, Run-time Event

Sequence	Object Name	Event	Subevent	Conditional Expression	Action Set Name
1	FINCORP Account	SetFieldValue	Type	(GetProfileAttr('Haley Message') IS NULL OR GetProfileAttr('Haley Message')<>'Y')	UCM Set FinancialAccountTypeChanged To Y
2	FINCORP Account	WriteRecordUpdated		GetProfileAttr('FinancialAccountTypChanged')='Y	'UCM Update Financial Account Type
1	FINCORP Account	WriteRecordNew		GetProfileAttr('FinancialAccountTypChanged')='Y	'UCM Set FinancialAccountTypeChanged To N

The runtime event on WriteRecordNew is used to suppress “Update Account Type” action for new Financial Account case.

Table 124 lists Update Account Type, where Action Set = “UCM Set FinancialAccountTypeChanged To Y” with Actions.

Table 124. Action Set = “UCM Set FinancialAccountTypeChanged To Y”

Name	Action Type	Sequence	More Info
UCM Set Profile Attribute	Attribute Set	1	Profile Attribute= FinancialAccountTypChanged; Set Operator=Set; Value="Y"

Table 125 lists Update Account Type, where Action Set = “UCM Update Financial Account Type” with Actions.

Table 125. Action Set = “UCM Update Financial Account Type”

Name	Action Type	Sequence	More Info
UCM Set Profile Attribute	Attribute Set	1	Profile Attribute= FinancialAccountTypChanged; Set Operator=Set; Value="N"
UCM Update Financial Account Type	Invoke Method	2	Method Name= AccountTypeChanged

Table 126 lists Update Account Type, where Action Set = “UCM Set FinancialAccountTypeChanged To N” with Actions.

Table 126. Action Set = “UCM Set FinancialAccountTypeChanged To N”

Name	Action Type	Sequence	More Info
UCM Set Profile Attribute	Attribute Set	1	Profile Attribute= FinancialAccountTypChanged; Set Operator=Set; Value="N"

Table 127 lists Address Change Update Privacy Setting, Run-time Event

Table 127. Address Change Update Privacy Setting, Run-time Event

Sequence	Object Name	Event	Subevent	Conditional Expression	Action Set Name
1	UCM FINCORP Account Contact Address	SetFieldValue	Personal Address Id	[Address Type] = LookupValue ('FINS_CF_ADDR_USE_TYPE', 'Garage') AND ([Contact Privacy Mod Number]<1 OR [Contact Privacy Mod Number] IS NULL)	UCM Set GarageAddressChanging To Y
1	UCM FINCORP Account Contact Address	WriteRecordUpdated		(GetProfileAttr('Haley Message') IS NULL OR GetProfileAttr('Haley Message')<>'Y') AND GetProfileAttr('GarageAddressChanging')='Y	'UCM Refresh Record On FINCORP Account Contact Address
2	UCM FINCORP Account Contact Address	WriteRecordUpdated		[Address Type] = LookupValue ('FINS_CF_ADDR_USE_TYPE', 'Garage') AND (GetProfileAttr('Haley Message') IS NULL OR GetProfileAttr('Haley Message')<>'Y') AND [Contact Privacy Mod Number]>0	UCM Set GarageAddressChanged To Y
1	UCM FINCORP Account Contact Address Source Data and History	WriteRecordNew		GetProfileAttr('GarageAddressChanged')='Y' AND [UCM Type Code] = LookupValue('UCM_SDH_TYPE_MLOV', 'History')	UCM Update Financial Account Contact Address

Table 128 lists Address Change Update Privacy Setting, Action Set = “UCM Set GarageAddressChanging To Y” with Actions.

Table 128. Action Set = “UCM Set GarageAddressChanging To Y”

Name	Action Type	Sequence	More Info
UCM Set Profile Attribute	Attribute Set	1	Profile Attribute= GarageAddressChanging; Set Operator=Set; Value="Y"

Table 129 lists Address Change Update Privacy Setting, Action Set = “UCM Refresh Record On FINCORP Account Contact Address” with Actions.

Table 129. Action Set = “UCM Refresh Record On FINCORP Account Contact Address”

Name	Action Type	Sequence	More Info
UCM Refresh Record	Invoke Method	1	Method Name= GetPrivacy

Table 130 lists Address Change Update Privacy Setting, Action Set = “UCM Set GarageAddressChanged To Y” with Actions.

Table 130. Action Set = “UCM Set GarageAddressChanged To Y”

Name	Action Type	Sequence	More Info
UCM Set Profile Attribute	Attribute Set	1	Profile Attribute= GarageAddressChanged; Set Operator=Set; Value="Y"

Table 131 lists Address Change Update Privacy Setting, Action Set = “UCM Update Financial Account Contact Address” with Actions.

Table 131. Action Set = “UCM Update Financial Account Contact Address”

Name	Action Type	Sequence	More Info
UCM Set Profile Attribute	Attribute Set	1	Profile Attribute= GarageAddressChanged; Set Operator=Set; Value="N"
UCM Refresh Record	Invoke Method	2	Method Name=GetFinancialAccount
UCM Update Financial Account Contact Address	Invoke Method	3	Method Name= AccountContactAddressChanged

Step 2 is used to get the parent Financial Account Id and pass to the workflow, so that it can query by this Id and construct the request message to rule engine.

Table 132 lists Retrieve Privacy Setting when contact is inactivated, Run-time Event

Table 132. Retrieve Privacy Setting when contact is inactivated, Run-time Event

Sequence	Object Name	Event	Subevent	Conditional Expression	Action Set Name
1	FINCORP Account Contact	SetFieldValue	Status	[Status] = LookupValue('CONTACT_STATUS', 'Inactive') AND (GetProfileAttr('Haley Message') IS NULL OR GetProfileAttr('Haley Message')<>'Y')	UCM Set ContactInactivated To Y
1	FINCORP Account Contact	WriteRecordUpdated		GetProfileAttr('ContactInactivated')='Y	'UCM Contact Inactivated

Table 133 lists Retrieve Privacy Setting when contact is inactivated, Action Set with = “UCM Set ContactInactivated To Y” with Actions.

Table 133. Action Set with = “UCM Set ContactInactivated To Y”

Name	Action Type	Sequence	More Info
UCM Set Profile Attribute	Attribute Set	1	Profile Attribute=ContactInactivated; Set Operator=Set; Value="Y"

Table 134 lists Retrieve Privacy Setting when contact is inactivated, Action Set with = “UCM Contact Inactivated” with Actions.

