Oracle® Communications Digital Business Experience

Cash to Care Implementation Guide

Release 26.4
G47939-01
May 2026

ORACLE

Oracle Communications Digital Business Experience Cash to Care Implementation Guide, Release 26.4

G47939-01

Contents

About This Content

1 Cash to Care Implementation Overview

About Cash to Care 1
- About the Request-to-Answer Business Process 2
- About the Complaint-to-Solution Business Process 2

2 Request-to-Answer Business Process

Overview of the Request-to-Answer Business Process 1
About Billing Management 2
- Billing Management Flow 2
- Solution Assumptions and Constraints 2

About Query and View Account Balances 3
About View CDR and Detailed Bills (Invoice) 5
About View Unbilled Usage 7
About View and Capture Payments 9
- Capturing Payments 10
  - Capturing Payments in Siebel CRM 10
- Viewing Payment History 11
- Searching for Payments 12

About Account Status Synchronization 12
Implementing the Request-to-Answer Business Process 13
- About Query and View Account Balances 13
  - QueryBalanceSummary Integration Flow 13
  - QueryBalanceGroupList Integration Flow 15
  - QueryBalanceDetails Integration Flow 16
  - QueryBalanceGroupServices Integration Flow 17
  - BRM Interfaces 19
  - Siebel CRM Interfaces 19
  - Industry Oracle AIA Components 19
  - Integration Services 20
- About View CDR and Detailed Bills (Invoice) 21

QueryInvoiceList Integration Flow 22
QueryInvoice Integration Flow 23
QueryInvoiceUsageAllocation Integration Flow 25
SearchInvoiceUsageAllocation Integration Flow 26
QueryInvoiceUsageAllocationResource Integration Flow 28
BRM Interfaces 29
Siebel CRM Interfaces 30
Industry Oracle AIA Components 30
Integration Services 30
About View Unbilled Usage 32
QueryServiceUsage Integration Flow 33
QueryServiceUsageAllocation Integration Flow 34
SearchServiceUsageAllocation Integration Flow 36
QueryServiceUsageAllocationResource Integration Flow 37
BRM Interfaces 39
Siebel CRM Interfaces 39
Industry Oracle AIA Components 39
Integration Services 40
About View and Capture Payments 41
CreateReceivedPayment Integration Flow 42
QueryReceivedPaymentList Integration Flow 43
SearchPayment Integration Flow 45
BRM Interfaces 47
Siebel CRM Interfaces 47
Industry Oracle AIA Components 47
Integration Services 48
Configuring Request-to-Answer Business Process 49
About Configuring Billing Management 50
Setting Up BRM for Integrated Billing Management 50
Setting Up Siebel CRM for Integrated Billing Management 50
Working with DVMs 50
Working with Cross-References 51
Handling Errors 51
Configuring Billing Management 52
Configuring Oracle HTTP Server for Billing Management 55

3 Complaint-to-Solution Business Process

Overview of the Complaint-to-Solution Business Process 1
About View and Capture Adjustments 2
About Collections Management 5
About the Collection Management Flow 5

Assumptions and Constraints for Working with Collection Management 6
Implementing the Complaint-to-Solution Business Process 7
About View and Capture Adjustments 7
QueryAccountBalanceAdjustment Integration Flow 7
CreateAccountBalanceAdjustment Integration Flow 9
BRM Interfaces 12
Siebel CRM Interfaces 12
Industry Oracle AIA Components 12
Integration Services 13
About Collections Management 15
Synchronizing Collection Scenarios and Actions Flow - BRM to Siebel CRM 16
Synchronizing Credit Alerts Flow - Siebel CRM to BRM 22
BRM Interfaces 25
Siebel CRM Interfaces 25
Industry Oracle AIA Components 25
Integration Services 25
Integrating Collections in a Multischema Environment 26
Configuring Complaint-to-Solution Business Process 27
About Configuring Collections Management 27
Setting Up BRM 27
Setting Up Siebel CRM 29
Setting Up Split Cross-Reference Tables 30
Working with DVMs 30
Working with Cross References 31
Handling Errors 31
Configuring the Collections Management Flow 32

4 Usage-to-Payment Business Process

Overview of the Usage-to-Payment Business Process 1
About Manage Customer Billing Process 1
Implementing Billing 2
About Generate Bill Invoice 3
About Invoice Template Customization 4
About Trial Billing 5
About Correct Bill Invoice 6
About Upload Invoice to DMS System 7
About Configure Billing 8
About Payments and Receivables Process 9
Placing an Adjustment 10
About General Ledger Requirements 11
About One-Time Payments 12

About Charging and Call Control 13
Performing Online Rating 13
Applying Usage Discounts 14
Applying Events to Bill Items 15
About Manage Collections Process 16
Qualifying Invoices for Collections 16
Implementing Collections B2B 17
Configuring Collections B2B 19
Implementing Collections B2C 20
Configuring Collections B2C 21
About Analysis of Siebel Collections Processing 23
About Collections Siebel Sends Message 24
About Collections Siebel Creates SR or Activity 25
About Collections Automated Disconnect, Barring, and Unbarring 26
About Exit Collections 27

A Mapping Siebel Billing Management UI Elements to BRM Customer Center

Billing Profile and Account Balance A-1
Bills A-2
Balance Group A-6
Unbilled Usage A-8
Payments A-9

B Reintroducing Enterprise Business Services

C Using the Oracle Mediator Resequencer Feature

Queues and Flows Enabled for Sequencing C-1
Resolving Errors in Flows with Resequencer C-3

About This Content

This guide describes how to implement the Cash to Care business process using Oracle Communications Digital Business Experience.

Audience

This document is intended for:

- Customer Service Representatives
- BRM Pricing Design Center administrators
- Individuals who are responsible for configuring, managing, and maintaining tasks and process flows for the Cash to Care business process.

You should be familiar with the TMF ODA Cash to Care business process and its flows.

1

Cash to Care Implementation Overview

This chapter describes the Cash to Care business processes in which you perform run time tasks related to billing, customer accounts, and other agent assisted care activities.

About Cash to Care

The Cash to Care journey starts after a sales order is received, processed, and fulfilled in the preceding Order to Cash journey. It covers all business processes that begin after a subscriber starts using a service.

The Cash to Care business processes are aligned with the TM Forum (TMF) Open Digital Architecture (ODA) business architecture.

The solution supports the following processes, which are realized by the functional capabilities and integration provided within and across Oracle Communications Billing and Revenue Management (BRM) and Siebel CRM:

- Request to Answer
- Complaint to Solution

When a subscriber calls a Customer Service Representative (CSR) to query about their mobile postpaid billing cycle, as part of the Request-to-Answer business process, the CSR navigates to Siebel CRM to fetch the details, retrieves the event details (CDR) from BRM, views the details in Siebel CRM, and answers the subscriber’s queries.

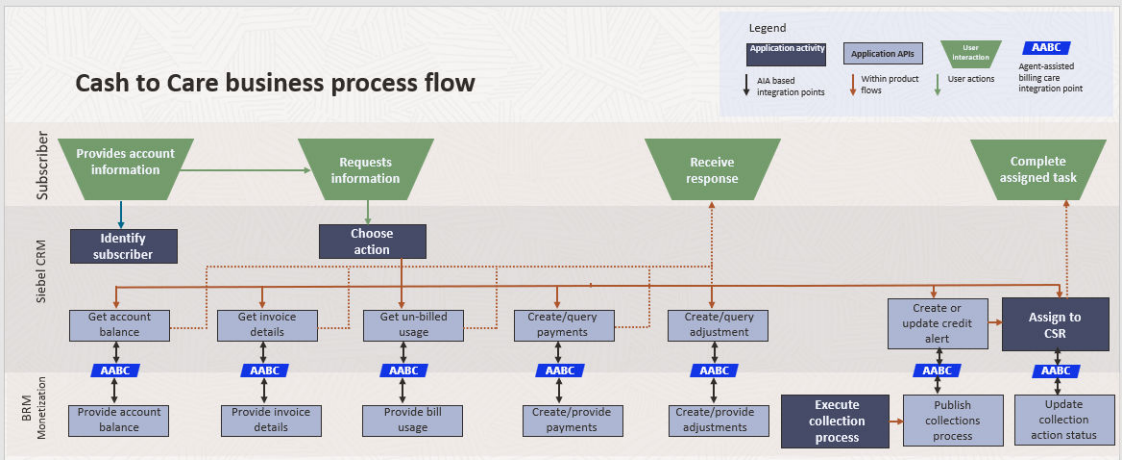
When a subscriber calls a CSR to query about the dispute noticed in their mobile postpaid invoice, as part of the Complaint-to-Solution business process, the CSR navigates to Siebel CRM to fetch the details, opens and checks the disputed invoice in Siebel CRM, creates an adjustment request in BRM, and sends the adjustment request details to the subscriber, which solves the subscriber's query.

The solution also uses Oracle Communications Application Integration Architecture (AIA), which provides end-to-end business process capabilities, to integrate business flows across Siebel CRM and BRM.

Figure 1-1 illustrates an overview of the Cash to Care business processes.

> **Note**
>
> AABC is a product within the Oracle Communications AIA application, which is referred as the AABC integration point in the following diagram.

Figure 1-1 Overview of Cash to Care Business Processes



The chapters that follow provide conceptual and procedural information about tasks related to billing, subscriber accounts, and other agent-assisted care activities.

About the Request-to-Answer Business Process

The Request-to-Answer business process comprises of activities relevant to managing customer requests across all communication channels (customer interfaces).

As a Customer Service Representative (CSR), you access subscriber-related billing information by using the Siebel CRM interface, including data that is created and maintained within the BRM system. You also synchronize subscriber information across Siebel CRM and BRM.

You answer queries pertaining to the following, using this business process between Siebel CRM and BRM:

- Subscriber management
- Billing management

See Overview of the Request-to-Answer Business Process for more information.

About the Complaint-to-Solution Business Process

The Complaint-to-Solution business process comprises of activities related to receiving a complaint (problem) initiated by the customer, analyzing it to identify the source of the issue, initiate resolution, monitor progress and close the trouble ticket.

As a Customer Service Representative (CSR), you create adjustments and manage collections. You also synchronize collection actions (based on the specified collection scenarios) defined in BRM to Siebel CRM, and administer these collection actions and credit alerts. The collection actions defined in BRM are synced to Siebel CRM using the Oracle Data Integrator (ODI) application.

You manage the collections process for subscribers who missed the payment due date. You also terminate the accounts of defaulters, which prevents subscribers from using services.

You handle subscriber queries regarding the following processes using this business process between Siebel CRM and BRM:

- Create adjustments
- Collection management

See Overview of the Complaint-to-Solution Business Process for more information.

2

Request-to-Answer Business Process

This chapter describes the Request-to-Answer business process and its features.

Overview of the Request-to-Answer Business Process

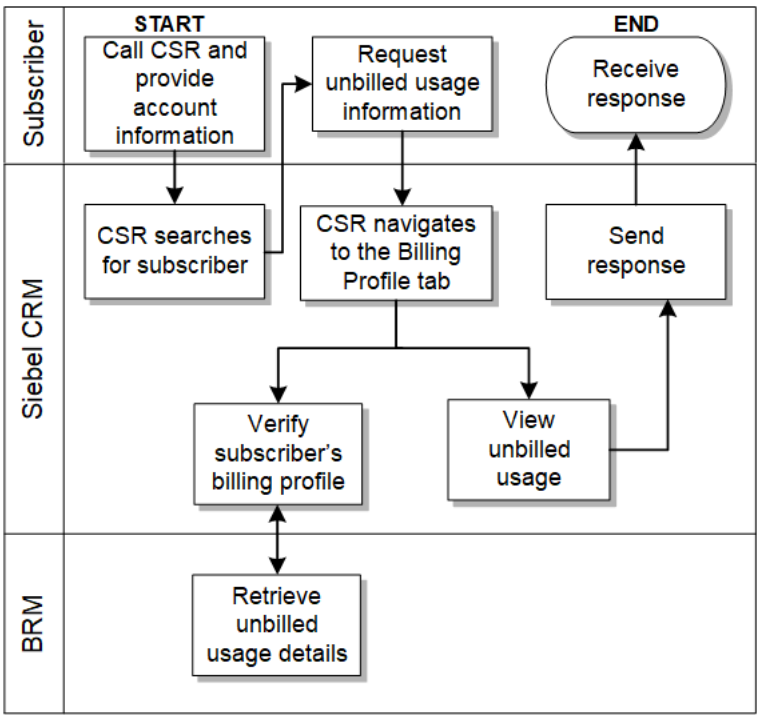
The Request-to-Answer business process comprises activities related to managing subscriber requests across all communication channels (customer interfaces). Any specific information requests or product requests from the subscriber are qualified and addressed by the Customer Service Representative (CSR).

It includes support for the following features:

- Billing management
- Query and view billing profile, accounts and customer balance summaries, balance groups, and balance details
- View unbilled usage information
- View CDR and detailed bills
- View and capture payments

Figure 2-1 illustrates a typical request to answer business process flow.

Figure 2-1 Typical Request-to-Answer Business Process Flow



The sections that follow describe the above mentioned features.

About Billing Management

The billing management flow enables a Customer Service Representative (CSR) to retrieve billing profile, account balances, balance groups, and balance summaries at a header, summary, and detail level.

After a service is activated, usage events (for example, a phone call, text message, or a data session) are sent from the communications network to the billing system. These events are rated, bills are generated, and then bills are sent to subscribers.

The service cycle of the billing management flow starts when subscribers call to query on the content of their bills. Billing information must be sent from BRM to Siebel CRM so that CSRs can respond to billing queries.

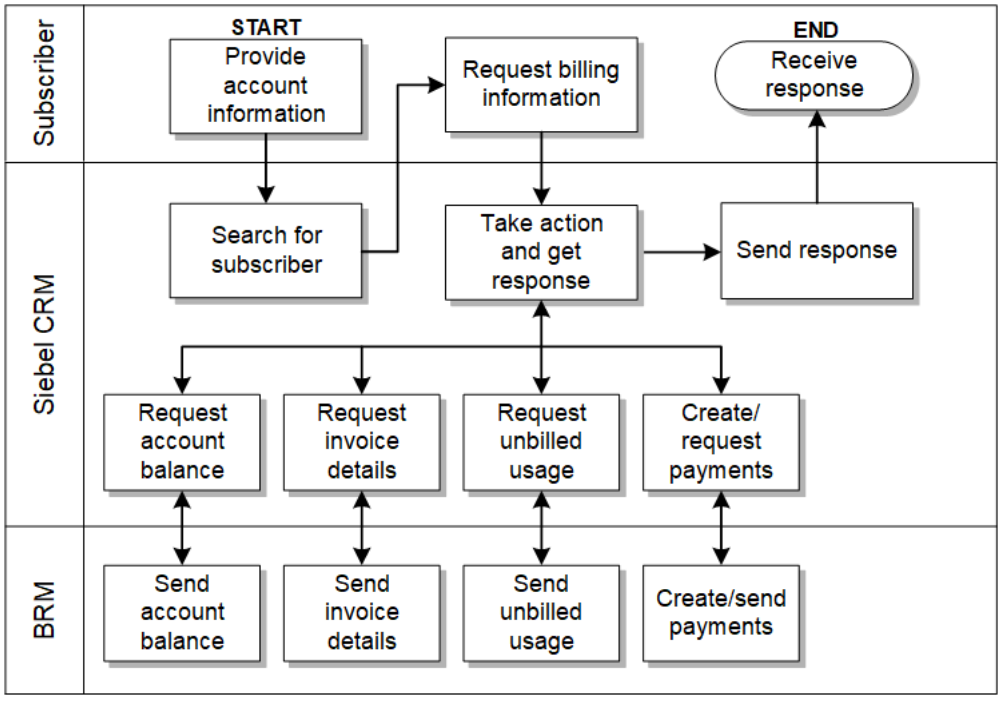
Most of the billing integration touch points do not replicate the billing data in Siebel CRM. Instead, the integration retrieves the billing data on demand from BRM and displays it within custom built billing UIs in Siebel CRM.

For this integration, Oracle supports Siebel CRM and BRM as the participating applications. You may use routing rules to retrieve billing data from multiple instances of BRM.

Billing Management Flow

Figure 2-2 illustrates the overall billing management flow.

Figure 2-2 Billing Management Flow



Solution Assumptions and Constraints

These are the assumptions and constraints for the billing management flow.

- One billing profile in Siebel CRM is associated with a single billing system.
- Billing data is not replicated in Siebel CRM. Rather, it is retrieved on demand from the billing system.
- Monetary and nonmonetary adjustments within a single adjustment request from Siebel CRM are not possible.
- A single adjustment request can have adjustments from only one level and not across multiple levels such as header, item, and event.
- Adjustment requests always originate in Siebel CRM and are sent to BRM for processing through collections management.
- Adjustment requests that are absolute and percentage value are supported at bill/header and event level only. Adjustment request of only absolute value are supported at the item level.
- Invoice adjustment requests are applicable at header, item, and event levels.
- Account-level adjustments are not supported.
- Unbilled adjustments are applicable only at the event level.
- Payment validation occurs in the billing system.
- Create Payments is used for making one-time payments and supports credit card and automatic debit payment methods.
- Viewing an invoice image in Siebel CRM is not supported for this release.

About Query and View Account Balances

The query and view account balances integration between Siebel CRM and BRM:

- Enables a Customer Service Representative (CSR) to view an account billing profile balance summary in Siebel CRM.
- Enables a CSR to view the list of balance groups under an account billing profile.
- Enables a CSR to view balance details under a balance group.
- Enables a CSR to view services under a balance group.