Table 134. Action Set with = “UCM Contact Inactivated”

Name	Action Type	Sequence	More Info
UCM Set Profile Attribute	Attribute Set	1	Profile Attribute=ContactInactivated; Set Operator=Set; Value="N"
UCM Contact Inactivated	Invoke Method	2	ContactInactivated

9 Privacy Management Solution

This chapter includes the following sections:

- "About Privacy Management" on page 237
- "Privacy Vocabulary" on page 238
- "Sample Privacy Rules" on page 241
- "Customization Methods" on page 244
- "About Blank Privacy Source Data and History Records" on page 248

About Privacy Management

This chapter describes how the privacy vocabulary and sample rules included with the Siebel Universal Customer Master (UCM) Privacy Management Solution can be tailored to meet your organization's requirements. The UCM Privacy Management Solution comes preconfigured with a vocabulary and sample rules designed to meet the requirements of financial services companies doing business in the United States. It also includes concepts that are useful to businesses managing compliance with fair information practices including notice and choice as well as global accepted privacy principles such as use limitation (purpose), disclosure (recipients), data collection limitation (data categories), and others.

In particular, the UCM Privacy Management Solution business rules environment enables business analysts to model, test, and deploy rules that manage privacy information stored in the UCM related to financial accounts and contacts. For more information, read *Siebel Business Rules Administration Guide*.

Role of HaleyAuthority in the Siebel Universal Customer Master Privacy Management Solution

HaleyAuthority 5.2 provides a platform for users to develop new or customize existing business rules as well as the testing capabilities. It is seamlessly integrated into the UCM Privacy Management Solution with two Siebel plug-ins: object importer and deployment wizard. The object importer enables you to select a subset of the Siebel object model and import it into HaleyAuthority to create concepts and relations. The deployment wizard guides you in deploying a rule module in and storing it in runtime tables. At runtime, requests are made to the business rule runtime to evaluate a single financial account and its related entities (contacts, addresses, privacy data, and so on) and the business rule runtime responds with the results of its rules evaluation.

Business Rules Architecture

Company business rules and policies are developed and tested within the HaleyAuthority. The sample business rules are organized by privacy related events. You may implement your business-specific rules in separate modules as well as tailor the existing sample rules. The conceptual role model contains the vocabulary of the privacy rules—the concepts, entities and values, the dictionary, the relations between concepts, the phrasing of those relations, the definitions of tables, and the standard instances of the concepts and relations (such as state codes, cover letter, and privacy notice type codes, and so on).

Privacy Vocabulary

The Siebel UCM Privacy Management Solution business rules are developed using English language sentences. These sentences are put together using concepts (nouns), relations between concepts, phrasing for those relations (verbs, adverbs, adjectives, prepositions and sentence fragments), and table type definitions which have been predefined in the sample knowledge base. Together these form the Conceptual Role Model. This section describes the key components of the Conceptual Role Model that you use and customize as described in "Customization Methods" on page 244.

The sample knowledge base is named Siebel_UCM_PrivacyMgmt.akb and is located in the Tools/Rule/KnowledgeBase directory.

Conceptual Role Model

The sample Conceptual Role Model included with the UCM Privacy Management Solution was developed based on input from several sources: language used in privacy policies across various industries, the privacy vocabulary research summarized the W3C's Platform for Privacy Preferences (P3P) project, and terminology used by some US financial services industry customers of Siebel. The role model is intended to cover a broad set of US financial services requirements as well as a starting point for privacy policy management for other industries and geographies.

You need to modify the conceptual role model to support the mapping of additional elements from UCM, to customize instances of communication message types.

Key Concepts—Entities

Entities are concepts that have attributes. In UCM Privacy Management Solutions, there are two types of entities: Siebel Object Importer-generated entities that come directly from Siebel Business Component definitions; and abstract entities that are built on other entities or value concepts for convenience when using the Authority.

To understand and maintain the Siebel object importer generated entities, refer to *Siebel Business Rules Administration Guide* for detailed instructions. The main entity objects generated by object importer include: fincorp account, fincorp account contact, ucm fincorp account contact address, ucm fincorp account contact garage address source data and history, ucm fincorp account contact privacy, and ucm fincorp account privacy.

The abstract entities in the Siebel UCM Privacy Solution include:

- **Abstract entity—privacy flag.** This is a named collection of privacy attributes for which an option or choice is elected. The default instances of privacy preferences reflect standard elections for US financial services companies: affiliate flag, telemarketing flag, non-affiliate flag, and channel flag (for dealer sharing). Privacy preferences can also be mapped to P3P privacy statements to maintain preferences consistent with P3P privacy policies.
- **Abstract entity—privacy flag source.** This is a flag, election or choice that indicates whether a privacy flag is allowed, denied, or pending, and whether the privacy flag was elected by default or by request based on customer direction. Typically this entity has values of opt in default, opt in request, opt out default, opt out request, pending default and pending request.
- **Abstract entity—state code.** This indicates which state out of all the states that come with all the financial account contacts on the financial account is the most restrictive state for that financial account. Some states such as California, Puerto Rico and Vermont, increase the privacy regulatory requirements set by the Graham-Leach-Bliley Act or FCRA, they are listed as separate instances. The other states are listed as one generic instance. For any state code other than the fifty states or DC or Puerto Rico, this code is set to XX. This is a common practice for managing various U.S. state legislation and policy. For example, customer can have 1 state code corresponds to 1 state, or 1 state code corresponds to all states under the same federal law such as GLBA law.

Key Concepts—Values

Value concepts are created by Siebel object importer, based on Siebel object model business component field definitions. Please refer to Siebel object importer documents for detailed instructions. To simplify handling of strings in tables, by convention string instances generally have the same instance name and string value.

Relations and Phrasings

Siebel object importer can create entity or value concepts, as well as establish the relations between these two and generate simple phrasings such as “an entity has a value” and so on. On top of it, Siebel UCM Privacy Solution also contains a rich library of privacy related complicated relations and phrasings.

Tables

The following tables are used in the Siebel UCM Privacy Solution.

Table Name		Table Details
Privacy Option With Source Ranking		
	Purpose	This table maintains the relative priority of privacy flag sources when rolling up privacy elections to the financial account level. Note that opt out request must always be ranked number 1. This table uses the entity, privacy flag source, and a manually created integer value concept, rank.
	Module	Privacy option and source hierarchy
	Standard phrasing to retrieve values	a privacy flag source has a rank
State Code Cover Letter Table and State Code Notice Type Table		
	Purpose	These tables stores the cover letter and privacy notice type codes based on the state code and owner type (communication message type).
	Module	CUST.1 Owner type tables
	Standard phrasing to retrieve values	a state code with an owner type determines a (cover letter | privacy notice sent) type
(owner type) State Code Change Cover Letter Table		
	Purpose	These tables store the cover letter type codes based on the original state code and the new state code. They are instances of the same table type and the correct table is selected by the applicability condition on the parent module based on the owner type.
	Module	CUST.2 Address changes tables
	Standard phrasing to retrieve values	A cover letter type is determined after a second state code changes from a first state code
State Restrictiveness Ranking Table		
	Purpose	This table ranks states by their restrictiveness and is used to determine which state must be the privacy state code for a financial account when there are multiple states that come with all the financial account contacts on the financial account
	Module	US. Country rules
	Standard phrasing to retrieve values	a state has a rank

Table Name		Table Details
Default Privacy for (privacy flag)		
	Purpose	These tables determine the default privacy flag options for each privacy flag based on the state code.
	Module	US.FS.1 Default privacy option tables
	Standard phrasing to retrieve values	a state determines a (privacy flag)

Sample Privacy Rules

The top-level modules are grouped by privacy events and the type of rules in the module (core rules as compared to specific business rules). In addition, modules are separated by order of evaluation because only top-level modules can have priorities.