The query and view account balances integration enables the CSR to retrieve balance information from BRM. The CSR can obtain balance information at a summary or detail level. They can also query a list of balance groups for a selected billing profile.

After the CSR obtains the balance group information from BRM, they can navigate and query additional details, such as monetary and nonmonetary balances, credit limits, and validity dates.

When the CSR accesses the Siebel Billing Profile screen, a request is made from Siebel CRM to retrieve an account balance summary. When the CSR clicks the **Balance Group** tab, a call is made to fetch the list of balance groups. When the CSR clicks the **View Detail** button for a balance group, the corresponding balance group detail data appears on the **Balance Group Detail** applet.

With the add-on support for viewing account balance and other billing data for service accounts, the account balance integration now enables the CSR to retrieve balance and billing information for the service account (in addition to the billing account) from BRM.

From the Account Summary view in Siebel CRM, the CSR can retrieve both the account balance information of the billing (paying) account and the service (nonpaying) account. The

click stream action determines which account billing information is retrieved from the billing system. By default there are the following three navigation paths available:

- From the **Billing Account Summary** page, navigate to the **Siebel Billing Portal** page for the billing account.

  If the intent is to see the billing account's billing data (account balance, bills, unbilled usage, and payments), then the user must use the **Billing Profile** link under the **Billing Profile** applet.

  In cases where a single billing profile is used to pay for both the billing account and service account(s) services, then the billing data that is displayed using this navigation option contains the consolidated billing information of account balance, bills, unbilled usage, and payments across all accounts that are tied to this billing profile.

  Alternatively, if different parent billing profiles are used to pay for self and other service accounts, then clicking a specific billing profile displays the billing data of account(s) that are tied to that billing profile.

- From the **Billing Account Summary** page, navigate to the **Siebel Billing Portal** page for one service account.

  If the intent is to see a particular service account's billing data, then the user must use the **Billing Profile** link under the **Billing Items** applet.

  The subscriber must identify the correct billing item that is mapped to the service account for which the billing data is to be seen. Billing data like account balance, bills, and unbilled usage that are specific to the selected service account can be viewed using this navigation option.

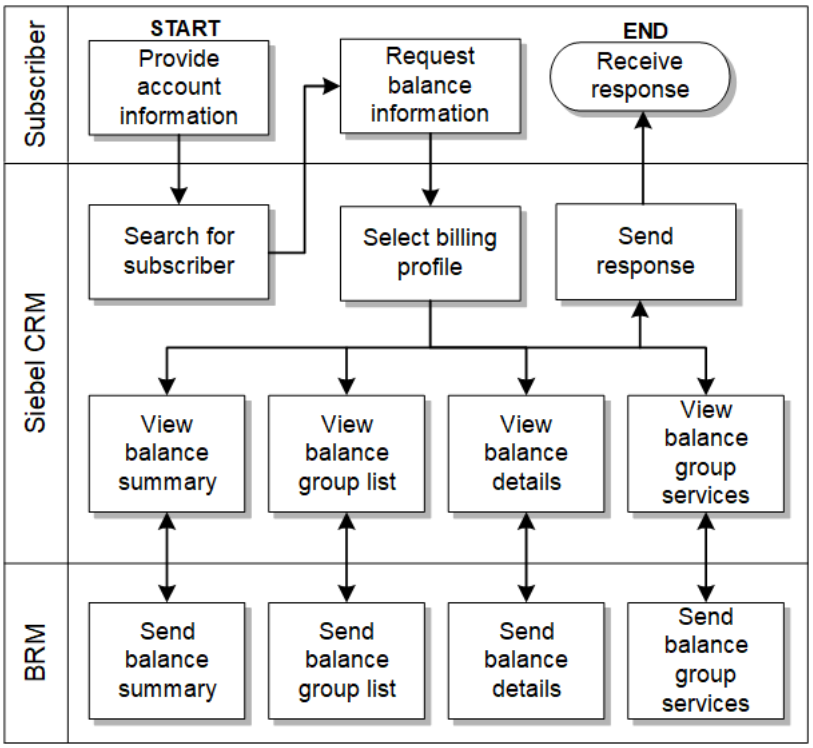
- From the **Service Account Summary** page, navigate to the **Siebel Billing Portal** page for the service account.

  If the intent is to see the billing data of the service account that is currently accessed in Siebel CRM, the subscriber must use the **Billing Profile** link under the **Installed Asset** applet.

  Clicking on any of the **Billing Profile** links under this applet navigates the subscriber to the service account's billing portal screen. Billing data like account balance, bills, unbilled usage, and adjustments that are specific to a service account can be viewed using this navigation option.

The Query and View Account Balances Flow illustrates the account balance flow.

Figure 2-3 The Query and View Account Balances Flow



To view account balance information for an account using the Billing Profile portal screen in Siebel CRM:

1. Navigate to the Accounts screen and query an account.
2. Scroll down to the **Billing Profile** applet for this account and click the **Billing Profile Name** link.

   You can now view the account's billing profile portal screen which has the applet with the balance summary for this account.
3. Click the **Balance Group** tab to view the list of balance groups under an account billing profile. Click **View Detail** to view the corresponding balance group detail on the Balance Detail applet.

See Mapping Siebel Billing Management UI Elements to BRM Customer Center for more information about the mapping of Siebel CRM elements to BRM elements.

See About Query and View Account Balances for information about implementing the account balances feature.

About View CDR and Detailed Bills (Invoice)

This integration between Siebel CRM and BRM:

- Enables you to view a list of invoices for an account billing profile in Siebel CRM.
- Enables you to view invoice details in Siebel CRM.
- Enables you to view invoice event details, which is also known as call detail records (CDR) in Siebel CRM.

- Enables you to search invoice event detail records (CDR) in Siebel CRM.
- Enables you to view nonmonetary resource balance details in Siebel CRM.

BRM generates invoices on a periodic basis after a bill cycle. Once generated, an invoice does not change. An invoice usually consists of sections for a header, a summary, and details. You can view invoice data at any of these levels, and resolve most bill inquiries by viewing the header and summary.

When you select the **Bills** tab of the Billing Profile screen, the integration shows a list of the latest invoices. You set the number of invoices to display in Siebel CRM, and the integration retrieves that many invoices from BRM. From this list, you can take the following actions:

- View invoice details:
  1. Click the **Bill Number** link.

     The Bill Detail view appears and the integration retrieves the invoice or bill header and item data, and displays it on the header and items applet of the Bill Detail view. A single request from Siebel CRM invokes two APIs in BRM that call the invoice header and items data and return a single hierarchical message to Siebel CRM.
  2. Click the **Payment** tab or the **A/R items** tab.

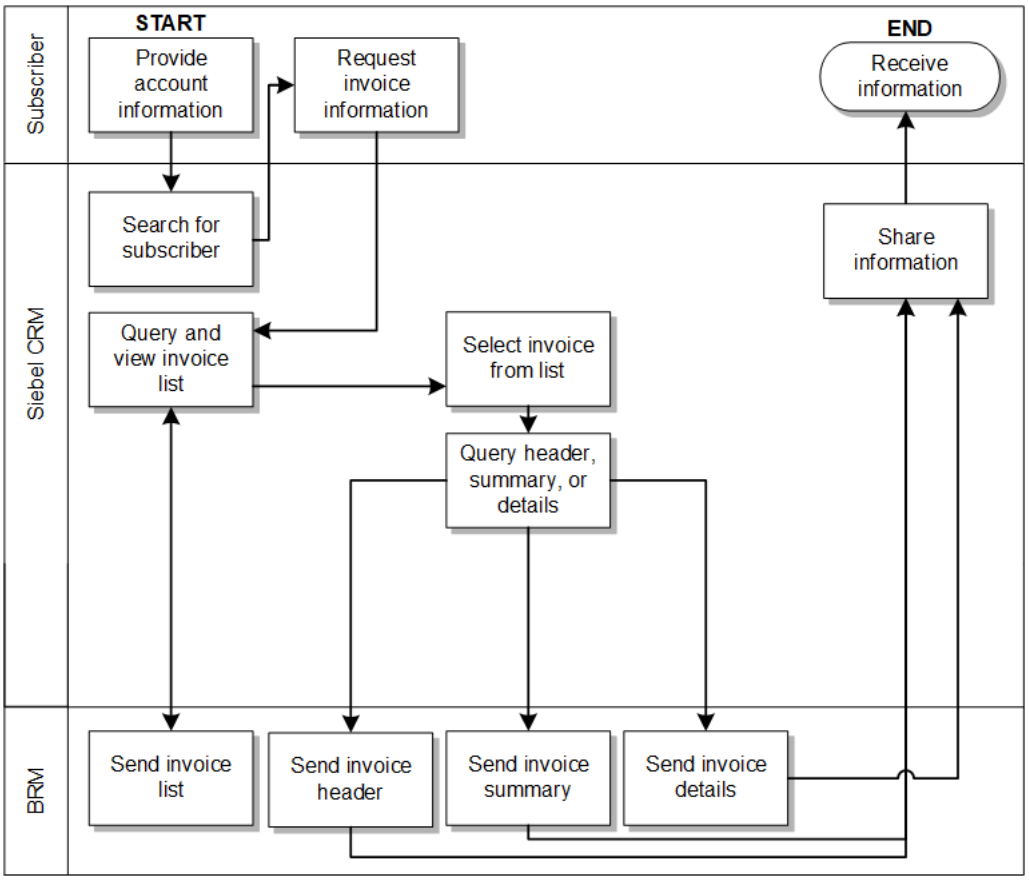
     The integration retrieves payment or A/R items data for the selected invoice or bill.
- View item or event details (CDR):
  1. Click the **Bill Number** link.
  2. Select an invoice item.
  3. Click the **Net Amount** link associated with the item charge.

     The integration retrieves the item or event details record from BRM. Because this item detail data is generally large, you can filter the data using search criteria.
- View individual service charges:
  1. Click the **Bill Number** link.
  2. Select the Service Charges applet.

> **Note**
>
> The date and time information of the event details and CDR displayed in Siebel CRM is specific to the time zone from where the event originated.

Figure 2-4 illustrates the view CDR and detailed bills flow.

Figure 2-4 The CDR and Detailed Bills Flow



To view invoice information for an account in the billing profile in Siebel CRM:

1. Navigate to the Accounts screen and query an account. Scroll down to the **Billing Profile** applet for this account and click the **Billing Profile Name** link.
2. Click the **Bills** tab to view a list of invoices.
3. Click the **Bill Number** link to view the details for that invoice.
4. Go to the Item Charges applet and click the **Net Amount** link to view the invoice usage allocation (CDRs) for that item.
5. Click the **Search** tab to search invoice event detail records.
6. In the Event Details, click **View Details** to query the resource impact details (non-monetary charges, for example free minutes, and so on) for an event from the billing system.

See Mapping Siebel Billing Management UI Elements to BRM Customer Center for more information on mapping Siebel CRM elements to BRM elements.

See About View CDR and Detailed Bills (Invoice) for information about implementing the view CDR and detailed bills feature.

About View Unbilled Usage

The Unbilled Usage integration between Siebel CRM and BRM:

- Enables a Customer Service Representative (CSR) to view account-billing-profile-service-usage-summary in Siebel CRM.
- Enables a CSR to view account-billing-profile-service-usage-charge-details in Siebel CRM.
- Enables a CSR to search account-billing-profile-service-usage-charge-details based on a few columns.
- Enables a CSR to view resource balances (for example, nonmonetary) for service usage events in Siebel CRM.

Events that have been captured by BRM but have not been billed are called unbilled usage events. Subscriber questions about unbilled (service) usage are the most common queries for CSRs. This is because Wireless communications service providers (CSPs) offer service plans that include free usage per bill cycle.

Examples include:

- 1000 Free Local and Long Distance Minutes per month.
- 10 Free SMS per month.
- 1MB of data download.

Unlike invoice information, unbilled usage information is constantly changing. The BRM has the most current status of all unbilled usage, so Siebel CRM must retrieve this information, in real time, so that the CSR can respond to subscribers' questions accurately.

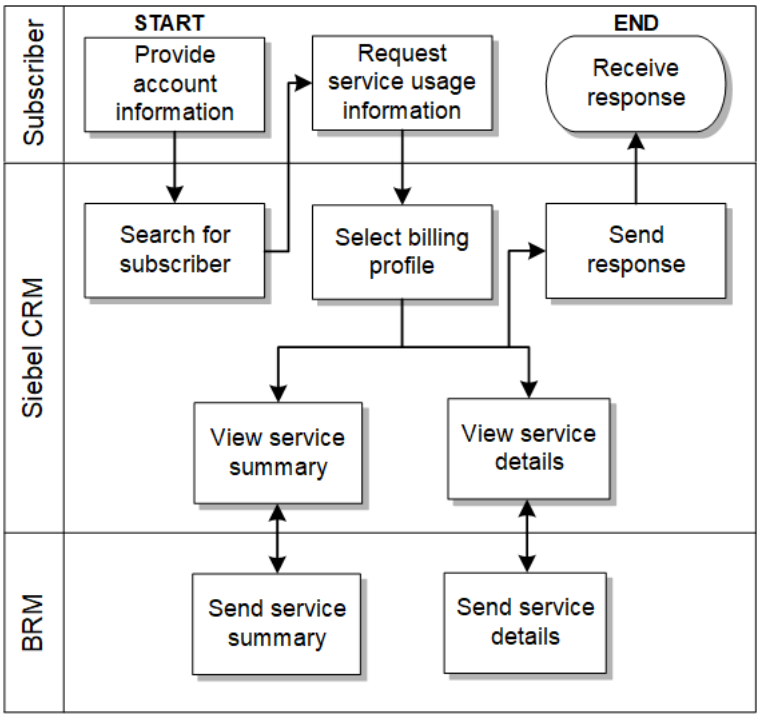
When the CSR clicks the **Unbilled Usage** tab on the Billing Profile screen, it triggers a request to retrieve item charges from the billing system. The integration process aggregates the item charge information to the services level so that service charges can be displayed in the **Service Charges** applet.

CSRs can request detailed unbilled data at the event level and the call detail records (CDR). This is a separate request to BRM. Because this data is generally large, CSRs can limit the data returned by specifying search criteria.

> **Note**
>
> The date and time information of the event details and CDR, which is displayed in the Siebel UI, is specific to the time zone from where the event originated.

The View Unbilled Usage Flow illustrates the View Unbilled Usage flow.

Figure 2-5 The View Unbilled Usage Flow



To view account billing profile service usage in Siebel CRM:

1. Navigate to the Accounts screen and query an account. Scroll down to the **Billing Profile** applet for this account and click the **Billing Profile Name** link.
2. Click the **Unbilled Usage** tab to view the account's billing profile service usage summary.
3. Click the **Net Amount** link for a particular item charge to view account billing profile service usage allocation (CDRs).
4. Click the **Search** tab to search account billing profile service usage charge details.
5. Click the **View Details** tab to view resource balances for service usage events.

See Mapping Siebel Billing Management UI Elements to BRM Customer Center for more information on mapping Siebel CRM elements to BRM elements.

See About View Unbilled Usage for information about implementing the unbilled usage feature.

About View and Capture Payments

The view and capture payments integration between Siebel CRM and BRM:

- Enables you to capture a payment in Siebel CRM either for an account at the billing profile level or at the invoice level and to post the payment in BRM.
- Enables you to view the history of payments in Siebel CRM at both the billing profile level and invoice level by retrieving payment records from BRM.
- Enables you to search for payment records in BRM to display in Siebel CRM for an account at the billing profile level or at the invoice level.

See About View and Capture Payments for information about implementing the view and capture payments feature.

Capturing Payments

Subscribers can make payments in many ways: using a credit card or debit card over the phone, using a credit or debit card at a self-service location, using a check by mail, or using an electronic payment from a bank account.

The integration accepts payments from these channels in two ways:

- From Siebel CRM: A Customer Service Representative (CSR) captures payment information in Siebel CRM.
- From the cross-channel system directly: Payment information is captured in the cross-channel system, such as subscriber self-service, retail locations, web commerce, and partner systems.

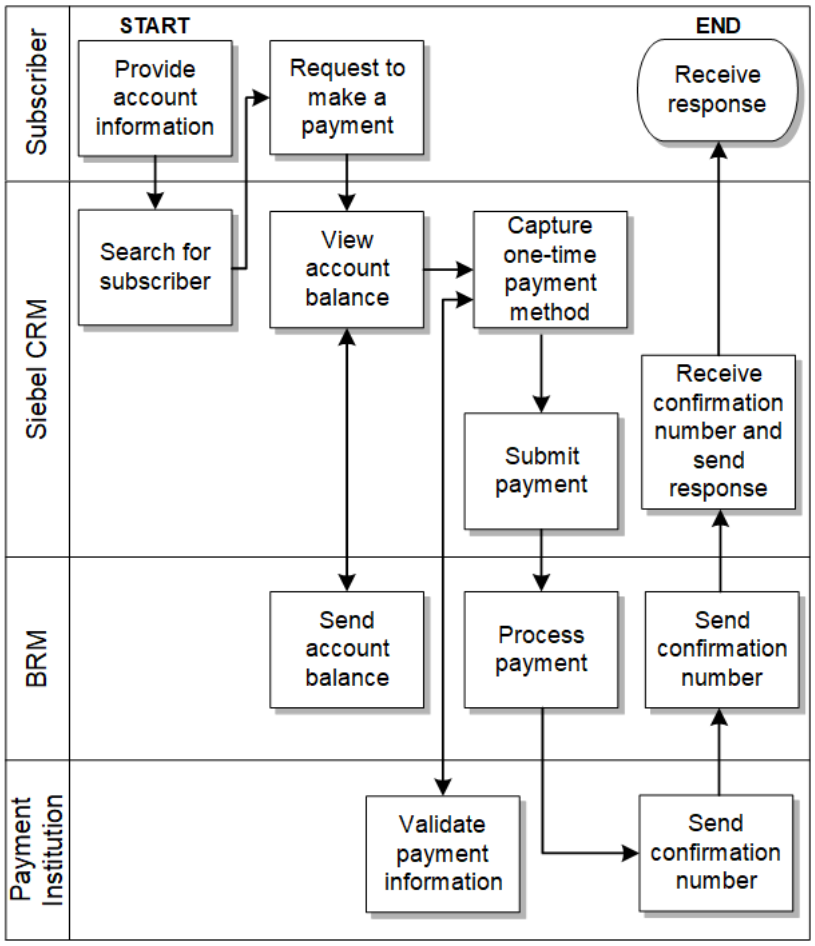
Capturing Payments in Siebel CRM

You capture one-time payments in Siebel CRM and send them to BRM for processing. BRM processes the payments and returns confirmation to Siebel CRM, which your subscribers can keep for their records.

You can accept one-time payments from self-paying accounts and non-paying subordinate accounts. All payments for the non-paying account are paid by a parent account. You capture one-time payments for non-paying accounts on the parent account's billing profile screen.

Figure 2-6 illustrates the flow for capturing payments in Siebel CRM.

Figure 2-6 The Capturing Payments Flow



To view the history of payments and to capture new payments in Siebel CRM:

1. Navigate to the Accounts screen and query an account. Scroll down to the **Billing Profile** applet for this account and click the **Billing Profile Name** link.
2. Click the **Payments** tab to view the history of payments.
3. Click **New** to capture a new payment for this account.
4. From the **Payments** tab, click **Search** to open the **Search** applet and search for specific payment records.

See Mapping Siebel Billing Management UI Elements to BRM Customer Center for more information on mapping Siebel CRM elements to BRM elements.

Viewing Payment History

Regardless of how a payment is made, it must be displayed in the payment history retrieved from the billing system into Siebel CRM. BRM maintains payment information for a subscriber for a fixed period which varies by your legal requirements and business needs.

During a sales or service process, subscribers may want to know their payment history. You can query a subscriber's past payments to respond to queries from the **Payments** tab of the Siebel CRM Billing Profile screen, triggering a request to the billing system to return the latest *n* payments. You can also specify search criteria to find the correct payment record if the initial

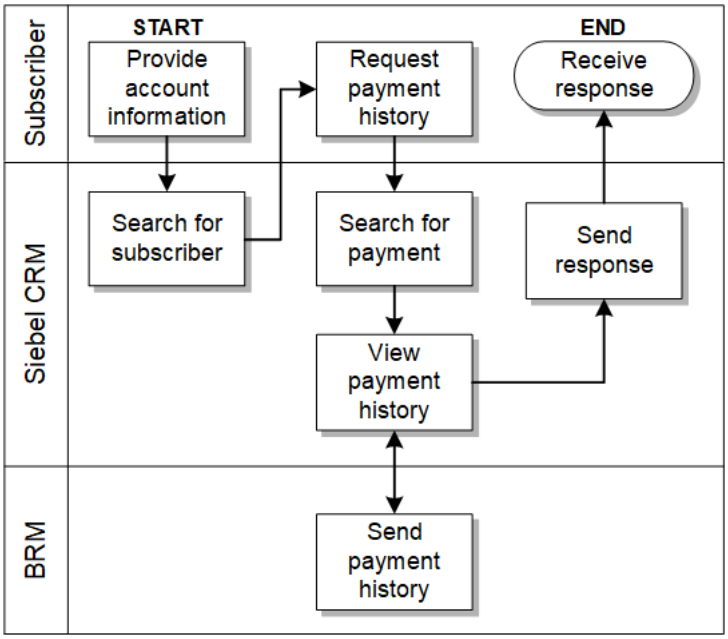
list of payments is not the right one. The **Payment History** view also indicates the subscriber of any payment reversals that might have been made on a payment that has been previously processed and allocated.

Searching for Payments

The Search Payment feature lets you search for specific payment records in BRM based on the subscriber request and displays these in Siebel CRM. The search is performed for an account at the billing profile level or at the invoice level. You search for payments on either a date range or a paid amount.

Figure 2-7 illustrates the flow for Viewing Payment History.

Figure 2-7 The View Payment History Flow



About Account Status Synchronization

You can synchronize account status changes from Siebel CRM to BRM. Account status synchronization enhances collections management, which is delivered by the Cash to Care business process. Oracle recommends that you enable account status synchronization only if you are also using the collections management process flow. Currently, the account status synchronization flow supports only **Active** and **Deactive** statuses.

The Complaint to Solution business process synchronizes collections actions generated by BRM as credit alerts in Siebel CRM, where a Customer Service Representative (CSR) can take actions on the subscriber's account, such as suspending or canceling services.

CSR can suspend or cancel services with change orders that are either manually submitted by a CSR or automatically generated based on credit alerts. Extend Siebel CRM to automatically generate change orders based on credit alerts. Using change orders ensures that service state changes are synchronized from Siebel CRM to BRM.

If you must deactivate a subscriber account due to continued delinquency, enabling account status synchronization ensures that account status change in Siebel CRM is synchronized to BRM.

Synchronizing account status to BRM is disabled by default. You can enable it by changing the value of the **EnableAccountStatusSync** property in the **AIAConfigurationProperties.xml** file. See Configuring Customer Management in *Oracle Communications Digital Business Experience Order to Cash Implementation Guide.* for detailed instructions.

When deactivating accounts in Siebel CRM, Oracle recommends the following:

- Deactivate accounts in Siebel CRM only after canceling all the services and account-level subscription products for that account in Siebel CRM. When you deactivate an account in Siebel CRM, the status change is immediately synchronized to BRM. BRM cascades status changes from the account to all of its **/billinfo** objects, so the services and products in BRM are canceled as well. If you deactivate the account before canceling the services and products in Siebel CRM, they continue to appear active in Siebel CRM even after BRM cancels them.
- To avoid inadvertent deactivation of accounts with active services, Oracle recommends restricting the ability to deactivate accounts to particular Siebel CRM users and roles. Siebel CRM does not let you restrict account status changes in other ways.

Implementing the Request-to-Answer Business Process

This section explains how the Oracle Application Integration Architecture (Oracle AIA) Oracle Communications Cash to Care business process implements the Request-to-Answer business process.

About Query and View Account Balances

The query and view account balances integration between Siebel CRM and BRM supports the following integration flows:

- QueryBalanceSummary enables a Customer Service Representative (CSR) to view an account billing profile balance summary in Siebel CRM.
- QueryBalanceGroupList enables a CSR to view the list of balance groups under an account billing profile.
- QueryBalanceDetails enables a CSR to view balance details under a balance group.
- QueryBalanceGroupServices enables a CSR to view services under a balance group.

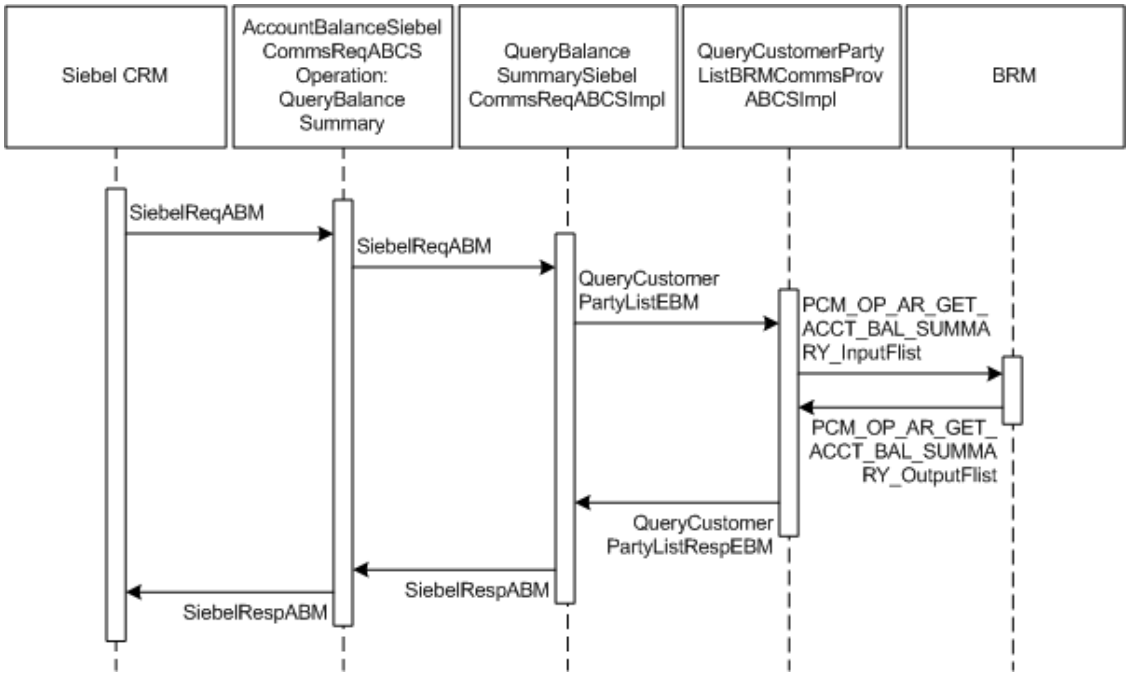
QueryBalanceSummary Integration Flow

This integration flow uses the following interfaces:

- AccountBalanceSiebelCommsReqABCS with operation QueryBalanceSummary
- QueryBalanceSummarySiebelCommsReqABCSImpl
- QueryCustomerPartyListBRMCommsProvABCSImpl

Figure 2-8 illustrates the QueryBalanceSummary integration scenario.

Figure 2-8 QueryBalanceSummary Integration Flow Sequence Diagram



When you initiate the QueryBalanceSummary process, the following events occur:

1. In Siebel CRM, a subscriber navigates to the Accounts screen, queries an account, and clicks a **Billing Profile** for the account.

   This opens up the Billing Profile BRM screen, and a web service call is made to query the balance summary from the billing system.

2. Navigating to the Billing BRM screen invokes AccountBalanceSiebelCommsReqABCS web service with the QueryBalanceSummary method, which in turn calls the AccountBalanceSiebelCommsReqABCS with operation QueryBalanceSummary.

   AccountBalanceSiebelCommsReqABCS is a generic Siebel Account Balance interface service with several operations defined on the application business message (ABM).

3. Invoking AccountBalanceSiebelCommsReqABCS with operation QueryBalanceSummary routes the QueryBalanceSummaryReqMsg to the QueryBalanceSummarySiebelCommsReqABCSImpl.

4. The QueryBalanceSummarySiebelCommsReqABCSImpl first transforms the QueryBalanceSummaryReqMsg into QueryCustomerPartyListReqMsgEBM and routes the QueryCustomerPartyListReqMsg to the appropriate billing systems.

   As delivered, QueryCustomerPartyListReqMsg is routed to the QueryCustomerPartyListBRMCommsProvABCSImpl.

5. QueryCustomerPartyListBRMCommsProvABCSImpl transforms QueryCustomerPartyListReqMsg into the input of PCM_OP_AR_GET_ACCT_BAL_SUMMARY and calls the opcode PCM_OP_AR_GET_ACCT_BAL_SUMMARY.

6. QueryCustomerPartyListBRMCommsProvABCSImpl then transforms the application programming interface (API) output PCM_OP_AR_GET_ACCT_BAL_SUMMARY _outputFlist into enterprise business message (EBM) QueryCustomerPartyListRespMsg and returns it to QueryBalanceSummarySiebelCommsReqABCSImpl.

7. QueryBalanceSummarySiebelCommsReqABCSImpl transforms the QueryCustomerPartyListRespMsg into QueryBalanceSummaryRespMsg, which is returned to the AccountBalanceSiebelCommsReqABCS.
8. AccountBalanceSiebelCommsReqABCS returns the QueryBalanceSummarySiebelMsg to the calling Siebel web service AccountBalanceSiebelCommsReqABCS.
9. The response is then written to the Siebel Balance Summary virtual business component (VBC) for the subscriber.

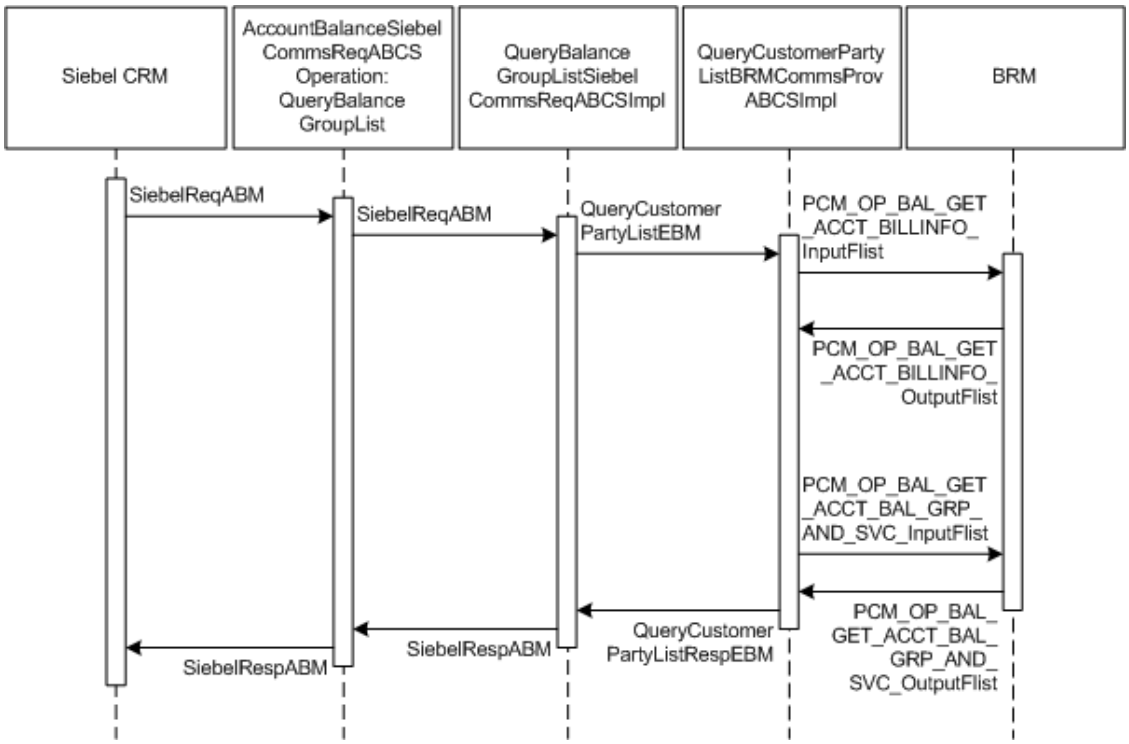
QueryBalanceGroupList Integration Flow

This integration uses the following service interfaces:

- AccountBalanceSiebelCommsReqABCS with operation QueryBalanceGroupList
- QueryBalanceGroupListSiebelCommslReqABCSImpl
- QueryCustomerPartyListBRMCommsProvABCSImpl

Figure 2-9 illustrates the QueryBalanceGroupList integration scenario.

Figure 2-9 QueryBalanceGroupList Integration Flow Sequence Diagram



When you initiate the QueryBalanceDetails process, the following events occur:

1. In Siebel CRM, a subscriber navigates to the Accounts screen, queries an account, and clicks a **Billing Profile** for the account.

   This displays the Billing Profile BRM screen. On the Billing Profile screen, when the subscriber clicks the **Balance Group** tab, a web service call is made to query the complete list of balance groups for that account billing profile.
2. Navigating to the Billing BRM screen and clicking the **Balance Group** tab invokes the AccountBalanceSiebelCommsReqABCS web service, which in turn invokes the Siebel

Account Balance Interface service AccountBalanceSiebelCommsReqABCS with operation QueryBalanceGroupList.

AccountBalanceSiebelCommsReqABCS is a generic Oracle AIA Application Business Connector Service (ABCS) interface service with several operations on the Siebel ABM.

3. Invoking AccountBalanceSiebelCommsReqABCS with operation QueryBalanceGroupList routes the QueryBalanceGroupListReqMsg to the QueryBalanceGroupListSiebelCommsReqABCSImpl.
4. The QueryBalanceGroupListSiebelCommslReqABCSImpl transforms the QueryBalanceGroupListReqMsg into QueryCustomerPartyListReqMsgEBM and routes the QueryCustomerPartyListReqMsg to the appropriate billing system.

   As delivered, QueryCustomerPartyListReqMsg is routed to the QueryCustomerPartyListBRMCommsProvABCSImpl.
5. QueryCustomerPartyListBRMCommsProvABCSImpl checks the Query Criteria code. If it is QueryBalanceGroupList, the QueryInvoiceListReqMsg is transformed into PCM_OP_BAL_GET_ACCT_BILLINFO_inputflist.

   This opcode call returns the list of BILLINFO and AR_BILLINFO of that account.
6. QueryCustomerPartyListBRMCommsProvABCSImpl first checks the Query Criteria code. If it is QueryBalanceGroupList, then it transforms QueryCustomerPartyListReqMsg into the input of PCM_OP_BAL_GET_ACCT_BAL_GRP_AND_SVC and then invokes the BRM API PCM_OP_BAL_GET_ACCT_BAL_GRP_AND_SVC to query the list of balance groups of the account billing profile.
7. From the response of PCM_OP_BAL_GET_ACCT_BILLINFO opcode, the appropriate BILLINFO and AR_BILLINFO are picked.