The following are the major modules:

- "Account Type Change Event module" on page 241
- "Consumer Address Change Event module" on page 241
- "Contact Inactivated Event module" on page 242
- "Core Privacy Rules module" on page 242
- "Generate Output module" on page 242
- "Initialization module" on page 242
- "Input Interpretation module" on page 243
- "New Financial Account_New Contact Event module" on page 243
- "Pending Period Expired Event module" on page 244
- "Reevaluate Privacy module" on page 244

Account Type Change Event module

Priority: 6

Purpose: This module contains rules that apply when a fincorp account type is changed from Consumer to Business or the other way.

Consumer Address Change Event module

Priority: 8

This module contains rules that apply when a fincorp account contact of a consumer type of fincorp account moves to a new state. Certain sub-modules are applicable only when this causes a change to the privacy state code for the fincorp account.

Contact Inactivated Event module

Priority: 0

Purpose: This module contains rules that apply when a fincorp account contact is inactivated from a fincorp account. The sample rules specific to Contact Inactivated Event are included in the sub-module "Customer Rules". The other sub-modules are actually a copy of the core modules including "Input interpretation", "Initialization", "Core Privacy Rules", "Generate Output" and "Reevaluate privacy". This is because "Contact Inactivated Event" is for both active and inactive contacts; whereas the other events are for active contacts only. They can not share the same core modules.

Core Privacy Rules module

Priority: 5

Purpose: This module contains the core rules that generally apply to all the evaluation requests.

- **CUST. Customer rules.** This module contains customer-specific core rules. The sample rule provided is for the consumer type of customers, which forces the channel partner sharing preference elections to be identical to the non-affiliate sharing preference elections for consumers outside of California who have a channel partner type of Brand X or Brand Y.
- **US.FS. Evaluate core privacy rules.** This module contains core rules that generally apply to all US Financial Services customers for any evaluation request.
- **US.FS.1. Determine privacy state code.** This module contains statements that set the privacy state code values for a business or consumer account and determine whether it has changed from a previous value.
- **US.FS.2. Determine precedence of privacy options and sources.** This module contains statements that use the privacy option with source hierarchy to determine the roll up of privacy preferences to the financial account level.
- **US.FS.3. Handle privacy notices and letters.** This module contains statements that determine whether a contact receives an annual notice and what cover letter and notice must be sent to the customer, except in the case where the contact has moved.

Generate Output module

Priority: 0

Purpose: This module contains the rules that must evaluate after the main body of business rules and be used to set values on the privacy record to be output to UCM. The sample rules provided set value to a concept only when the concept value need to be changed after the evaluation of the main business rules.

Initialization module

Priority: 10

Purpose: This module contains business rules including definitions and tables of default values:

- **CUST. Customer rules.** This module contains initial business rules and table instances that are customer-specific. This module includes table instances of cover letter types and privacy notice types based on privacy state code and owner type (communication message type) and the cover letter types based on state address change and owner type.
- **Expired Pending Period.** This module contains a business rule to determine that a pending period has expired.
- **Initial values.** This module contains initial business rules and table instances that are not country or industry-specific. Includes the table instance establishing the hierarchy of privacy option with sources (Opt In Request < Opt In Default < Pend < Opt Out Default < Opt Out Request).
- **US. Country rules.** This module contains initial business rules and table instances that apply to US companies (for example, a rule establishing the most restrictive state for each contact). This module includes a table instance ranking the restrictiveness of US states. This module contains some examples and please consults with your company's consumer privacy best practices and tailors these samples rules accordingly.
- **US.FS. Financial Services industry rules.** This module contains initial business rules and table instances that apply to US financial services industry companies. These rules are most commonly managed in financial industries; however, depending on your company's asset and privacy policy, you may reuse these rules or tailor to your asset need. Note: FINCORP is a Siebel business component and corresponds to 'S_ASSET' the underlying Siebel schema. If you would like to reuse these rules but for your industry-specific business component based on S_ASSET. For details, please read "Customization Methods" on page 244.
- **US.FS.1. Default privacy option tables.** This sub-module includes tables of default privacy options for each privacy flag based on state code.

Input Interpretation module

Priority: 11

Purpose: This module contains rules that interpret input data and determine key privacy related facts:

- Annual notice requirement
- Contact Delete
- Determine garage address state
- Determine primary fincorp account contact
- Privacy State Code

New Financial Account_New Contact Event module

Priority: 9

Purpose: This module contains the rules that apply when a new fincorp account or a fincorp account contact is added.

- **Determine New financial account or New financial account contact.** This module contains the rules that decide on the event object type: whether it is for a new fincorp account or for a new fincorp account contact.
- **New contact.** This module contains the rules that apply for a new fincorp account contact case. Business and Consumer types of fincorp account have different privacy regulations further specified in two separate submodules.
- **New financial account.** This module contains the rules that apply for a new fincorp account case.

Pending Period Expired Event module

Priority: 0

Purpose: This module contains rules that apply when the pending period for the Consumer type fincorp account expires (such as setting the default privacy options for each privacy preference based on the privacy state code).

Reevaluate Privacy module

Priority: 0

Purpose: This module contains the statements that interpret and recognize the input privacy options with sources on fincorp account contact privacy level. By executing rules in this module, all customer requested flag values are preserved and not just limited to OUT Requested.

Customization Methods

The following customization methods are covered in this section:

- "Customization Strategies" on page 244
- "Required Customizations" on page 245
- "Customization Scenarios" on page 246

Customization Strategies

The sample rules for the UCM Privacy Management Solution are intended to be a template for a customer-specific implementation of their own business policies. As such, they can be modified, simplified, translated, replaced, or used as a design model as dictated by customer needs.

However, by following some consistent guidelines on customizing the sample knowledge base, the process of integrating future releases of the solution can be simplified.

1. Insert rules into the existing CUST customer specific modules in the sample knowledge base whenever practical. The sample CUST modules contain rules and tables that either require customization for each customer or are intended to isolate customizations from the prepackaged solution. To the extent that these customizations involve rules specific to multiple jurisdictions, the jurisdiction prefix must be included in the modules.
2. Customer installations require customization of instances such as owner type, Fincorp Account type/customer type, and so on. If the instances are different, then the changes can be done in the knowledge base only. To the extent that further modifications to entities or relations in the Conceptual Role Model are required, then these modifications must be done in the Siebel repository and copied to the knowledge base using the Siebel Object Importer. The importer does not remove the old concepts and relations, as it is better further identify them using the Version custom property or a Comment so that they can be easily located later.
3. Modifications to the “standard” sample rules are necessary at times. You must your own rules within new customer-specific modules identified with an appropriate prefix. The modified or deleted sample rule can easily be retailed for history as a draft rule by inserting non-recognized text at the beginning or by changing the effective date of the rule to be expired. In addition, using the custom Version property provides an additional means of tagging the customization.