   QueryCustomerPartyListReqMsg is transformed into the input of PCM_OP_BAL_GET_ACCT_BAL_GRP_AND_SVC and calls the BRM opcode PCM_OP_BAL_GET_ACCT_BAL_GRP_AND_SVC. Several balance groups can be in the billing system for an account billing profile. Based on the value of *n* passed from Siebel CRM, the API returns <=n number of balance groups.
8. The list of balance groups from the BRM output list is transformed into QueryCustomerPartyListRespMsg and returned to the QueryBalanceGroupListSiebelCommslReqABCSImpl service.
9. QueryBalanceGroupListSiebelCommslReqABCSImpl then transforms the response messages into QueryBalanceGroupListRespMsg, which is returned to the AccountBalanceSiebelCommsReqABCS.
10. AccountBalanceSiebelCommsReqABCS returns the QueryBalanceGroupListRespMsg to the calling Siebel web service AccountBalanceSiebelCommsReqABCS.
11. The response message is then written to the Siebel Balance Group VBCs for the subscriber.

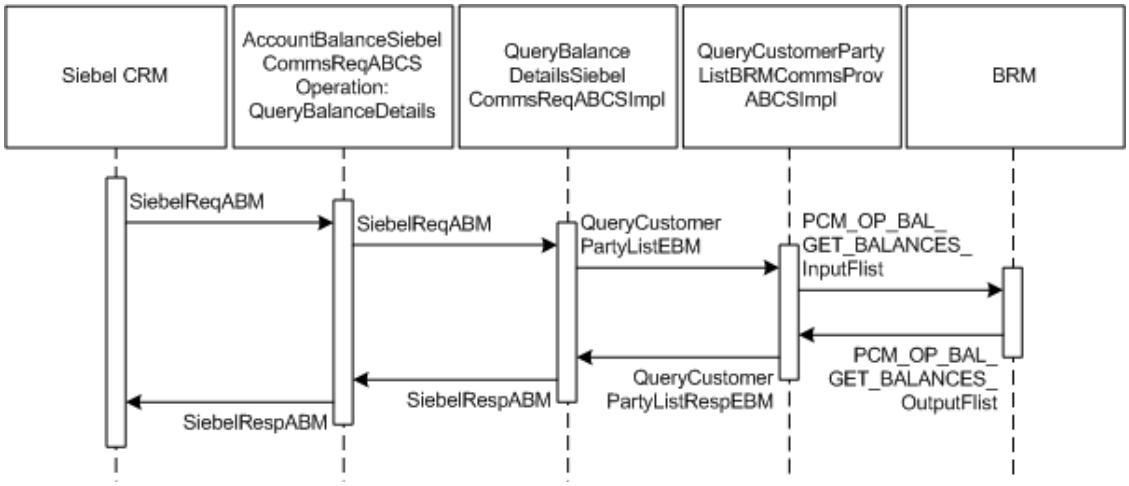
QueryBalanceDetails Integration Flow

This integration uses the following service interfaces:

- AccountBalanceSiebelCommsReqABCS with operation QueryBalanceDetails
- QueryBalanceDetailsSiebelCommsReqABCSImpl
- QueryCustomerPartyListBRMCommsProvABCSImpl

Figure 2-10 illustrates the QueryBalanceDetails integration scenario.

Figure 2-10 QueryBalanceDetails Integration Flow Sequence Diagram



When you initiate the QueryBalanceDetails process, the following events occur:

1. Siebel web service calls AccountBalanceSiebelCommsReqABCS (operation - QueryBalanceDetails) with QueryBalanceDetailsReqMsg ABM comprising account ID, billing profile ID, and balance group ID.
2. AccountBalanceSiebelCommsReqABCS invokes QueryBalanceDetailsSiebelCommsReqABCSImpl with QueryBalanceDetailsReqMsg ABM.
3. QueryBalanceDetailsSiebelCommsReqABCSImpl transforms QueryBalanceGroupListReqMsg ABM to QueryCustomerPartyListReqMsgEBM and sets the value of the field, Query Criteria Code, to Query Balance Details.
4. QueryBalanceDetailsSiebelCommsReqABCSImpl then routes the QueryCustomerPartyListReqMsgEBM to QueryCustomerPartyListBRMCommsProvABCSImpl.
5. QueryCustomerPartyListBRMCommsProvABCSImpl ensures that the value in Query Criteria Code is Query Balance Details and transforms QueryCustomerPartyListReqMsgEBM into BRM ABM and calls the BRM API, PCM_OP_BAL_GET_BALANCES.
6. PCM_OP_BAL_GET_BALANCES takes balance group ID as the input and returns the balance and balance details for that balance group.
7. QueryCustomerPartyListBRMCommsProvABCSImpl transforms the BRM output to QueryCustomerPartyListResMsgEBM.
8. QueryCustomerPartyListResMsgEBM goes as a response to QueryBalanceDetailsSiebelCommsReqABCSImpl.

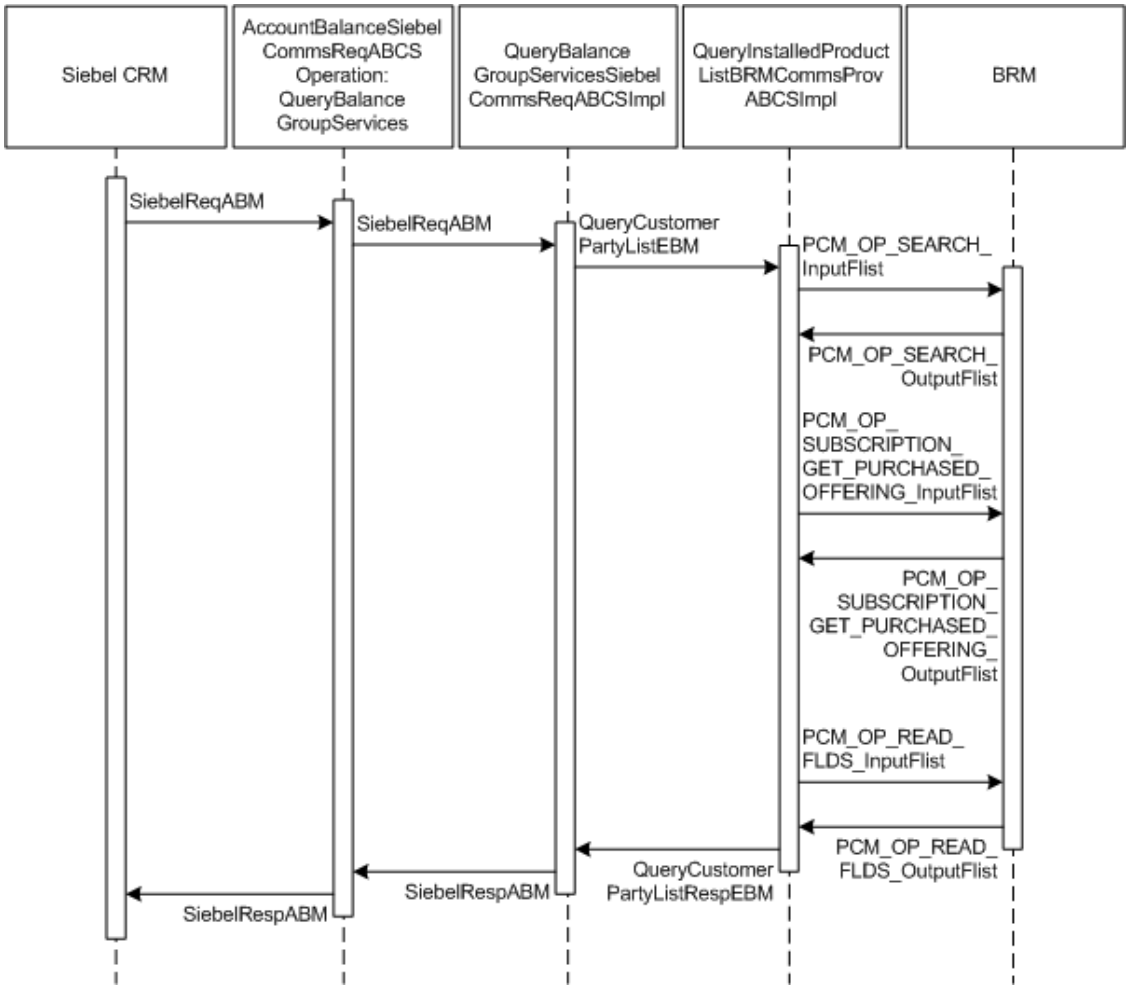
QueryBalanceGroupServices Integration Flow

This integration uses the following service interfaces:

- AccountBalanceSiebelCommsReqABCS
- QueryBalanceGroupServicesSiebelCommsReqABCSImpl
- QueryInstalledProductListBRMCommsProvABCSImpl

Figure 2-11 illustrates the QueryBalanceGroupServices integration scenario.

Figure 2-11 QueryBalanceGroupServices Integration Flow



When you initiate the QueryBalanceGroupServices process, the following events occur:

1. Siebel web service calls AccountBalanceSiebelCommsReqABCS using operation QueryBalanceGroupServices with QueryBalanceGroupServicesReqMsg ABM comprising account ID, billing profile ID, and balance group ID.
2. AccountBalanceSiebelCommsReqABCS invokes QueryBalanceGroupServicesSiebelCommsReqABCSImpl with QueryBalanceGroupServicesReqMsg ABM.
3. QueryBalanceGroupServicesSiebelCommsReqABCSImpl transforms QueryBalanceGroupServicesReqMsg ABM to QueryInstalledProductListReqMsgEBM and sets the value of the field, Query Criteria Code, to Query Balance Group Services.
4. QueryBalanceGroupServicesSiebelCommsReqABCSImpl routes QueryInstalledProductListReqMsgEBM to QueryInstalledProductListBRMCommsProvABCSImpl.

   This service:

   a. Transforms QueryInstalledProductListReqMsgEBM to BRM ABM and calls the BRM API, PCM_OP_SEARCH, which takes a query statement involving balance group ID as input and returns the list of service IDs for that balance group.

b. Calls the BRM API, PCM_OP_SUBSCRIPTION_GET_PURCHASED_OFFERINGS for each of the service IDs queried, which accepts a service ID as input and returns the list of product IDs associated with that service.

c. Calls the BRM API, PCM_OP_READ_FLDS for each of the product IDs queried, which takes a product ID as input and returns the product details for that ID.

5. QueryInstalledProductListBRMCommsProvABCSImpl merges and transforms the BRM output to QueryInstalledProductListResMsg enterprise business message (EBM).

6. QueryInstalledProductListResMsgEBM goes as a response to QueryBalanceGroupServicesSiebelCommsReqABCSImpl.

BRM Interfaces

The QueryBalanceSummary integration flow uses this opcode:

- PCM_OP_AR_GET_ACCT_BAL_SUMMARY

The QueryBalanceGroupList integration flow uses these opcodes:

- PCM_OP_AR_GET_ACCT_BILLS
- PCM_OP_BAL_GET_ACCT_BAL_GRP_AND_SVC

The QueryBalanceDetails integration flow uses this opcode:

- PCM_OP_BAL_GET_BALANCES

The QueryBalanceGroupServices integration flow uses these opcodes:

- PCM_OP_SEARCH
- PCM_OP_SUBSCRIPTION_GET_PURCHASED_OFFERINGS
- API, PCM_OP_READ_FLDS

See *Oracle Communications Billing and Revenue Management Opcode Flist Reference* for more information.

Siebel CRM Interfaces

The query and view account balances flow uses this Siebel CRM interface:

- AccountBalanceSiebelCommsReqABCS

See the web services reference information in *Siebel Order Management Guide Addendum for Communications* for more details about this web service.

Industry Oracle AIA Components

The query and view account balances flow uses the following delivered enterprise business objects (EBOs) and enterprise business messages (EBMs):

- CustomerPartyEBO
- InstalledProductEBO
- QueryCustomerPartyListEBM
- QueryCustomerPartyListResponseEBM
- QueryInstalledProductListEBM
- QueryInstalledProductListResponseEBM

The following directories contain the industry component files:

- Enterprise business object (EBO) and enterprise business message (EBM) XML schema files:

  COMMS_AIA_HOME/**comms_home/source/soainfra/apps/AIAMetaData/AIAComponents/EnterpriseObjectLibrary/Industry/Communications/EBO/**

- Enterprise business service (EBS) WSDL files:

  COMMS_AIA_HOME/**comms_home/source/soainfra/apps/AIAMetaData/AIAComponents/EnterpriseBusinessServiceLibrary/Industry/Communications/EBO/**

For detailed documentation of individual EBOs and EBMs, click the AIA Reference Doc link on EBO and EBM detail pages in Oracle Enterprise Repository (OER).

EBOs can be extended such as adding new data elements. These extensions are protected and will remain intact even after a patch or an upgrade, so long as the extensibility guidelines are followed.

See the discussion of Oracle AIA assets extensibility patterns in *Oracle Fusion Middleware Developer's Guide for Oracle SOA Core Extension* for more information about extending EBOs.

Integration Services

These services are delivered with the query and view account balances integration:

- AccountBalanceSiebelCommsReqABCS
- QueryBalanceSummarySiebelCommsReqABCSImpl
- QueryCustomerPartyListBRMCommsProvABCSImpl
- QueryBalanceGroupListSiebelCommsReqABCSImpl
- QueryBalanceDetailsSiebelCommsReqABCSImpl
- QueryBalanceGroupServicesSiebelCommsReqABCSImpl
- QueryInstalledProductListBRMCommsProvABCSImpl

AccountBalanceSiebelCommsReqABCS

AccountBalanceSiebelCommsReqABCS mediates calls between the subscribers and the provider. AccountBalanceSiebelCommsReqABCS exposes the following operations related to Query and View Account Balances flow on the Siebel ABM.

- QueryBalanceSummary:
  - Routes QueryBalanceSummaryReqMsg to the requester implementation service
  - Routes QueryBalanceSummaryRespMsg to the requester
- QueryBalanceDetails:
  - Routes QueryBalanceDetailsReqMsg to the requester implementation service
  - Routes QueryBalanceDetailsRespMsg to the requester
- QueryBalanceGroupList:
  - Routes QueryBalanceGroupListReqMsg to the requester implementation service
  - Routes QueryBalanceGroupListRespMsg to the requester
- QueryBalanceGroupServices:

- Routes QueryBalanceGroupServicesReqMsg to the requester implementation service
- Routes QueryBalanceGroupServicesRespMsg to the requester

QueryBalanceSummarySiebelCommsReqABCSImpl

QueryBalanceSummarySiebelCommsReqABCSImpl transforms the Siebel message into QueryBalanceSummaryEBM and calls the provider to query the balance summary response from the billing system. It then transforms the EBM response back to a Siebel message and returns it to the calling Siebel web service.

QueryCustomerPartyListBRMCommsProvABCSImpl

QueryCustomerPartyListBRMCommsProvABCSImpl transforms the QueryBalanceSummaryEBM into BRM API input format and calls the API to query the balance summary output from the billing system. It then transforms the output from the API back to a CustomerPartyBalanceEBM message and returns it to the requestor.

QueryBalanceGroupListSiebelCommslReqABCSImpl

The QueryBalanceGroupListSiebelCommslReqABCSImpl transforms the QueryBalanceGroupListReqMsg into QueryCustomerPartyListReqMsgEBM.

QueryBalanceDetailsSiebelCommsReqABCSImpl

The QueryBalanceDetailsSiebelCommsReqABCSImpl is a Business Process Execution Language (BPEL) process that transforms the Siebel message into the QueryBalanceDetailsEBM and calls the provider to query the balance group and balance group balance details response from BRM. It then transforms the EBM response back to a Siebel message and returns it to the calling Siebel web service.

QueryBalanceGroupServicesSiebelCommsReqABCSImpl

QueryBalanceGroupServicesSiebelCommsReqABCSImpl transforms the Siebel message into QueryInstalledProductListEBM and calls the provider to query the balance group list from the billing system. It then transforms the EBM response back to a Siebel message and returns it to the calling Siebel web service.

QueryInstalledProductListBRMCommsProvABCSImpl

QueryInstalledProductListBRMProvABCSImpl transforms QueryInstalledProductListReqMsgEBM to BRM ABM and calls the BRM API, PCM_OP_SEARCH, which takes a query statement involving balance group ID as input and returns the list of service IDs for that balance group.

For each of the service IDs queried, QueryInstalledProductListBRMProvABCSImpl calls the BRM API, PCM_OP_SUBSCRIPTION_GET_PURCHASED_OFFERINGS, which takes a service ID as input and returns the list of product IDs associated with that service.

For each of the product IDs queried, QueryInstalledProductListBRMProvABCSImpl calls the BRM API, PCM_OP_READ_FLDS, which takes a product ID as input and returns the product details for that ID.

About View CDR and Detailed Bills (Invoice)

The view CDR and detailed bills integration between Siebel CRM and BRM supports the following integration scenarios:

- QueryInvoiceList lets you view a list of invoices for an account billing profile in Siebel CRM.
- QueryInvoice lets you view invoice details in Siebel CRM.
- QueryInvoiceUsageAllocation lets you view invoice event details, which is also known as call detail records (CDR) in Siebel CRM.
- SearchInvoiceUsageAllocation lets you search invoice event detail records (CDR) in Siebel CRM.
- QueryInvoiceUsageAllocationResource lets you view nonmonetary resource balance details in Siebel CRM.

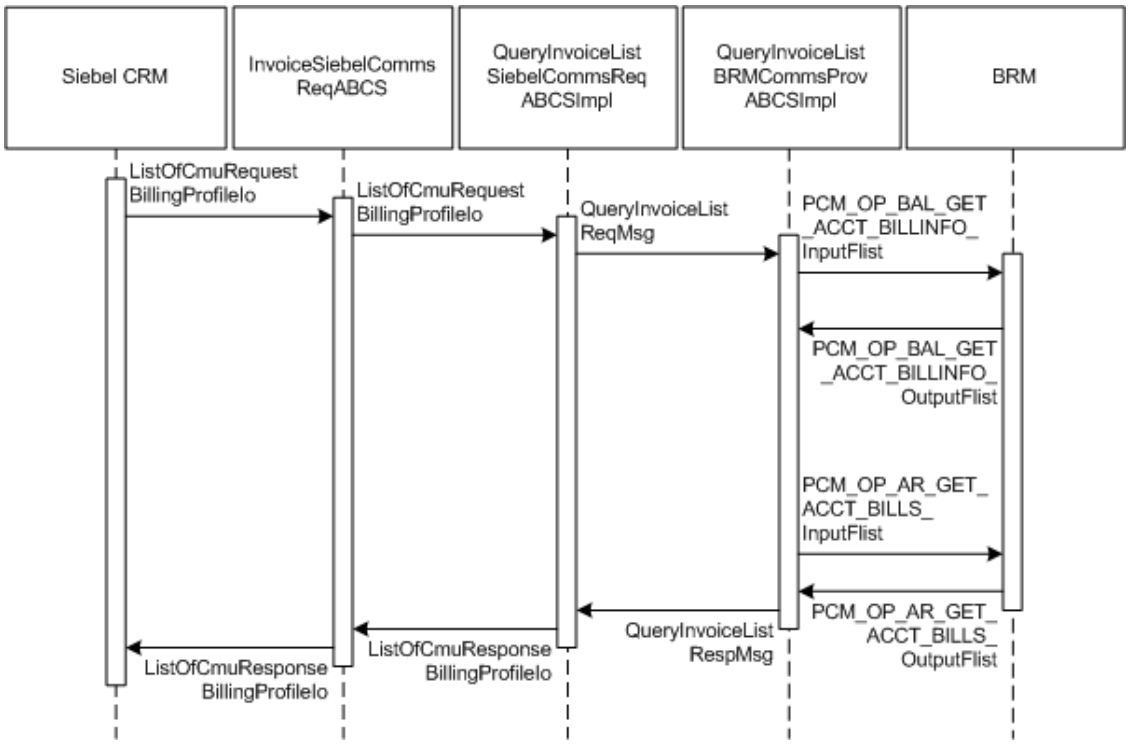
QueryInvoiceList Integration Flow

This integration flow uses the following interfaces:

- InvoiceSiebelCommsReqABCS with operation QueryInvoiceList
- QueryInvoiceListSiebelCommsReqABCSImpl
- QueryInvoiceListBRMCommsProvABCSImpl

Figure 2-12 illustrates the QueryInvoiceList integration scenario.

Figure 2-12 QueryInvoiceList Integration Flow Sequence Diagram



When you initiate the QueryInvoiceList process, the following events occur:

1. In Siebel CRM, a subscriber navigates to the Accounts screen, queries an account, and clicks a billing profile for the account.

   This displays the Billing Profile BRM screen, **Bills** tab, and a web service call is made to query the list of *n* invoices for that billing profile. Many invoices may be in BRM, but the number *n* of invoices to be fetched is sent from Siebel CRM as part of the request

message and <=*n* number of invoices are returned from BRM. The default values of invoices (*n*) queried depends on the configuration in Siebel Integration Object CMU Request Billing Profile IO integration component Com Invoice Profile field Maximum Number Of Records.

2. Navigating to the **Invoice** applet in the Billing Profile BRM screen invokes outbound web service InvoiceSiebelCommsReqABCS, which in turn calls the InvoiceSiebelCommsReqABCS with operation QueryInvoiceList.

   InvoiceSiebelCommsReqABCS is a generic Invoice interface service with several operations defined on the invoice application business message (ABM).

3. Invoking InvoiceSiebelCommsReqABCS with operation QueryInvoiceList routes the QueryInvoiceListReqMsg to the QueryInvoiceListSiebelCommsReqABCSImpl.

4. The QueryInvoiceListSiebelCommsReqABCSImpl first transforms the QueryInvoiceListReqMsg into QueryInvoiceListRequest enterprise business message (EBM) and routes the QueryInvoiceListReqMsg to the appropriate billing systems.

   As delivered, QueryInvoiceListReqMsg is routed to the QueryInvoiceListBRMCommsProvABCSImpl.

5. QueryInvoiceListBRMCommsProvABCSImpl first checks the Query Criteria code.

   If it is Query Invoice List, QueryInvoiceListReqMsg is transformed into PCM_OP_BAL_GET_ACCT_BILLINFO_inputflist. This opcode call returns the list of BILLINFO and AR_BILLINFO of that account.

6. From the response of PCM_OP_BAL_GET_ACCT_BILLINFO opcode, the appropriate BILLINFO and AR_BILLINFO are picked.

   QueryInvoiceListReqMsg is transformed into the input of PCM_OP_AR_GET_ACCT_BILLS and calls the BRM opcode PCM_OP_AR_GET_ACCT_BILLS. Many invoices may be in the billing system for an account billing profile. Based on the value of *n* passed from Siebel CRM, the application programming interface (API) returns <=n number of invoices.

7. QueryInvoiceListBRMCommsProvABCSImpl then transforms the API output PCM_OP_AR_GET_ACCT_BILLS_RespMsg into EBM QueryInvoiceListRespMsg and returns it to QueryInvoiceListSiebelCommsReqABCSImpl.

8. QueryInvoiceListSiebelCommsReqABCSImpl then transforms the QueryInvoiceListRespMsg into QueryInvoiceListRespMsg, which is returned to the InvoiceSiebelCommsReqABCS.

9. InvoiceSiebelCommsReqABCS returns the QueryInvoiceListRespMsg to the calling Siebel web service InvoiceSiebelCommsReqABCSService.

10. The system then writes the list of bills to the Siebel Invoice virtual business component (VBC) for the subscriber.

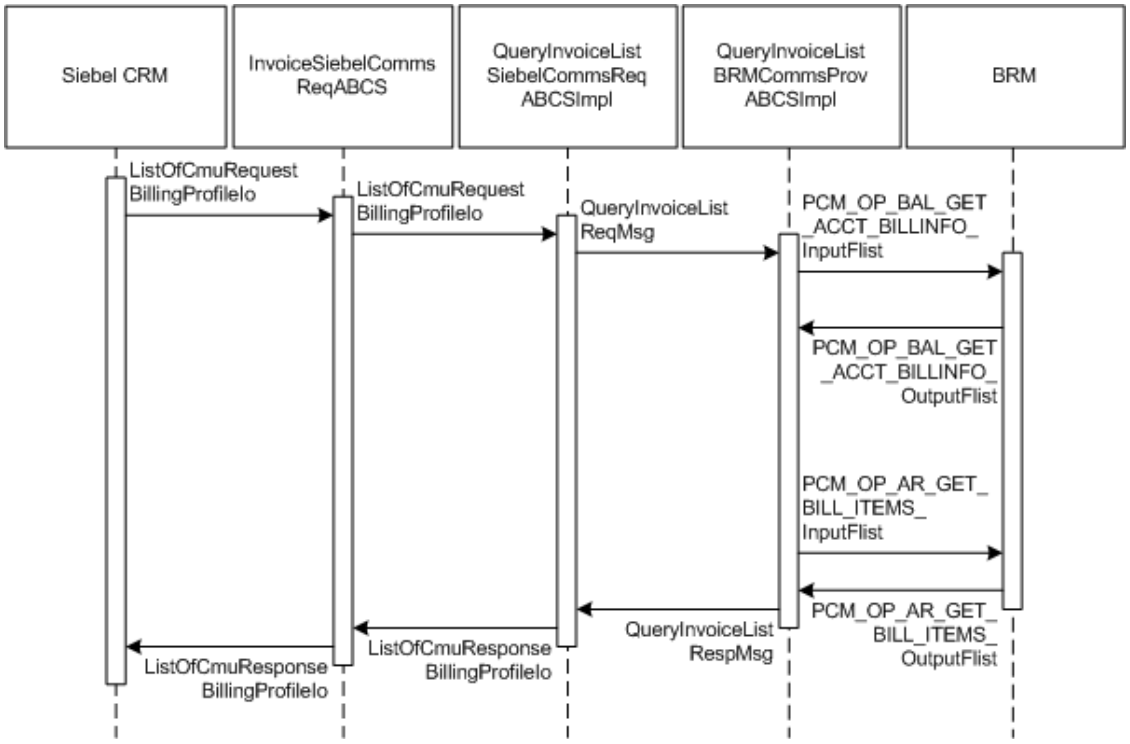
QueryInvoice Integration Flow

This integration flow uses the following interfaces:

- InvoiceSiebelCommsReqABCS with operation QueryInvoice
- QueryInvoiceSiebelCommsReqABCSImpl
- QueryInvoiceListBRMCommsProvABCSImpl

Figure 2-13 illustrates the QueryInvoice integration scenario.

Figure 2-13 QueryInvoice Integration Flow Sequence Diagram



When you initiate the QueryInvoice process, the following events occur:

1. In Siebel CRM, navigate to the Accounts screen, query an account, and click a billing profile for the account.

   This displays the Billing Profile screen. Click the **Bills** tab and drill-down on the Bill Number to call the query invoice information (header, service charges, and items) from the billing system.

2. The InvoiceSiebelCommsReqABCS with the operation QueryInvoice is called.

   The InvoiceSiebelCommsReqABCS is a generic Siebel invoice interface service with several operations defined on the Invoice EBO.

3. The InvoiceSiebelCommsReqABCS with the operation QueryInvoice routes the QueryInvoiceReqMsg to the QueryInvoiceSiebelCommsReqABCSImpl.

4. The QueryInvoiceSiebelCommsReqABCSImpl transforms the QueryInvoiceReqMsg into the Invoice EBM and routes the QueryInvoiceListReqMsg to the appropriate billing system.

   As delivered, QueryInvoiceListReqMsg is routed to the QueryInvoiceListBRMCommsProvABCSImpl.

5. The QueryInvoiceListBRMCommsProvABCSImpl first checks the Query Criteria code. If it is Query Invoice, it then transforms QueryInvoiceListReqMsg into the input of PCM_OP_BAL_GET_ACCT_BILLINFO_inputFlist and invokes PCM_OP_BAL_GET_ACCT_BILLINFO to get the Bill Info object.

   This information is used to populate PCM_OP_AR_GET_BILL_ITEMS_inputFlist and calls the BRM opcode PCM_OP_AR_GET_BILL_ITEMS. The opcode returns the invoice header, service charges, and items in a flat message to the calling QueryInvoiceListBRMCommsProvABCSImpl.

6. The QueryInvoiceListBRMCommsProvABCSImpl transforms the API output PCM_OP_AR_GET_BILL_ITEMS_outputFlist into the EBM QueryInvoiceListRespMsg and returns it to QueryInvoiceSiebelCommsReqABCSImpl.
7. The QueryInvoiceSiebelCommsReqABCSImpl transforms the QueryInvoiceListRespMsg into the QueryInvoiceRespMsg and returns it to the Siebel Invoice ABC interface service.

   The QueryInvoiceListRespMsg is a flat message from which service charges are calculated and the invoice header, service charges, and items are returned as the QueryInvoiceRespMsg to the calling InvoiceSiebelCommsReqABCS.
8. The InvoiceSiebelCommsReqABCS returns the QueryInvoiceRespMsg to the calling Siebel web service.

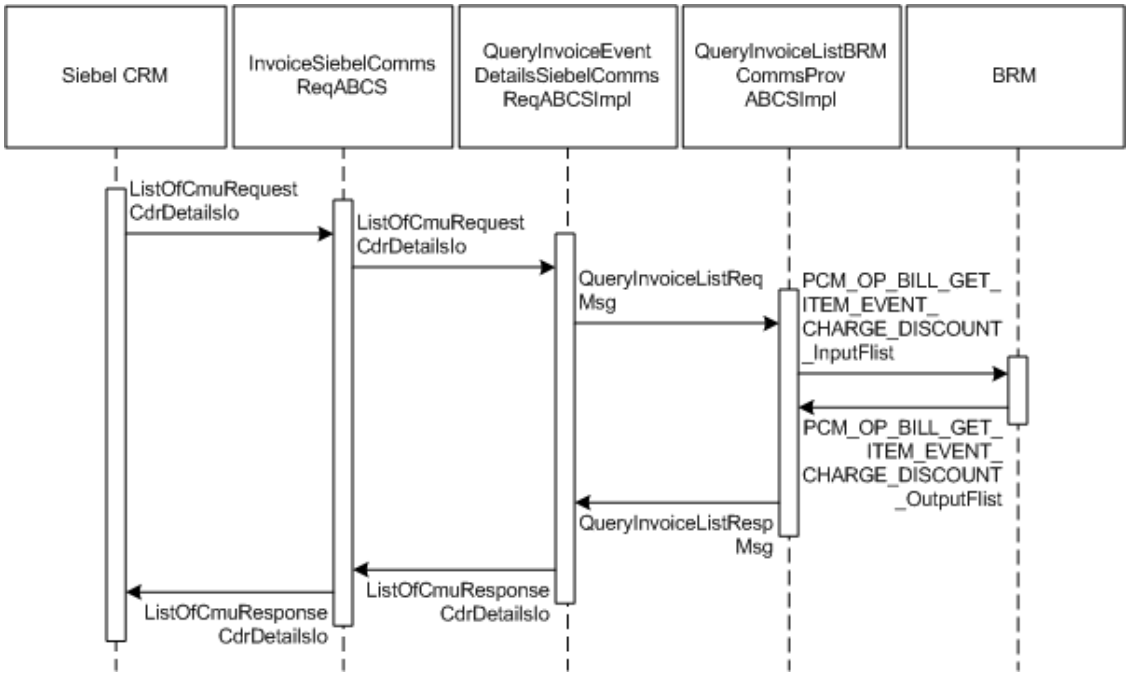
QueryInvoiceUsageAllocation Integration Flow

This integration flow uses the following interfaces:

- InvoiceSiebelCommsReqABCS with operation QueryEventDetails
- QueryInvoiceEventDetailsSiebelCommsReqABCSImpl
- QueryInvoiceListBRMCommsProvABCSImpl

Figure 2-14 illustrates the QueryInvoiceUsageAllocation integration scenario.

Figure 2-14 QueryInvoiceUsageAllocation Integration Flow Sequence



When you initiate the QueryInvoiceUsageAllocation process, the following events occur:

1. In Siebel CRM, navigate to the Accounts screen, query an account, and select the billing profile for the account.

   In the Billing Profile screen, click the **Bills** tab and select an invoice. Select an item charge and click the **Net Amount** link of the item to view **Event Details**. This action calls Siebel outbound web service InvoiceSiebelCommsReqABCS to query the event details from the billing system.

2. Navigating to the **Invoice Event Details** (call detail (CDR) records) applet in the Billing BRM screen invokes the InvoiceSiebelCommsReqABCSService web service with operation QueryEventDetails, which in turn calls the InvoiceSiebelCommsReqABCS with operation QueryEventDetails.

   InvoiceSiebelCommsReqABCS is a generic Siebel Invoice interface service with several operations defined in the Invoice EBO.

3. Invoking InvoiceSiebelCommsReqABCS with operation QueryEventDetails routes the QueryEventDetailsReqMsg to the QueryInvoiceEventDetailsSiebelCommsReqABCSImpl.

4. The QueryInvoiceEventDetailsSiebelCommsReqABCSImpl transforms the QueryEventDetailsReqMsg into Invoice EBM and routes the QueryInvoiceListReqMsg to the appropriate billing system.

   As delivered, QueryInvoiceListReqMsg is routed to the QueryInvoiceListBRMCommsProvABCSImpl.

5. QueryInvoiceListBRMCommsProvABCSImpl checks query criteria code.

   If it is Query Usage Allocation, then it transforms QueryInvoiceListReqMsg into the input of PCM_OP_ BILL_GET_ITEM_EVENT_CHARGE_DISCOUNT and calls the opcode PCM_OP_ BILL_GET_ITEM_EVENT_CHARGE_DISCOUNT.

6. QueryInvoiceListBRMCommsProvABCSImpl then transforms the API output PCM_OP_ BILL_GET_ITEM_EVENT_CHARGE_DISCOUNT_RespMsg into EBM QueryInvoiceListRespMsg and returns it to QueryInvoiceEventDetailsSiebelCommsReqABCSImpl.

7. QueryInvoiceEventDetailsSiebelCommsReqABCSImpl transforms the QueryInvoiceListRespMsg into QueryEventDetailsRespMsg, which is returned to InvoiceSiebelCommsReqABCS.