By following these guidelines, it is possible to identify the customizations if a customer desires to integrate capabilities of future releases of the knowledge base.

Required Customizations

The following customizations are absolutely required for any new installation of the UCM Privacy Management Solution:

- “CUST. Customer Rules” on page 245
- “Owner Types and Cover Letter Types” on page 245

CUST. Customer Rules

The sample knowledge base has sample rules in this module that force the channel flag to be identical to nonaffiliate flag for consumers outside of CA who are customers of Brand X or Brand Y. You must customize these conditions to your business policies for their channel flags or other criteria or delete these statements.

Owner Types and Cover Letter Types

You must determine what owner types to use, if any. UCM default has the Brand X and Brand Y as sample owner type values. If the customer has an owner, channel partner, or dealer and these affect the version of the cover letters or privacy notices, then you must make these changes:

- Modify the instances of the owner type based on the values used in UCM.
- Edit the State Code Cover Letter Table and State Code Privacy Notice Table column headings to match the new owner types and modify the table entries to reflect the customer's cover letter type codes and privacy notice type codes.

- Edit the sub-module names and State Change Cover Letter Table instance names in CUST.2 Address changes tables to reflect the new owner types. Add additional modules and table instances as necessary if more than two communication message types are used. Edit the table entries to reflect the appropriate cover letter type codes for state changes for your company.

If you have only a single version of the cover letters or privacy notices (other than state-specific versions), then you must:

- Add an instance to the concept owner type with name = standard and value = “standard” and delete the Brand X and Brand Y.
- Edit the State Code Cover Letter Table and State Code Privacy Notice Table column headings to have a single communication message type = standard and modify the table entries to reflect the customer's cover letter type codes and privacy notice type codes.
- Change the sub-module name of CUST.2.1. BrandX tables to CUST.2.1. Standard tables and change the name of the Brand X State Change Cover Letter Table instance to Standard State Change Cover Letter Table instance. Edit the table entries to reflect the appropriate cover letter type codes for state changes for your company.

Customization Scenarios

Review the following scenarios to see if they apply to your implementation:

- “Capture Privacy Elections for a New Business Channel” on page 246
- “Change Address for Privacy Preferences” on page 247
- “Implementing a New Company Policy to be Compliant with OFAC” on page 247
- “Modifying New State Default Privacy Options upon Pending Expiration” on page 247
- “Modifying the State Restrictiveness Ranking Table” on page 248
- “New Business Components in Siebel UCM, Reuse Module or Rules in Privacy Knowledge Base” on page 248

Capture Privacy Elections for a New Business Channel

Scenario: You want to capture privacy elections for a new channel added to the business, for example: Insurance.

Solution: Adding a new privacy preference involves several changes to the business rules after the privacy preference is added in Siebel table and repository. Make these changes:

- Run the Siebel Object Importer to bring the change into knowledge base conceptual role model. New concepts and relations are generated automatically.
- Add the instances for the privacy preference to the privacy preference concept Insurance.
- Add a new table with the default privacy preference values for insurance, copying the format of the existing Default Privacy Option Tables.
- Add business rules for this new privacy preference.

Change Address for Privacy Preferences

Scenario: You want to evaluate the privacy preferences based on addresses other than Garage Address, for example: Permanent Address. In such a case, both the Siebel configuration and HaleyAuthority rules need to be changed.

Solution 1: If you do not insist on consistency in Concept/Entity naming, the changes needed are minimal. In Siebel UCM, make these changes:

- Add the LOV “Permanent” to the Address Type field for FINCOPR Account Contact Address.
- Change business component 'UCM FINCORP Account Contact Garage Address Source Data and History' search spec from:

  ```
  [Address Type] = 'Garage' AND [UCM Type Code] = 'History' AND [Parent Type] =
  'Financial Account Contact'
  to:
  [Address Type] = 'Permanent' AND [UCM Type Code] = 'History' AND [Parent Type] =
  'Financial Account Contact'
  ```

- Modify the runtime event triggering to look for changes on Permanent Address instead of Garage Address.

In HaleyAuthority, search for the word “Address Type”, replace “Garage” with “Permanent” in all the statements found, then deploy the changed modules.

Solution 2: If you care about the wording in Siebel and Haley Concept/Entity/Rules, then additional effort is needed as follows:

- Make the changes specified in Solution 1.
- Create a new business component.
- Modify the business object and integration object definitions.
- Modify the rules administration configuration.
- Run the Siebel object importer to bring the new concept and relations into HaleyAuthority.
- Build new rules based on the existing rules for Garage Address.
- Create a new module to hold the new rules.

Implementing a New Company Policy to be Compliant with OFAC

Scenario: You pass a new company policy to be compliant with OFAC list, and privacy rules need to be created around OFAC.

Solution: The existing field exists in both Siebel repository and Haley Conceptual model, so there is no need to run Siebel Object Importer again. Add a new sub-module under the corresponding event module, and then add the relevant business logic.

Modifying New State Default Privacy Options upon Pending Expiration

Scenario: You want the default privacy options to be set for each customer based on the privacy state code for each preference.

Solution: Modify table entries in the Default Privacy Option Table in US.FS.1 Default privacy option tables. Entries for additional state regulations can be added by inserting additional rows in this table. Note that XX represents “invalid states”—state codes that do not apply to the 50 US states, DC, or Puerto Rico.

Modifying the State Restrictiveness Ranking Table

Scenario: You want to alter the ranking of states used to determine the privacy state code for the financial account.

Solution: Modify the priority of the states in the State Restrictiveness Ranking Table in US. Country rules or add additional states with priorities. Note that if Puerto Rico has the same rank as another entry, Puerto Rico has a rule that makes it take precedence in module US. Country rules.

New Business Components in Siebel UCM, Reuse Module or Rules in Privacy Knowledge Base

Scenario: FINCORP is a standard business component and corresponds to 'S_ASSET'. You want to reuse the rules related to FINCORP Account, but use your industry-specific business component based on S_ASSET.

Solution: Make these modifications:

- Run Siebel Object Importer to bring the new business components into knowledge base conceptual role model. New concepts and relations are generated automatically.
- Change the rules you want to reuse to use the concepts/relations you just imported (or copy the original rules instead of changing them).
- Reparse the rules.

About Blank Privacy Source Data and History Records

Blank Privacy History records may be created in the database. This is because external applications send financial account insert requests to UCM. To evaluate financial account/financial account contact's privacy preferences, the external application is required to send in a privacy record along with the financial account/financial account contact insert request. The source is captured in the UCM transaction manager. If there is no privacy preference preselected, the privacy preferences in the source record are empty. This record in the source data and history (SDH) table is considered a “blank” record.