8. InvoiceSiebelCommsReqABCS returns the QueryEventDetailsRespMsg to the calling Siebel web service.

9. The system then writes the response message to the Siebel Invoice VBC for the subscriber.

SearchInvoiceUsageAllocation Integration Flow

This integration flow uses the following interfaces:

- InvoiceSiebelCommsReqABCS with operation SearchEventDetails
- SearchInvoiceEventDetailsSiebelCommsReqABCSImpl
- QueryInvoiceListBRMCommsProvABCSImpl

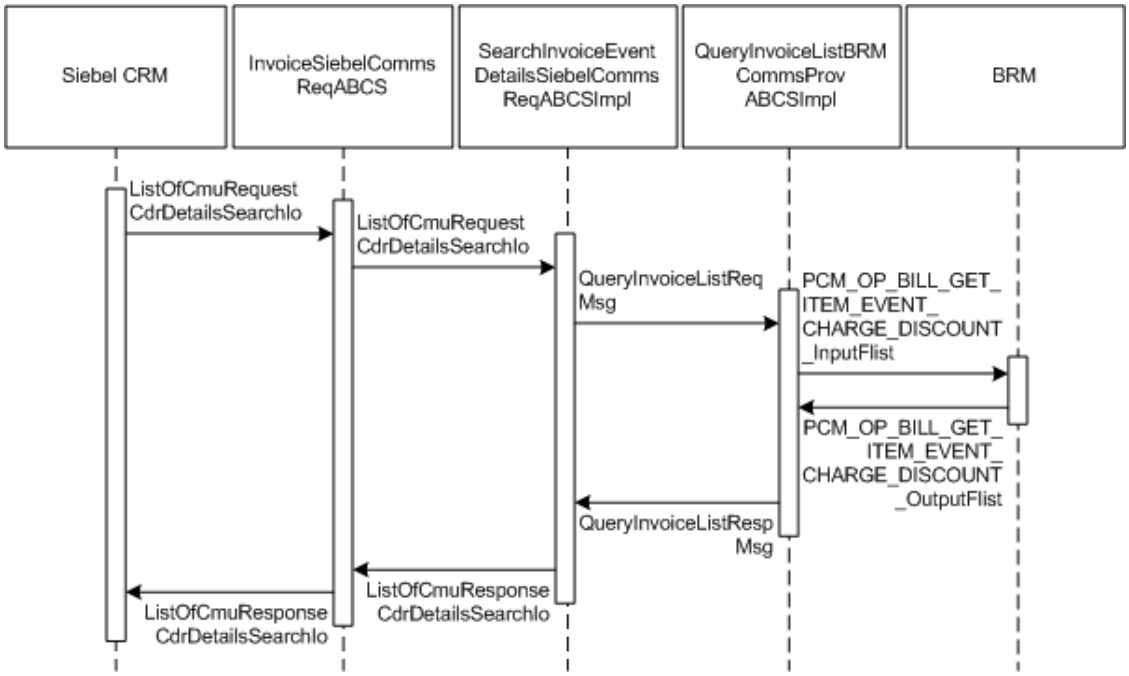
The SearchInvoiceUsageAllocation supports searching invoice event details (CDR records) on the following columns in addition to the account ID and the item charge ID that are passed to Query Invoice Details:

- Minimum Amount
- Maximum Amount
- Start Date
- End Date

The BRM opcode PCM_OP_BILL_GET_ITEM_EVENT_CHARGE_DISCOUNT supports only simple queries, so the search supports passing only the data through and not passing complex query criteria, (for example, >, <, between, and so on).

Figure 2-15 illustrates the SearchInvoiceUsageAllocation integration scenario.

Figure 2-15 SearchInvoiceUsageAllocation Integration Flow Sequence Diagram



When you initiate the SearchInvoiceUsageAllocation process, the following events occur:

1. In Siebel CRM, navigate to the Accounts screen, query an account, and select the billing profile for the account.

   In the Billing Profile screen, click the **Bills** tab and select an invoice. Select an item charge, and click the **Net Amount** link of the item to view event details of all CDR records for that item charge.

2. Clicking the **Search** button on the **Invoice Event Details** (CDR details) applet opens the search applet for query.

3. Entering the search criteria and clicking **Go** invokes an outbound web service InvoiceSiebelReqABCS that in turn calls the InvoiceSiebelCommsReqABCS with operation SearchEventDetails.

   InvoiceSiebelCommsReqABCS is a generic Siebel Invoice interface service with several operations defined on the Invoice EBO.

4. Invoking InvoiceSiebelCommsReqABCS with operation SearchEventDetails routes the SearchEventDetailsReqMsg to the SearchInvoiceEventDetailsSiebelCommsReqABCSImpl.

5. The SearchInvoiceEventDetailsSiebelCommsReqABCSImpl transforms the SearchEventDetailsReqMsg into an Invoice EBM and routes the QueryInvoiceListReqMsg to the appropriate billing systems.

   As delivered, QueryInvoiceListReqMsg is routed to the QueryInvoiceListBRMCommsProvABCSImpl.

6. QueryInvoiceListBRMCommsProvABCSImpl checks the query criteria code. If the query criteria code is Usage Allocation, it then transforms QueryInvoiceListReqMsg into the input

of PCM_OP_ BILL_GET_ITEM_EVENT_CHARGE_DISCOUNT and calls the BRM opcode PCM_OP_ BILL_GET_ITEM_EVENT_CHARGE_DISCOUNT.

7. QueryInvoiceListBRMCommsProvABCSImpl transforms the API output PCM_OP_ BILL_GET_ITEM_EVENT_CHARGE_DISCOUNT_RespMsg into EBM QueryInvoiceListRespMsg and returns it to SearchInvoiceEventDetailsSiebelCommsReqABCSImpl.
8. SearchInvoiceEventDetailsSiebelCommsReqABCSImpl transforms the QueryInvoiceListRespMsg into a SearchEventDetailsRespMsg, which is returned to InvoiceSiebelCommsReqABCS.
9. InvoiceSiebelCommsReqABCS returns the SearchEventDetailsRespMsg to the calling Siebel web service.
10. The system writes the search response to the Siebel Invoice VBC for the subscriber.

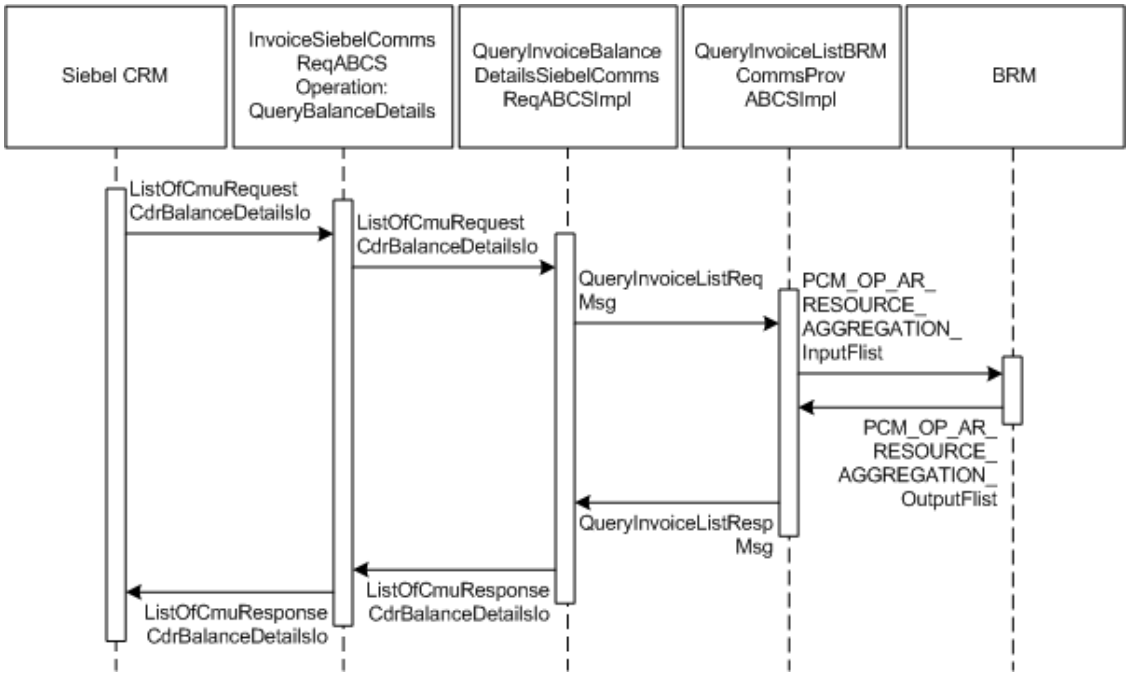
QueryInvoiceUsageAllocationResource Integration Flow

This integration flow uses the following interfaces:

- InvoiceSiebelCommsReqABCS with operation QueryBalanceDetails
- QueryInvoiceBalanceDetailsSiebelCommsReqABCSImpl
- QueryInvoiceListBRMCommsProvABCSImpl

Figure 2-16 illustrates the QueryInvoiceUsageAllocationResource integration scenario.

Figure 2-16 QueryInvoiceUsageAllocationResource Integration Flow Sequence Diagram



When you initiate the QueryInvoiceUsageAllocationResource process, the following events occur:

1. In Siebel CRM, navigate to the Accounts screen, query an account, and select the billing profile for the account.

In the Billing Profile screen, click the **Bills** tab and select an invoice. In **Event Details**, click **View Detail** to query the resource impact details for an event from the billing system. A web service call is made to query resource impact details for a particular event from the billing system, which in turn calls the InvoiceSiebelCommsReqABCS with operation QueryResourceBalance.

InvoiceSiebelCommsReqABCS is a generic Siebel Invoice interface service with several operations defined on the Invoice EBO.

2. Invoking InvoiceSiebelCommsReqABCS with operation QueryBalanceDetails routes the QueryBalanceDetailsReqMsg to the QueryInvoiceBalanceDetailsSiebelCommsReqABCSImpl.
3. The QueryInvoiceBalanceDetailsSiebelCommsReqABCSImpl transforms the QueryBalanceDetailsReqMsg into Invoice EBM and routes the QueryInvoiceListReqMsg to the appropriate billing system.

   As delivered, QueryInvoiceListReqMsg is routed to the QueryInvoiceListBRMCommsProvABCSImpl.
4. QueryInvoiceListBRMCommsProvABCSImpl checks the query criteria code.

   If the query criteria code is Query Balance Details, it transforms QueryInvoiceListReqMsg into the input of PCM_OP_ AR_RESOURCE_AGGREGATION and calls the opcode PCM_OP_ AR_RESOURCE_AGGREGATION.
5. QueryInvoiceListBRMCommsProvABCSImpl transforms the API output PCM_OP_ AR_RESOURCE_AGGREGATION_RespMsg into EBM QueryInvoiceListRespMsg and returns it to QueryInvoiceBalanceDetailsSiebelCommsReqABCSImpl.
6. QueryInvoiceBalanceDetailsSiebelCommsReqABCSImpl transforms the QueryInvoiceListRespMsg into QueryBalanceDetailsRespMsg, which is returned to the InvoiceSiebelCommsReqABCS.
7. InvoiceSiebelCommsReqABCS returns the QueryBalanceDetailsRespMsg back to the calling Siebel web service.
8. The system writes the message to the Siebel UsageAllocationResource VBC for the subscriber.

BRM Interfaces

The integration uses opcodes with the integration flows as shown in Table 2-1.

Table 2-1 View CDR and Detailed Bills Flow Opcodes

Opcode	Integration Flow
PCM_OP_AR_GET_ACCT_BIILS	QueryInvoiceList
PCM_ OP_AR_GET_BILL_ITEMS	QueryInvoice
PCM_OP_BILL_GET_ITEM_EVENT_CHARGE_DISCOUNT	QueryInvoiceUsageAllocation
PCM_OP_BILL_GET_ITEM_EVENT_CHARGE_DISCOUNT	SearchInvoiceUsageAllocation
PCM_OP_AR_RESOURCE_AGGREGATION_RespMsg	QueryInvoiceUsageAllocationResource

See *Oracle Communications Billing and Revenue Management Opcode Flist Reference* and *Oracle Communications Billing and Revenue Management Developer's Guide* for more information about individual opcodes.

Siebel CRM Interfaces

The View CDR and Detailed Bills flow uses the following Siebel CRM interface:

- InvoiceSiebelCommsReqABCS

See the *Siebel Order Management Guide Addendum for Communications* for more information about individual web services.

Industry Oracle AIA Components

The View CDR and Detailed Bills flow uses the following delivered enterprise business object (EBO) and enterprise business messages (EBMs):

- InvoiceEBO
- QueryInvoiceEBM
- QueryInvoiceResponseEBM
- QueryInvoiceListEBM
- QueryInvoiceListResponseEBM

The following directories contain the industry component files:

- Enterprise business object (EBO) and enterprise business message (EBM) XML schema files:

  COMMS_AIA_HOME/**comms_home/source/soainfra/apps/AIAMetaData/AIAComponents/EnterpriseObjectLibrary/Industry/Communications/EBO/**

- Enterprise business service (EBS) WSDL files:

  COMMS_AIA_HOME/**comms_home/source/soainfra/apps/AIAMetaData/AIAComponents/EnterpriseBusinessServiceLibrary/Industry/Communications/EBO/**

For detailed documentation of individual EBOs and EBMs, click the AIA Reference Doc link on EBO and EBM detail pages in Oracle Enterprise Repository (OER).

EBOs can be extended such as adding new data elements. These extensions are protected and will remain intact even after a patch or an upgrade, so long as the extensibility guidelines are followed.

See the discussion of Oracle AIA assets extensibility patterns in *Oracle Fusion Middleware Developer's Guide for Oracle SOA Core Extension* for more information about extending EBOs.

Integration Services

These services are delivered with the View CDR and Detailed Bills flow:

- InvoiceSiebelCommsReqABCS with operations QueryInvoiceList, QueryInvoice, QueryEventDetails, SearchEventDetails, and QueryBalanceDetails
- QueryInvoiceListSiebelCommsReqABCSImpl
- QueryInvoiceListBRMCommsProvABCSImpl
- QueryInvoiceSiebelCommsReqABCSImpl
- QueryInvoiceEventDetailsSiebelCommsReqABCSImpl
- SearchInvoiceEventDetailsSiebelCommsReqABCSImpl

- QueryInvoiceBalanceDetailsSiebelCommsReqABCSImpl

InvoiceSiebelCommsReqABCS

InvoiceSiebelCommsReqABCS exposes the following operations related to the Invoice integration on the Siebel ABM:

- QueryInvoiceList:
  - Routes QueryInvoiceListReqMsg to the requester implementation service
  - Routes QueryInvoiceListRespMsg to the requester
- QueryInvoice:
  - Routes QueryInvoiceReqMsg to the requester implementation service
  - Routes QueryInvoiceRespMsg to the requester
- QueryEventDetails:
  - Routes QueryEventDetailsReqMsg to the requester implementation service
  - Routes QueryEventDetailsRespMsg to the requester
- SearchEventDetails:
  - Routes SearchEventDetailsReqMsg to the requester implementation service
  - Routes SearchEventDetailsRespMsg to the requester
- QueryBalanceDetails:
  - Routes QueryBalanceDetailsReqMsg to the requester implementation service
  - Routes QueryBalanceDetailsRespMsg to the requester

QueryInvoiceListSiebelCommsReqABCSImpl

QueryInvoiceListSiebelCommsReqABCSImpl transforms the Siebel message into a QueryInvoiceList EBM and calls the provider to query the invoice list response from the billing system. It then transforms the EBM response back to a Siebel message and returns it to the calling Siebel web service.

QueryInvoiceListBRMCommsProvABCSImpl

QueryInvoiceListBRMCommsProvABCSImpl transforms:

- QueryInvoiceListRequestEBM into BRM API input format and calls the API to query the invoice list output from the billing system.

  It then transforms the output from the API back to an Invoice EBM message and returns it to the calling requestor.
- QueryInvoice EBM into BRM API input formats and calls the APIs to Query the Invoice output from the billing system.

  It then transforms the output from the APIs back to an Invoice EBM message and returns it to the calling requestor.
- QueryInvoiceList EBM into BRM API input formats and calls the APIs to Query the Invoice Event Details output from the billing system.

  It then transforms the output from the APIs back to an Invoice EBM message and returns it to the calling requestor.

- Invoice EBM into BRM API input formats and calls the APIs to Query the Resource Impact output from the billing system.

  It then transforms the output from the APIs back to an Invoice EBM message and returns it to the calling requestor.

QueryInvoiceSiebelCommsReqABCSImpl

QueryInvoiceSiebelCommsReqABCSImpl transforms the Siebel message into QueryInvoiceEBM and calls the provider to query the invoice from the billing system. It then transforms the EBM response back to a Siebel message and returns it to the calling Siebel web service.

QueryInvoiceEventDetailsSiebelCommsReqABCSImpl

QueryInvoiceEventDetailsSiebelCommsReqABCSImpl transforms the Siebel message into QueryInvoiceList EBM and calls the provider to query the invoice event details response from the billing system. It then transforms the EBM response back to a Siebel message and returns it to the calling Siebel web service.

SearchInvoiceEventDetailsSiebelCommsReqABCSImpl

SearchInvoiceEventDetailsSiebelCommsReqABCSImpl transforms the Siebel message into QueryInvoiceList EBM and calls the provider to query the Invoice Event Details response from the billing system. It then transforms the EBM response back to a Siebel message and returns it to the calling Siebel web service.

SearchInvoiceCharge supports searching invoice event details (CDR records) on the following columns in addition to the account ID and the item charge ID that are passed to Query Invoice Details:

- Minimum Amount
- Maximum Amount
- Start Date
- End Date

The BRM opcode PCM_OP_ BILL_GET_ITEM_EVENT_CHARGE_DISCOUNT supports only simple queries, so the search supports passing only the data through and not passing complex query criteria (for example: >, <, between, and so on).

QueryInvoiceBalanceDetailsSiebelCommsReqABCSImpl

QueryInvoiceBalanceDetailsSiebelCommsReqABCSImpl transforms the Siebel message into an Invoice EBM and calls the provider to query the Resource Impact response from the billing system. It then transforms the EBM response back to a Siebel message and returns it to the calling Siebel web service.