If you do not preselect privacy preferences before sending information to the rule engine, then the the blank record is captured as history. Even though there is no privacy preference in this history record, internally the record does have some other values, such as a foreign key to its parent record type. There are two reasons why this occurs:

1. S_PRIVACY is a separate table. Creating a financial account (FA) or financial account contact (FAC) does not automatically create a child privacy record.

2. There is no NewRecord action available in the core rule engine to callback Siebel UCM to create a new privacy record. A privacy record must be created before sending to the rule engine even though it is a new FA/FAC event.

Because of these reasons, a blank record exists in the base table and the SDH table. The base table blank record exists for a very short period of time, because the rule engine is triggered. However, blank records in the SDH remain. You may not want to see these records because they do not provide useful information in their privacy management solution.

If you want to remove these blank records, there are several possible solutions. You need to evaluate the situation according to your deployment needs. Be aware that there may ne a performance drawback to removing blank privacy SDH records.

A possible solution is to write a script in business Service UCM FINCORP Account Source Data and History Service. When the Insert method is called, check the values for some required fields in UCM FINCORP Account Privacy Source Data and History, UCM FINCOPR Account Contact Privacy Source Data and History. If the values are empty, delete the record.

To add a script to a business service

1. In Siebel Tools, find business Service UCM FINCORP Account Source Data and History Service.
2. Right click and chose Edit Sever Script.
3. In Service_InvokeMethod, add the following script in eScript (Program Language = JS)
4. Save and compile the changes.

Sample Script to Remove Blank Records

NOTE: You may have different criteria for what constitutes a blank record. Modify the following script accordingly.

```
function Service_InvokeMethod (MethodName, Inputs, Outputs)
{
   if (MethodName == "Insert")
   {
      var oBusObj = TheApplication().GetBusObject("UCM FINCORP Account Source Data and
History");
      var oBusCompFAP = oBusObj.GetBusComp("UCM FINCORP Account Privacy Source Data and
History");
      var oBusCompFACP = oBusObj.GetBusComp("UCM FINCORP Account Contact Privacy Source
Data and History");

      with (oBusCompFAP)
      {
         ClearToQuery();
            ActivateField("Affiliate Flag");
            ActivateField("Nonaffiliate Flag");
            ActivateField("Telemarketing Flag");
            ActivateField("Channel Flag");
            SetSearchSpec("Affiliate Flag", "IS NULL");
            SetSearchSpec("Nonaffiliate Flag", "IS NULL");
            SetSearchSpec("Telemarketing Flag", "IS NULL");
```

```
            SetSearchSpec("Channel Flag", "IS NULL");
            ExecuteQuery(ForwardBackward);
            while(FirstRecord())
            {
               DeleteRecord();
               NextSelected();
            }
         }

         with (oBusCompFACP)
         {
            ClearToQuery();
               ActivateField("Affiliate Flag");
               ActivateField("Nonaffiliate Flag");
               ActivateField("Telemarketing Flag");
               ActivateField("Channel Flag");
               SetSearchSpec("Affiliate Flag", "IS NULL");
               SetSearchSpec("Nonaffiliate Flag", "IS NULL");
               SetSearchSpec("Telemarketing Flag", "IS NULL");
               SetSearchSpec("Channel Flag", "IS NULL");
               ExecuteQuery(ForwardBackward);
               while(FirstRecord())
               {
                  DeleteRecord();
                  NextSelected();
               }
         }
      }
   }
```

A Additional License Restrictions

Siebel Master Data Applications are licensed subject to additional license grants and restrictions, see "Additional License Grants and Restrictions Regarding Siebel Master Data Applications." These restrictions supersede all other grants and restrictions.

Additional License Grants and Restrictions Regarding Siebel Master Data Applications

Siebel Master Data Applications are licensed to Customer under the terms of the Agreement applicable to Programs, subject to the following additional grants and restrictions:

- Customer shall have no right to use the user interface layer of the Siebel Master Data Applications without purchase of the Customer Data Steward or Product Data Steward modules;
- Customer shall have the right to access the database containing the Siebel Data Model in any way required including direct SQL access through an external application;
- Customer shall have the right to use the Siebel Data Model solely as a source of customer, product, activity, sales, service, marketing, or field service information and any extensions made through the use of Siebel Tools;
- Customer shall have the right to use the Siebel Tools to configure the Siebel Master Data Applications solely in accordance with the Documentation to configure only those Business Objects and Business Components (BusComps) included with the licensed Siebel Master Data Applications (as set forth in Appendix 1) through the Object Manager and Enterprise Application Integration (EAI), Application Services Interfaces (ASIs), Enterprise Integration Manager (EIM), and Business Process Designer.
- Customer shall have the right to use System Administration solely in accordance with the Documentation including Business Process Designer and EIM administration and EAI configuration;
- Customer shall have the right to have programmatic access only to those Business Objects and BusComps included with the licensed Siebel Master Data Applications, solely through Object Manager and EAI, EIM, and Business Process Designer.
- Notwithstanding anything to the contrary, Customer shall have no right to (i) use screens/views/applets or the Siebel user interface layer; (ii) extend or modify the Siebel Data Model other than through Siebel Tools; (iii) use the tables in the Siebel Data Model for any Siebel Programs not licensed by Customer (for example, Siebel PRM); (iv) use the Siebel Tools for general application development purposes, or (iv) use or access the Siebel Master Data Applications (including the Siebel Data Model) for Siebel non-CRM (Customer Relationship Management) applications.

- Customer will not use this Siebel Master Data Applications licensing model to avoid paying Program License Fees for (1) full use of the functionality contained in other Siebel programs other than the Siebel Master Data Applications, or (2) the ability to use the Siebel Master Data Applications Programs outside of the specific license grants listed above.

Table 135 sets forth the Business Components included in Siebel Master Data Applications.

Table 135. Business Components Included in Siebel Master Data Applications

Master Data Module	Business Components
Siebel Universal Customer Master	Party
	Party Contact
	Party Relationship To
	Account
	Contact
	Contact Relationship
	Household
	CUT Address
	Personal Address
	Channel Partner
	Partnership Contact
	Opportunity
	Position
	Employee
	Expense
	FINCORP Account (Asset)
	FINCORP Deposit Account
	FINCORP Loan Account
	FINS AG Agent
	FINS All Policy Details
	FINS Group Policy
	FINS Household Policies (Read Only)
	FINS Life Policy
	Household FIN Accounts (Read Only)
	INS All Policies
	INS Policy (Asset)

Table 135. Business Components Included in Siebel Master Data Applications

Master Data Module	Business Components
Siebel Universal Activity Master	Action
	Contact Note
	Contact Private Note
	Personal Payment Profile
	Account Profile Attributes
	Account Credit Profile
	Account Note
	Account Private Note
	Account External Product
	Asset Mgmt (Asset)
	Sub Account
	Entitlement Account
	Entitlement Contact
	Service Agreement
	Coverage Attributes
	Partner Certification
	Action Copy
	Coverage Attributes
	Household Action (Read Only)
	Household Contact Note (Read Only)
	Household Service Agreement (Read Only)
	Partner Certification Criteria
	Partner Product
	Product Line
	FINCORP Client Contact Management
	FIN Contact Income
	FIN Contact Employment
	FINS Financial Statement Contact
	FINS Financial Accounts for Contacts
	FINCORP Contact Education
	FINCORP Contact Experience

Table 135. Business Components Included in Siebel Master Data Applications

Master Data Module	Business Components
Siebel Universal Activity Master (continued)	FINCORP Investor
	FINCORP Transaction
	FINS AG Agency's Contracts
	FINS AG Agency's Errors And Omissions
	FINS AG Agency's Licenses And Appointments
	FINS AG Agent's Contracts
	FINS AG Agent's Errors And Omissions
	FINS AG Agent's Licenses And Appointments
	FINS AG Agent's NASD Registration
	FINS Billing Accounts
	FINS Billing Accounts (No Contact Copy)
	FINS Contact Household Contact
	FINS Household Billing Accounts (Read Only)

Table 135. Business Components Included in Siebel Master Data Applications

Master Data Module	Business Components
Siebel Universal Product Master	Complex Product
	Internal Product
	Internal Product Attributes
	Internal Product Attachment
	Internal Product Image
	Internal Product Translation
	Component Product
	Price List
	Price List Item
	Price Book
	Price Book Attributes
	Price Book Attribute Values
	Price Book Groups
	Admin Product Line
	Admin Product Line Fee
	Consumer Product
	Catalog Category
	Product Entitlement Details
	Key Feature
	Admin Product Comparison
	Sales Tool by Product
	XA Attribute Value
	Cost List Item
	Product Defect
	Partner Product
	Revenue

Table 135. Business Components Included in Siebel Master Data Applications

Master Data Module	Business Components
Siebel Universal Sales Master	Opportunity
	Opportunity (Orders)
	Competitor
	Competitor 2
	Quote
	Quote Attachment
	Quote Item
	Orders
	Order Entry (Orders)
	Order Entry (Line Items)
	Order Entry (Order Terms)
	Decision Issue
	Decision Issue Attachment
	Opportunity Forecast
	Opportunity Forecast Detail (Wide)
	Opportunity Forecast Line
	Opportunity Person Forecast
	Opportunity Product Forecast

Table 135. Business Components Included in Siebel Master Data Applications

Master Data Module	Business Components
Siebel Universal Service Master	Service Request
	Related Service Request
	Solution
	Related Solution
	Resolution Activity
	SR Resolution Item
	RMA
	Service Agreement Account
	Service Agreement
	Service Agreement Contact
	Service Agreement Product
	Service Agreement Attachment
	Product Defect
	Decision Issue
	Customer Survey
	Customer Product
	Activity Plan
	Related Service Request
	FIN Service Request
Siebel Universal Marketing Master	Campaign
	Campaign Achievement
	Campaign Contact
	Campaign Groups
	Campaign List Contact
	Campaign Lists
	Campaign Occurrence Offer
	Campaign Recipient
	Prospect
	Contact-Prospect Campaigns
	Prospect Partner
	Prospect Recipients

Table 135. Business Components Included in Siebel Master Data Applications

Master Data Module	Business Components
Siebel Universal Marketing Master	Prospect Substitution
	List Management
	List Mgmt List Member Prospect
	List Mgmt Prospective Contact
	List Mgmt Prospective Contact Mkt Segment
	List Mgmt Prospective Contact (Attributes)
	LOY Attribute Definition
	LOY Household
	LOY Loan Repayment
	LOY Member
	LOY Member Tier
	LOY Membership Card
	LOY Point Block
	LOY Product Points
	LOY Program
	LOY Promotion
	LOY Promotion Bucket
	LOY Statement
	LOY Tier
	LOY Tier Class
	LOY Transaction
	LOY Transaction Accrual Item
	LOY Transaction Redemption Item
	LOY Voucher
	Offer
	Response
	Segments
	Programs
	Program Container
	Program Container Occurrences
	Program Occurrences

Table 135. Business Components Included in Siebel Master Data Applications

Master Data Module	Business Components
Siebel Universal Field Service Master	FS Activities (Time, Expense, Material, Skills, Steps, Instructions)
	Service Agreement
	Entitlement
	Warranty
	Asset Management
	Asset Measurement
	Preventive Maintenance
	Invoice
	Inventory
	Order entry
	Cycle Counting
	Shipping
	Receiving
	Replenishment
	Repair
	FS Price List
	FS Product
	Scheduling
Siebel Universal Identification Master	Contact
	CIF Contact Reference
	Account
	CIF Account Reference

The Siebel Master Data Applications are licensed based on:

- The number of unique Contact Records which the customer may store in the Universal Customer Master.
- The number of Application Instances integrated to the Siebel Universal Customer Master.

A Contact Record is a database record of an individual contact that is stored in the Siebel Data Model. For those customers who cannot identify or otherwise count the number of individual contacts for which the application will provide master functionality, on an exception basis the products may be priced and licensed on the basis of the number of account records (records in the Siebel Data Model) in the Siebel Universal Customer Master deployment, using a ratio that assumes 20 contact records for each account record. Approval is required to price and license on this basis.

An Application Instance is each individual installation of an application. All Application Instances that meet the following criteria are counted as Application Instances that are integrated to the Siebel Master Data Applications:

- Access information from Siebel Master Data Applications as a source of information.
- Send information to Siebel Master Data Applications.
- Receive information from Siebel Master Data Applications.

Such Application Instances can be integrated to Siebel Master Data Applications by means of direct connection, integration server, a messaging bus, flat file, or any other means of real-time, near real-time or batch integration. Application Instances used solely for inter-application communication, such as IBM MQ Series, are not counted as Application Instances integrated to Siebel Master Data Applications.

The Siebel Universal Product Master Applications are licensed based on:

- The number of unique Product Records which the customer may store in the Universal Product Master.
- The number of Application Instances integrated to the Siebel Universal Product Master.

A Product Record is a database record of an individual product that is stored in Oracle's Siebel Data Model.