About View Unbilled Usage

The Unbilled Usage integration between Siebel CRM and BRM supports the following integration scenarios:

- QueryServiceUsage enables a Customer Service Representative (CSR) to view account-billing-profile-service-usage-summary in Siebel CRM.

- QueryServiceUsageAllocation enables a CSR to view account-billing-profile-service-usage-charge-details in Siebel CRM.
- SearchServiceUsageAllocation enables a CSR to search account-billing-profile-service-usage-charge-details based on a few columns.
- QueryServiceUsageAllocationResource enables a CSR to view resource balances (for example, nonmonetary) for service usage events in Siebel CRM.

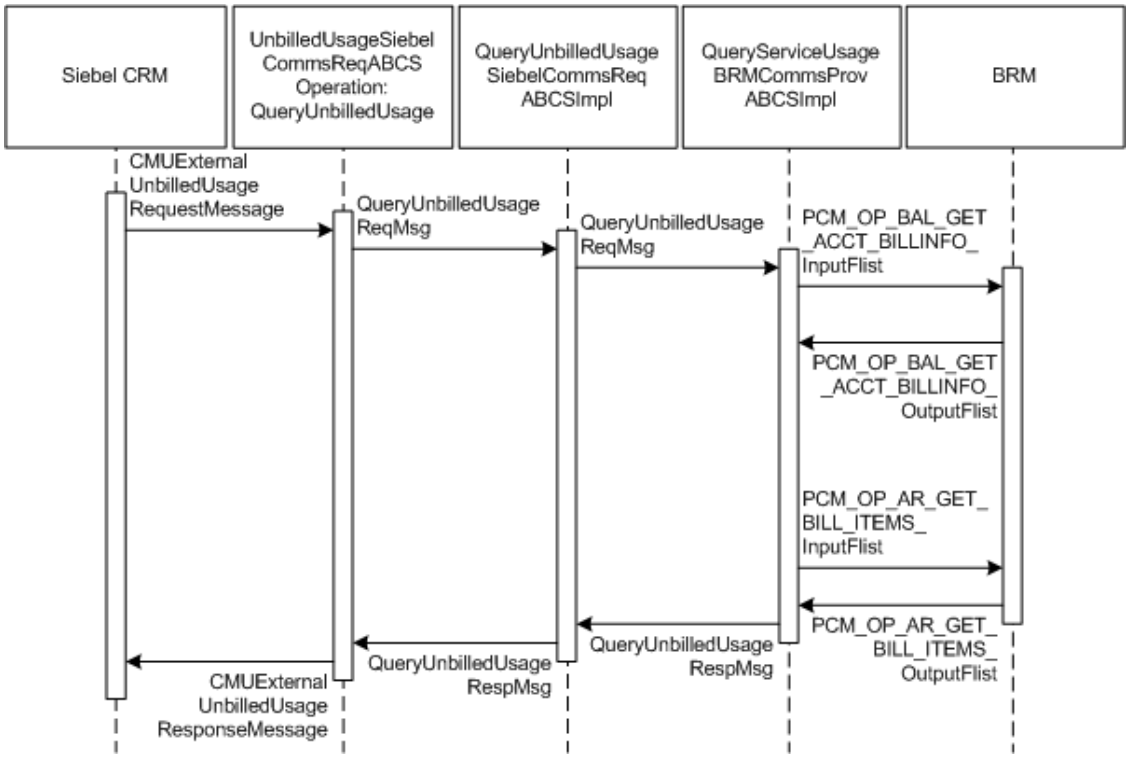
QueryServiceUsage Integration Flow

This integration flow uses the following interfaces:

- UnbilledUsageSiebelCommsReqABCS with operation QueryUnbilledUsage
- QueryUnbilledUsageSiebelCommsReqABCSImpl
- QueryServiceUsageBRMCommsProvABCSImpl

Figure 2-17 illustrates the QueryServiceUsage integration scenario.

Figure 2-17 QueryServiceUsage Integration Flow Sequence Diagram



When you initiate the QueryServiceUsage process, the following events occur:

1. In Siebel CRM, navigate to the Accounts screen, query an account, and click a billing profile for the account.

   This displays the Billing Profile BRM screen. When you click the **Unbilled Usage** tab for the billing profile, the Unbilled Usage screen opens and a web service UnbilledUsageSiebelCommsReqABCS call is made to get the unbilled usage details from the billing system.

2. Navigating to the **Unbilled Usage** tab invokes the UnbilledUsageSiebelCommsReqABCS web service, which in turn calls the UnbilledUsageSiebelCommsReqABCS with operation QueryUnbilledUsage.

   UnbilledUsageSiebelCommsReqABCS is a generic Siebel UnbilledUsage interface service with several operations defined.

3. Invoking UnbilledUsageSiebelCommsReqABCS with operation QueryUnbilledUsage routes the Siebel QueryUnbilledUsageReqMsg to the QueryUnbilledUsageSiebelCommsReqABCSImpl.

4. The QueryUnbilledUsageSiebelCommsReqABCSImpl first transforms the QueryUnbilledUsageReqMsg into QueryServiceUsageEBM and routes the QueryServiceUsageListReqMsg to the appropriate billing systems.

   As delivered, QueryServiceUsageListReqMsg is routed to the QueryServiceUsageListBRMCommsProvABCSImpl.

5. QueryServiceUsageBRMCommsProvABCSImpl looks up QueryCode in enterprise business message (EBM) and transforms QueryServiceUsageListReqMsg into the input of PCM_OP_BAL_GET_ACCT_BILLINFO and calls the BRM opcode PCM_OP_BAL_GET_ACCT_BILLINFO.

6. The BRM application programming interface (API) PCM_OP_BAL_GET_ACCT_BILLINFO returns the list of billinfos of that account along with their AR_Billinfo to the calling QueryServiceUsageListBRMCommsProvABCSImpl.

7. This information is used to populate PCM_OP_AR_GET_BILL_ITEMS_inputFlist and calls the BRM opcode PCM_OP_AR_GET_BILL_ITEMS. The opcode returns the invoice header, service charges, and items in a flat message.

8. QueryServiceUsageBRMCommsProvABCSImpl then transforms the BRM API PCM_OP_AR_GET_BILL_ITEMS output into QueryServiceUsageListRespMsg and returns it to QueryUnbilledUsageSiebelCommsReqABCSImpl.

9. QueryUnbilledUsageSiebelCommsReqABCSImpl transforms the QueryUnbilledUsageRespMsg into QueryUnbilledUsageRespMsg, which is returned to UnbilledUsageSiebelCommsReqABCS.

10. UnbilledUsageSiebelCommsReqABCS returns the QueryUnbilledUsageRespMsg to the calling CMUUnbilledUsage as CMUUnbilledUsageResponseMessage.

11. The system writes the CMUUnbilledUsageResponseMessage to the Siebel unbilled usage Details virtual business component (VBC) for the subscriber.

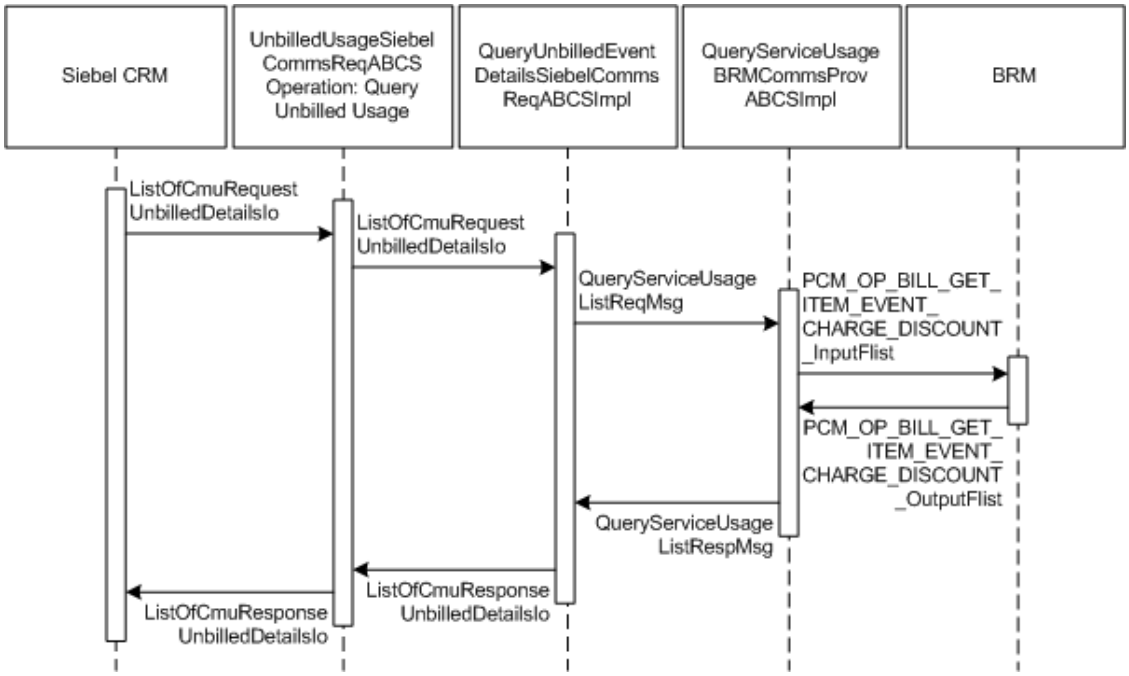
QueryServiceUsageAllocation Integration Flow

This integration flow uses the following interfaces:

- UnbilledUsageSiebelCommsReqABCS with operation QueryEventDetails
- QueryUnbilledEventDetailsSiebelCommsReqABCSImpl
- QueryServiceUsageBRMCommsProvABCSImpl

Figure 2-18 illustrates the QueryServiceUsageAllocation integration scenario.

Figure 2-18 QueryServiceUsageAllocation Integration Flow Sequence Diagram



When you initiate the QueryServiceUsageAllocation process, the following events occur:

1. In Siebel CRM, navigate to the Accounts screen, query an account, and click the **Billing Profile** tab.

   This displays the Billing Profile screen. Click the **Unbilled Usage** tab to open the Unbilled Usage screen.

2. Clicking the **Net Amount** link for a particular item charge invokes the UnbilledUsageSiebelCommsReqABCS web service, which in turn calls the UnbilledUsageSiebelCommsReqABCS with operation QueryEventDetails.

   UnbilledUsageSiebelCommsReqABCS is a generic Siebel UnbilledUsage interface service with several operations defined.

3. Invoking UnbilledUsageSiebelCommsReqABCS with operation QueryEventDetails routes the Siebel QueryEventDetailsReqMsg to the QueryUnbilledEventDetailsSiebelCommsReqABCSImpl.

4. The QueryUnbilledEventDetailsSiebelCommsReqABCSImpl transforms the QueryEventDetailsReqMsg into QueryServiceUsageListEBM and routes the QueryServiceUsageListReqMsg to the appropriate billing systems.

   As delivered, QueryServiceUsageListReqMsg is routed to the QueryServiceUsageListBRMCommsProvABCSImpl.

5. QueryServiceUsageBRMCommsProvABCSImpl looks up the value of QueryCode and transforms QueryServiceUsageListReqMsg into the input of PCM_OP_BILL_GET_ITEM_EVENT_CHARGE_DISCOUNT and calls the opcode PCM_OP_BILL_GET_ITEM_EVENT_CHARGE_DISCOUNT.

6. API PCM_OP_BILL_GET_ITEM_EVENT_CHARGE_DISCOUNT returns the Item Charge Details or Events output to the calling QueryServiceUsageBRMCommsProvABCSImpl.

7. QueryServiceUsageListBRMCommsProvABCSImpl then transforms the BRM API output into QueryServiceUsageListRespMsg and returns it to QueryUnbilledEventDetailsSiebelCommsReqABCSImpl.
8. QueryUnbilledEventDetailsSiebelCommsReqABCSImpl transforms the QueryServiceUsageListRespMsg into QueryEventDetailsRespMsg, which is returned to UnbilledUsageSiebelCommsReqABCS.
9. UnbilledUsageSiebelCommsReqABCS returns the QueryEventDetailsRespMsg to the calling UnbilledUsageSiebelCommsReqABCS as CMUUnbilledDetailsResponseMessage.
10. The system writes the CMUUnbilledDetailsResponseMessage to the Siebel Unbilled Item Charge Details VBC for the subscriber.

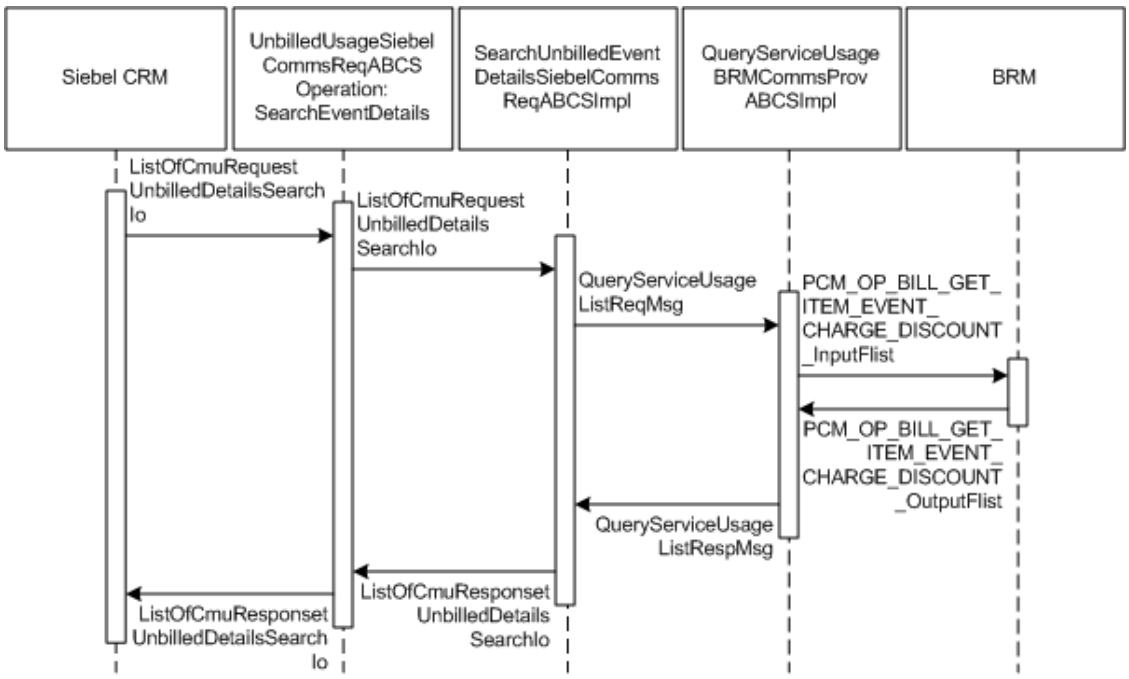
SearchServiceUsageAllocation Integration Flow

This integration flow uses the following interfaces:

- UnbilledUsageSiebelCommsReqABCS with operation SearchEventDetails
- SearchUnbilledEventDetailsSiebelCommsReqABCSImpl
- QueryServiceUsageBRMCommsProvABCSImpl

Figure 2-19 illustrates the SearchServiceUsageAllocation integration scenario.

Figure 2-19 SearchServiceUsageAllocation Integration Flow Sequence Diagram



When you initiate the SearchServiceUsageAllocation process, the following events occur:

1. In Siebel CRM, navigate to the Accounts screen, query an account, and click the **Billing Profile** tab.

   This displays the Billing Profile screen. Click the **Unbilled Usage** tab to access the Unbilled Usage screen. Click the **Net Amount** link for an item charge. The event details for that item charge displays on the **Event Details** applet. Click the **Search** button to open a

separate applet to specify search criteria. After entering values for these fields, click the **Go** button.

2. Clicking **Go** invokes the UnbilledUsageSiebelCommsReqABCS web service, which in turn calls UnbilledUsageSiebelCommsReqABCS with operation SearchEventDetails.

   UnbilledUsageSiebelCommsReqABCS is a generic Siebel UnbilledUsage interface service with several operations defined.

3. Invoking UnbilledUsageSiebelCommsReqABCS with operation SearchEventDetails routes the Siebel SearchEventDetailsReqMsg to SearchUnbilledEventDetailsSiebelCommsReqABCSImpl.

4. SearchUnbilledEventDetailsSiebelCommsReqABCSImpl transforms SearchEventDetailsReqMsg into QueryServiceUsageEBM and routes the QueryServiceUsageListReqMsg to the appropriate billing systems.

   As delivered, QueryServiceUsageListReqMsg is routed to QueryServiceUsageListBRMCommsProvABCSImpl.

5. QueryServiceUsageBRMCommsProvABCSImpl transforms QueryServiceUsageListReqMsg into the input of PCM_OP_BILL_GET_ITEM_EVENT_CHARGE_DISCOUNT and calls the BRM opcode PCM_OP_BILL_GET_ITEM_EVENT_CHARGE_DISCOUNT.

6. The BRM API PCM_OP_BILL_GET_ITEM_EVENT_CHARGE_DISCOUNT returns the Item Charge Details output to the calling QueryServiceUsageBRMCommsProvABCSImpl.

7. QueryServiceUsageBRMCommsProvABCSImpl then transforms the BRM API output into QueryServiceUsageListRespMsg and returns it to SearchUnbilledUsageEventDetailsSiebelCommsReqABCSImpl.

8. SearchUnbilledEventDetailsSiebelCommsReqABCSImpl then transforms the QueryServiceUsageListRespMsg into SearchEventDetailsRespMsg, which is returned to UnbilledUsageSiebelCommsReqABCS.