Index

A

Access Control, enforcing 23
Account Activity messages
about 181
Account Activity Delete Integration Components 183
Account Activity Insert Integration Components 182
Account Activity Integration Component Fields 183
Account Activity Query by Name Integration Components 182
Account Activity Query Integration Components 181
Account Activity Update Integration Components 182
Account Address messages
about 176
Account Address Delete Integration Components 178
Account Address Insert Integration Components 177
Account Address Integration Component Fields 178
Account Address Query by Name Integration Components 177
Account Address Query Integration Components 176
Account Address Update Integration Components 177
Account messages
about 201
Account Delete Integration Components 203
Account Insert Integration Components 202
Account Integration Component Fields 203
Account Query by Name Integration Components 202
Account Query Integration Components 201, 215
Account Update Integration Components 202
Account Product messages
about 188
Account Product Delete Integration Components 189
Account Product Insert Integration Components 189
Account Product Integration Component Fields 190
Account Product Query by Name Integration Components 188
Account Product Query Integration Components 188
Account Product Update Integration Components 189
Account Profile messages
about 172
Account Profile Integration Component Fields 174
Account Profile Query by Name Integration Components 172
Account Profile Query Integration Components 172
Account Profile Update Integration Components 173
Account Reference record, creating 84
Account Relationship messages
about 194
Account Relationship Delete Integration Components 195
Account Relationship Insert Integration Components 195
Account Relationship Integration Component Fields 195
Account Relationship Query by Name Integration Components 194
Account Relationship Query Integration Components 194
Account Relationship Update Integration Components 194
administering Siebel UCM Applications 23
administration
Account Reference record, creating 84
deleting a registered system 69
registering systems connected with UCM applications 67
Siebel UCM Application integration objects, about 62
Siebel UCM Application integration objects, modifying 62
Siebel UCM Applications integration services 40
system privileges, setting 69

System Publish and Subscribe, configuring 73
UCM dispatcher map integration objects, about 63
UCM dispatcher map integration objects, viewing user properties 65
UCM envelope integration objects, modifying 62
architecture
about 25
about and diagram 22
administering Siebel UCM Applications 23
database tables, functions and services 22
integration services, about 24
Siebel Connector and integration 30
UCM business objects layer 26
UCM Data Objects layer 30
UCM user interface layer 26

B

business objects layer
about 26
object manager and data manager 26
Siebel Industry EAI Architecture 26
business process flow, support of Siebel Industry EAI architecture 28
business services, support of Siebel Industry EAI architecture 30

C

check mark in Subscribe column 71
CIFOperation_GetValue, about and user properties 42
CIFOperation_Query 41
CIFOperation_XMLQuery 41
communication
inbound communication and UCM Converter 31
components 14
Contact Activity messages
about 130
Contact Activity Delete Integration Components 131
Contact Activity Insert Integration Components 131
Contact Activity Integration Component Fields 131
Contact Activity Query by Name Integration Components 130
Contact Activity Query Integration Components 130
Contact Activity Update Integration Components 130
Contact Address messages
about 126
Contact Address Delete Integration Components 127
Contact Address Insert Integration Components 127
Contact Address Integration Component Fields 128
Contact Address Query by Name Integration Components 126
Contact Address Query Integration Components 126
Contact Address Update Integration Components 127
Contact messages
about 143
Contact Delete Integration Components 144
Contact Integration Component Fields 144
Contact Query by Name Integration Components 143
Contact Query Integration Components 143
Contact Update Integration Components 143
Contact Product messages
about 134
Contact Product Delete Integration Components 135
Contact Product Insert Integration Components 135
Contact Product Integration Component Fields 135
Contact Product Query by Name Integration Components 134
Contact Product Query Integration Components 134
Contact Product Update Integration Components 134
Contact Profile messages
about 122
Contact Profile Delete Integration Components 123
Contact Profile Insert Integration Components 123
Contact Profile Integration Component Fields 124
Contact Profile Query by Name Integration Components 122
Contact Profile Query Integration Components 122
Contact Profile Update Integration Components 123
Contact Relationship messages
about 138
Contact Relationship Delete Integration Components 139

Contact Relationship Insert Integration Components 139
Contact Relationship Integration Component Fields 139
Contact Relationship Query by Name Integration Components 138
Contact Relationship Query Integration Components 138
Contact Relationship Update Integration Components 138
CRMML standards
about and XML messages 24
body section 118
CRMML syntax and rules 30
envelope and header section 116
Siebel Industry EAI architecture, support of 30
Cross Reference messages
about 215
customer identification
Account Reference record, creating 84
customer information
Siebel Universal Activity Master, about using 12
Customer Relationship Management Markup Language
See CRMML standard

D

D&B information, about 14
Daily Batch publish types
about 32
data manager layer, about 26
database auto-reconnect, about 25
database tables 23
functions and services 22
Delete functionality 22
deleting, a registered system 69
deployment
stand-alone application scenario 20

E

Entity Relationship Designer 14
entity relationship diagrams (ERDs), about creating 14
envelope integration objects, modifying 62
exporting
See Siebel Enterprise Integration Manager

F

fault tolerance, designed for 25
features, new 7
field service
storing organization's field service information 13
framework
administering Siebel UCM Applications 23
architecture and diagram 22
database tables, functions and services 22
integration services 24
Siebel Connector and integration 30
UCM business objects layer 26
UCM Data Objects layer 30
UCM user interface layer 26

G

grants and restrictions 251

H

Household messages
about 154
Household Delete Integration Components 155
Household Insert Integration Components 155
Household Integration Component Fields 156
Household Query by Name Integration Components 154
Household Query Integration Components 154
Household Update Integration Components 155

I

importing
See Siebel Enterprise Integration Manager
inbound communication, and UCM Converter 31
inbound data flow, support of Siebel Industry EAI architecture 28
Inbound HTTP Transport, configuring the Siebel Web Engine 60
inbound messages, and UCM Dispatcher 32
inbound server administration
MQSeries Service Receiver parameters, configuring 57
named subsystem, configuring 56
Insert functionality 22
integration
Siebel Connector, and 30
Siebel EIM and integration 34
Transport Adapter 33
UCM Converter user properties 31
UCM Dispatcher user properties 32

UCM Publish/Subscribe Service methods 32
UCM Security Manager user properties 33
UCM Transaction Manager 33
Integration Object Builder, about 29
integration objects
about 62
modifying 62
Siebel Industry EAI architecture, support of 29
Siebel UCM Applications integration services 40
UCM dispatcher map integration objects 63
UCM dispatcher map integration objects, viewing user properties 65
UCM envelope integration objects, modifying 62
integration services
about and Siebel Connector 24
architecture, about 25

K

key maps
Account Reference record, creating 84

L

license 251

M

marketing
storing organization's marketing information 12
master file modules
See modules
messages
about 122
Account Address 176
application service interfaces supported 109
inbound messages and UCM Dispatcher 32
UCM Account Activity messages 181
UCM Account messages 201
UCM Account Product messages 188
UCM Account Profile messages 172
UCM Account Relationship messages 194
UCM Contact Activity messages 130
UCM Contact Address messages 126
UCM Contact messages 143
UCM Contact Product messages 134
UCM Contact Profile messages 122
UCM Contact Relationship messages 138
UCM Cross Reference messages 215
UCM Household messages 154
UCM Personal Address messages 159
modules
about 11
Siebel Universal Activity Master 12
Siebel Universal Customer Master, about 12
Siebel Universal Customer Master, as base module 9
Siebel Universal Field Service Master 13
Siebel Universal Identification Master 13
Siebel Universal Marketing Master 12
Siebel Universal Product Master 12
Siebel Universal Sales Master 12
Siebel Universal Service Master 13
MQSeries Server Receiver parameter, configuring 57

N

named subsystem, configuring 56
native SQL, and UCM Data Objects layer 30
new features 7

O

object manager layer, about 26

P

Personal Address messages
about 159
FINCORP Account Integration Component Fields 161
FINCORP Account Query Integration Components 161
Personal Address Integration Component Fields 160
Personal Address Query Integration Components 159
UCM FINCORP 161
product components 14
products
storing organization product's information 12
Publish services, about 22

Q

Query functionality 22

R

Real Time publish types
about 32
registering systems
connected with UCM applications 67
deleting a registered system 69
Request and Reply services, about 22
restrictions 251

S

sales
storing organization's sales information 12
scenarios 11
Siebel Universal Customer Master scenario 1 19
service
storing organization's service information 13
Siebel Application Object Manager 14
Siebel Business Application Integration 14
Siebel Business Process Designer 14
Siebel Business Process Designer, about and business data flow 29
Siebel Connector
about and integration 30
Siebel EIM and integration 34
Transport Adapter 33
UCM Converter user properties 31
UCM Dispatcher user properties 32
UCM Publish/Subscribe Service methods 32
UCM Security Manager user properties 33
UCM Transaction Manager 33
Siebel Connector, about using 24
Siebel Connectors
Siebel Industry EAI architecture, support of 26
Siebel EIM, and integration 34
Siebel Enterprise Integration Manager 14
Siebel Entity Relationship Designer 14
Siebel Industry EAI architecture
about 26
business process flow 28
business services 30
CRMML standard 30
inbound data flow 28
integration objects 29
workflow integration 29
Siebel UCM Applications integration objects
about 62
modifying 62
modifying UCM envelope integration objects 62
Siebel UCM Applications integration services
about 40
Siebel Universal Master Application, configuring 60
Siebel Web Engine, configuring to run Inbound HTTP Transport 60
UCM Transaction Manager 41
Siebel Universal Activity Masters 12
Siebel Universal Customer Master
contents of 9
module, about 12
scenario 1 19
Siebel Universal Field Service Master 13
Siebel Universal Identification Master 13
Siebel Universal Marketing Master 12
Siebel Universal Master application
Extension Service, configuring to run 60
Siebel Universal Sales Master 12
Siebel Universal Service Master 13
Siebel Web Engine
Inbound HTTP Transport, configuring 60
SQL, and UCM Data Objects layer 30
stand-alone application scenario 20
Subscribe column, check mark in 71
system privileges, setting 69
System Publish and Subscribe
configuring 73

T

third-party solutions
See Siebel Business Application Integration
Transport Adapter, and Siebel Connector 33

U

UCM Account Activity messages
about 181
Account Activity Delete Integration Components 183
Account Activity Insert Integration Components 182
Account Activity Integration Component Fields 183
Account Activity Query by Name Integration Components 182
Account Activity Query Integration Components 181
Account Activity Update Integration Components 182
UCM Account Address messages
about 176
Account Address Delete Integration Components 178
Account Address Insert Integration Components 177
Account Address Integration Component Fields 178
Account Address Query by Name Integration Components 177
Account Address Query Integration Components 176
Account Address Update Integration Components 177
UCM Account messages
about 201

Account Delete Integration Components 203
Account Insert Integration Components 202
Account Integration Component Fields 203
Account Query by Name Integration Components 202
Account Query Integration Components 201, 215
Account Update Integration Components 202
UCM Account Product messages
about 188
Account Product Delete Integration Components 189
Account Product Insert Integration Components 189
Account Product Integration Component Fields 190
Account Product Query by Name Integration Components 188
Account Product Query Integration Components 188
Account Product Update Integration Components 189
UCM Account Profile messages
about 172
Account Profile Integration Component Fields 174
Account Profile Query by Name Integration Components 172
Account Profile Query Integration Components 172
Account Profile Update Integration Components 173
UCM Account Relationship messages
about 194
Account Relationship Delete Integration Components 195
Account Relationship Insert Integration Components 195
Account Relationship Integration Component Fields 195
Account Relationship Query by Name Integration Components 194
Account Relationship Query Integration Components 194
Account Relationship Update Integration Components 194
UCM business objects layer
about 26
object manager and data manager 26
Siebel Industry EAI architecture 26
UCM Contact Activity messages
about 130
Contact Activity Delete Integration Components 131
Contact Activity Insert Integration Components 131
Contact Activity Integration Component Fields 131
Contact Activity Query by Name Integration Components 130
Contact Activity Query Integration Components 130
Contact Activity Update Integration Components 130
UCM Contact Address messages
about 126
Contact Address Delete Integration Components 127
Contact Address Insert Integration Components 127
Contact Address Integration Component Fields 128
Contact Address Query by Name Integration Components 126
Contact Address Query Integration Components 126
Contact Address Update Integration Components 127
UCM Contact messages
about 143
Contact Delete Integration Components 144
Contact Integration Component Fields 144
Contact Query by Name Integration Components 143
Contact Query Integration Components 143
Contact Update Integration Components 143
UCM Contact Product messages
about 134
Contact Product Delete Integration Components 135
Contact Product Insert Integration Components 135
Contact Product Integration Component Fields 135
Contact Product Query by Name Integration Components 134
Contact Product Query Integration Components 134
Contact Product Update Integration Components 134
UCM Contact Profile messages
about 122
Contact Profile Delete Integration Components 123
Contact Profile Insert Integration Components 123

Contact Profile Integration Component Fields 124
Contact Profile Query by Name Integration Components 122
Contact Profile Query Integration Components 122
Contact Profile Update Integration Components 123
UCM Contact Relationship messages
about 138
Contact Relationship Delete Integration Components 139
Contact Relationship Insert Integration Components 139
Contact Relationship Integration Component Fields 139
Contact Relationship Query by Name Integration Components 138
Contact Relationship Query Integration Components 138
Contact Relationship Update Integration Components 138
UCM Converter
user properties 31
UCM Converter, about and user properties 44
UCM Cross Reference messages
about 215
UCM Data Objects layer
about 30
UCM Dispatcher
about and user properties 45
mapping integration objects, about 63
user properties 32
viewing map integration objects user properties 65
UCM envelope integration objects, modifying 62
UCM FINCORP Account messages 161
UCM Household messages
about 154
Household Delete Integration Components 155
Household Insert Integration Components 155
Household Integration Component Fields 156
Household Query by Name Integration Components 154
Household Query Integration Components 154
Household Update Integration Components 155
UCM Personal Address messages
about 159
FINCORP Account Integration Component Fields 161
FINCORP Account Query Integration Components 161
Personal Address Integration Component Fields 160
Personal Address Query Integration Components 159
UCM FINCORP 161
UCM Publish/Subscribe Service
methods 32
UCM Security Manager
user properties 33
UCM Security Manager, about and user properties 45
UCM Transaction Manager
about and user properties 33
Operation user properties 41
UCM user interface layer 26
universal identification
secure repository for customers' social security numbers 13
Update functionality 22
user interface layer 26

W

workflow integration
support of Siebel Industry EAI architecture 29

X

XML-based applications, about integrating 24

Z

zero downtime, designed for 25