9. UnbilledUsageSiebelCommsReqABCS returns the SearchEventDetailsRespMsg to the calling Siebel web service UnbilledUsageSiebelCommsReqABCS as CMUUnbilledDetailsSearchResponseMessage.

10. The system writes the CMUUnbilledDetailsSearchResponseMessage to the Siebel Unbilled Item Charge Details VBC for the subscriber.

SearchServiceUsageAllocation supports searching service usage item details (CDR records) on the following columns in addition to the account ID and item charge ID that are passed to SearchEventDetails:

- Minimum Amount
- Maximum Amount
- Start Date
- End Date

The BRM opcode PCM_OP_ BILL_GET_ITEM_EVENT_CHARGE_DISCOUNT supports only simple queries and passes only the data through. It does not pass complex query criteria (for example, >, <, between, and so on).

QueryServiceUsageAllocationResource Integration Flow

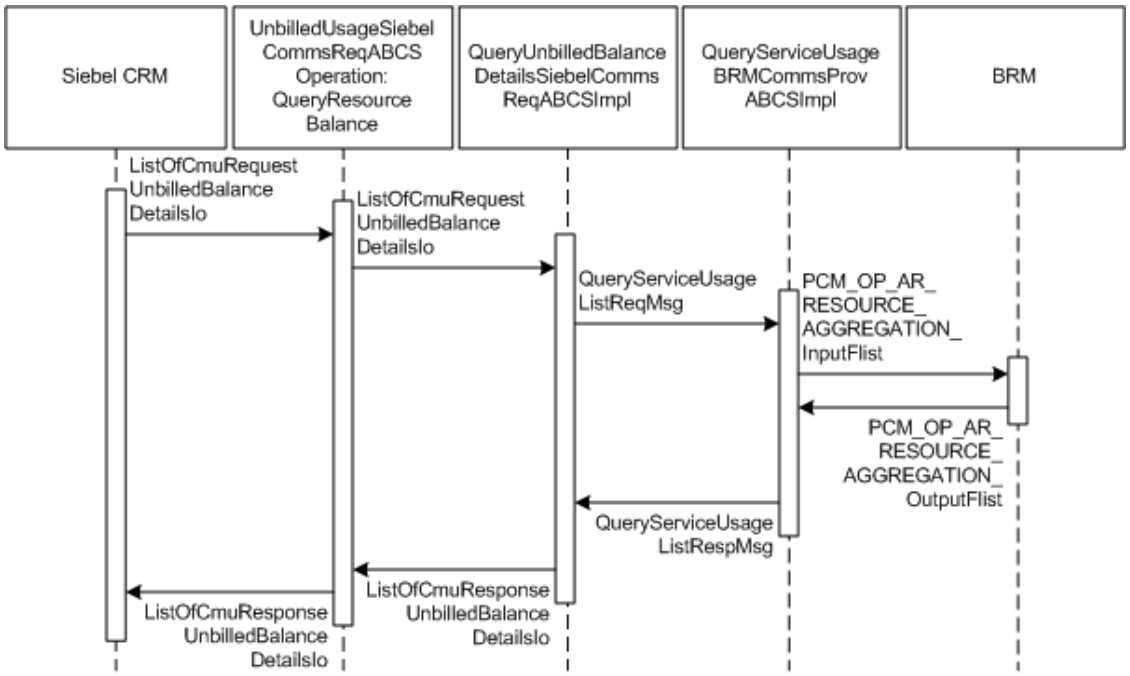
This integration flow uses the following interfaces:

- UnbilledUsageSiebelCommsReqABCS with operation QueryBalanceDetails

- QueryUnbilledBalanceDetailsSiebelCommsReqABCSImpl
- QueryServiceUsageBRMCommsProvABCSImpl

Figure 2-20 illustrates the QueryServiceUsageAllocationResource integration scenario.

Figure 2-20 QueryServiceUsageAllocationResource Integration Flow Sequence Diagram



When you initiate the QueryServiceUsageAllocationResource process, the following events occur:

1. In Siebel CRM Query Account, navigate to the Account Summary screen, and drill down on the **Billing Profile Name** in the **Billing Profile** applet.

   This displays the Billing Profile screen. Click the **Unbilled Usage** tab, and then click the **Net Amount** link for an item charge. This accesses the **Event Details** applet for that item charge.

2. Clicking **View Details** for a particular nonmonetary event invokes the outbound web service UnbilledUsageSiebelCommsReqABCS to query resource balance details for a particular event from the billing system, which in turn calls UnbilledUsageSiebelCommsReqABCS with operation QueryResourceBalance.

3. Invoking UnbilledUsageSiebelCommsReqABCS with operation QueryBalanceDetails routes the QueryBalanceDetailsReqMsg to QueryUnbilledBalanceDetailsSiebelCommsReqABCSImpl.

4. QueryUnbilledBalanceDetailsSiebelCommsReqABCSImpl transforms the QueryBalanceDetailsReqMsg into a Service Usage EBM and routes the QueryServiceUsageListReqMsg to the appropriate billing systems.

   As delivered, QueryServiceUsageListReqMsg is routed to QueryServiceUsageListBRMCommsProvABCSImpl.

5. QueryServiceUsageBRMCommsProvABCSImpl transforms QueryServiceUsageListReqMsg into the input of PCM_OP_

AR_RESOURCE_AGGREGATION and calls the BRM API PCM_OP_ AR_RESOURCE_AGGREGATION.

6. The BRM API PCM_OP_ AR_RESOURCE_AGGREGATION returns the resource balance records for a particular event as part of PCM_OP_AR_RESOURCE_AGGREGATION_outputFlist to QueryServiceUsageBRMCommsProvABCSImpl.
7. QueryServiceUsageBRMCommsProvABCSImpl then transforms the API output PCM_OP_ AR_RESOURCE_AGGREGATION_RespMsg into QueryServiceUsageListRespMsg and returns it to QueryUnbilledUsageBalanceDetailsSiebelCommsReqABCSImpl.
8. QueryUnbilledBalanceDetailsSiebelCommsReqABCSImpl transforms the QueryServiceUsageListRespMsg into QueryBalanceDetailsRespMsg, which is returned to UnbilledUsageSiebelCommsReqABCS.
9. UnbilledUsageSiebelCommsReqABCS returns the QueryBalanceDetailsRespMsg to the calling Siebel web service.
10. The system writes the message to the Siebel Balance Total VBC for the subscriber.

BRM Interfaces

The QueryServiceUsage integration flow uses this opcode:

- PCM_OP_BAL_GET_ACCT_BILLINFO

The QueryServiceUsageAllocation integration flow uses this opcode:

- PCM_OP_BILL_GET_ITEM_EVENT_CHARGE_DISCOUNT

The SearchServiceUsageAllocation integration flow uses this opcode:

- PCM_OP_BILL_GET_ITEM_EVENT_CHARGE_DISCOUNT

The QueryServiceUsageAllocationResource integration flow uses this opcode:

- PCM_OP_AR_RESOURCE_AGGREGATION

See *Oracle Communications Billing and Revenue Management Opcode Flist Reference* for more information.

Siebel CRM Interfaces

The Unbilled Usage flow uses this Siebel CRM interface:

- UnbilledUsageSiebelCommsReqABCS

See *Siebel Order Management Guide Addendum for Communications* for more information about this web service.

Industry Oracle AIA Components

The Unbilled Usage flow uses the following delivered Industry Oracle AIA components:

- ServiceUsageEBO
- QueryServiceUsageListEBM
- QueryServiceUsageListRequestEBM

The following directories contain the industry component files:

- Enterprise business object (EBO) and enterprise business message (EBM) XML schema files:

  COMMS_AIA_HOME/**comms_home/source/soainfra/apps/AIAMetaData/AIAComponents/EnterpriseObjectLibrary/Industry/Communications/EBO/**

- Enterprise business service (EBS) WSDL files:

  COMMS_AIA_HOME/**comms_home/source/soainfra/apps/AIAMetaData/AIAComponents/EnterpriseBusinessServiceLibrary/Industry/Communications/EBO/**

For detailed documentation of individual EBOs and EBMs, click the AIA Reference Doc link on EBO and EBM detail pages in the Oracle Enterprise Repository (OER).

EBOs can be extended, for instance, to add new data elements. These extensions are protected and remain intact after a patch or an upgrade, so long as the extensibility guidelines are followed.

See the discussion of Oracle AIA assets extensibility patterns in *Oracle Fusion Middleware Developer's Guide for Oracle SOA Core Extension* for more information about extending EBOs.

Integration Services

These services are delivered with the Unbilled Usage flow:

- UnbilledUsageSiebelCommsReqABCS with operations QueryUnbilledUsage, QueryEventDetails, SearchEventDetails, and QueryBalanceDetails
- QueryUnbilledUsageSiebelCommsReqABCSImpl
- QueryServiceUsageBRMCommsProvABCSImpl
- QueryUnbilledEventDetailsSiebelCommsReqABCSImpl
- SearchUnbilledEventDetailsSiebelCommsReqABCSImpl
- QueryUnbilledBalanceDetailsSiebelCommsReqABCSImpl

UnbilledUsageSiebelCommsReqABCS

UnbilledUsageSiebelCommsReqABCS exposes the following operations:

- QueryUnbilledUsage:
  - Routes QueryUnbilledUsageReqMsg to the requester implementation service
  - Routes QueryUnbilledUsageRespMsg to the requester
- QueryEventDetails:
  - Routes QueryEventDetailsReqMsg to the requester implementation service
  - Routes QueryEventDetailsRespMsg to the requester
- SearchEventDetails:
  - Routes SearchEventDetailsReqMsg to the requester implementation service
  - Routes SearchEventDetailsRespMsg to the requester
- QueryBalanceDetails:
  - Routes QueryBalanceDetailsReqMsg to the requester implementation service
  - Routes QueryBalanceDetailsRespMsg to the requester

See *Siebel Order Management Guide Addendum for Communications* for more information about these web services.

QueryUnbilledUsageSiebelCommsReqABCSImpl

This service transforms the Siebel message into a QueryServiceUsageListRequest EBM and calls the provider to get the QueryServiceUsage response from the billing system. It then transforms the EBM response back to a Siebel message and returns it to the calling Siebel web service.

QueryServiceUsageBRMCommsProvABCSImpl

QueryServiceUsageBRMCommsProvABCSImpl transforms:

- ServiceUsageEBM into BRM API input format and calls the API to get the service items output from the billing system.

  It then transforms the output from the API back to a ServiceUsage EBM message and returns it to the calling requestor.
- QueryServiceUsageListEBM into BRM API input formats and calls the APIs to query the resource balance output from the billing system.

  It then transforms the output from the APIs back to a ServiceUsage EBM message and returns it to the calling requestor.

QueryUnbilledEventDetailsSiebelCommsReqABCSImpl

QueryUnbilledEventDetailsSiebelCommsReqABCSImpl transforms the QueryEventDetailsReqMsg into QueryServiceUsageListEBM.

SearchUnbilledEventDetailsSiebelCommsReqABCSImpl

SearchUnbilledEventDetailsSiebelCommsReqABCSImpl transforms SearchEventDetailsReqMsg into QueryServiceUsageEBM.

QueryUnbilledBalanceDetailsSiebelCommsReqABCSImpl

QueryUnbilledBalanceDetailsSiebelCommsReqABCSImpl transforms the Siebel message into a QueryServiceUsageListEBM and calls the provider to query the resource balance response from the billing system. It then transforms the EBM response back to a Siebel message and returns it to the calling Siebel web service.

About View and Capture Payments

The View and Capture Payments integration between Siebel CRM and BRM supports the following integration scenarios:

- CreateReceivedPayment lets you capture a payment in Siebel CRM either for an account at the billing profile level or at the invoice level and to post the payment in BRM.
- QueryReceivedPaymentList lets you view the history of payments in Siebel CRM at both the billing profile level and invoice level by retrieving payment records from BRM.
- SearchPayment lets you search for payment records in BRM to display in Siebel CRM for an account at the billing profile level or at the invoice level.

CreateReceivedPayment Integration Flow

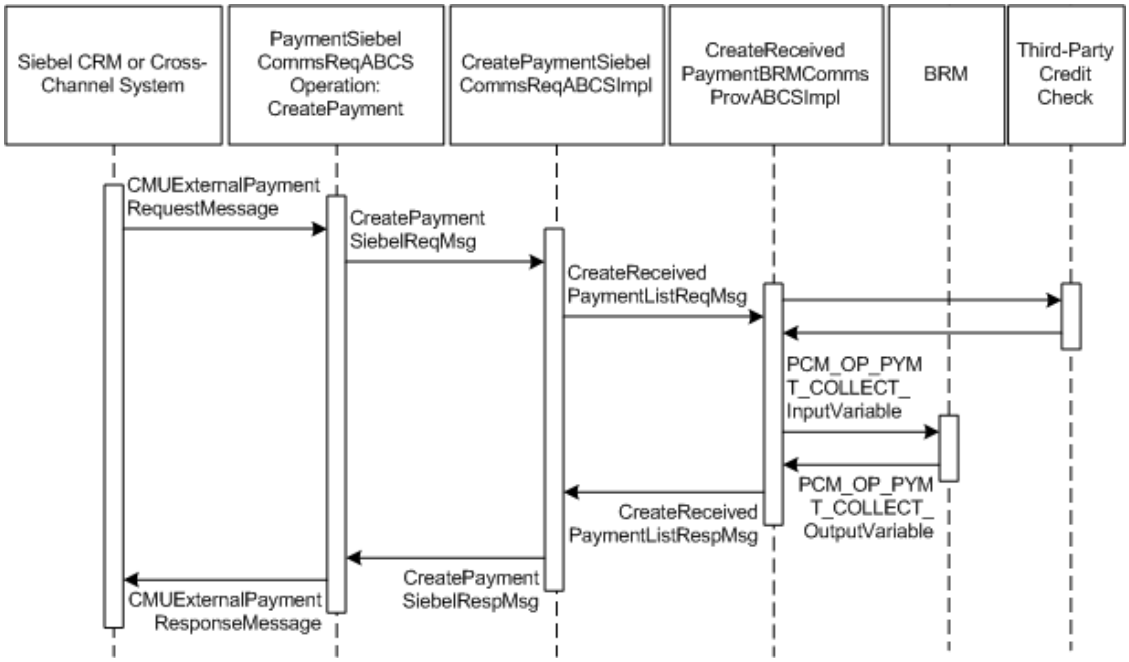
This integration flow uses the following interfaces:

- PaymentSiebelCommsReqABCS with operations CreatePayment and CreateInvoicePayment
- CreatePaymentSiebelCommsReqABCSImpl
- CreateInvoicePaymentSiebelCommsReqABCSImpl
- CreateReceivedPaymentBRMCommsProvABCSImpl

> **Note**
>
> The integration flow for CreateInvoicePayment is similar to the integration flow for CreatePayment.

Figure 2-21 illustrates the CreateReceivedPayment integration flow.

Figure 2-21 CreateReceivedPayment Integration Flow Sequence Diagram



The CreateReceivedPayment integration flow occurs as follows:

1. You initiate the CreateReceivedPayment process by submitting a one-time payment for a billing profile from Siebel CRM or a payment channel. See the discussion of recording profile level payments in *Siebel Communications Guide* for more information.
2. Siebel CRM or the channel invokes the PaymentSiebelCommsReqABCS web service with the operation **CreatePayment**.
3. The integration routes the CreatePaymentSiebelReqMsg message to the CreatePaymentSiebelCommsReqABCSImpl service.

4. The CreatePaymentSiebelCommsReqABCSImpl service transforms the CreatePaymentReqMsg into the CreateReceivedPaymentListReqMsg enterprise business message (EBM) and routes it to the appropriate billing system.

   By default, the CreateReceivedPaymentListReqMsg EBM is sent to the CreateReceivedPaymentBRMCommsProvABCSImpl service to be routed to BRM.

5. If the CreateReceivedPaymentListReqMsg EBM indicates that a third-party credit check is required (the **OneTimePayment** property is set to **T**), the CreateReceivedPaymentBRMCommsProvABCSImpl service initiates the third-party credit check and receives the authorization response.

6. If the third-party credit check is successful or is not required, the CreateReceivedPaymentBRMCommsProvABCSImpl service transforms the CreateReceivedPaymentListReqMsg EBM into the input of PCM_OP_PYMT_COLLECT and calls the BRMPymtServices web service with the PCM_OP_PYMT_COLLECT opcode.

   When calling the PCM_OP_PYMT_COLLECT opcode, the CreateReceivedPaymentBRMCommsProvABCSImpl service sets the value of the **PIN_FLD_COMMAND** field as follows:

   - If the **OneTimePayment** property in the EBM is set to **Y**, the **PIN_FLD_COMMAND** field is set to **4**.
   - If the **OneTimePayment** property in the EBM is set to **O** or **T**, the **PIN_FLD_COMMAND** field is set to **0**.

   If the third-party credit check is not successful, a BPEL exception is thrown and a failure response message is returned to Siebel CRM or the cross-channel system.

7. If the **PIN_FLD_COMMAND** field is set to **4**, the BRM initiates the payment authorization with a third-party system and receives the response.

8. If the BRM authorization is successful or was not required, the BRMPymtServices web service calls the PCM_OP_PYMT_COLLECT opcode and returns the payment object output to the CreateReceivedPaymentBRMCommsProvABCSImpl service.

   If the BRM authorization is not successful, a BPEL exception is thrown and a failure response message is returned to Siebel CRM or the cross-channel system.

9. The CreateReceivedPaymentBRMCommsProvABCSImpl service transforms the BRM API output into a CreateReceivedPaymentListRespMsg EBM and returns it to the CreatePaymentSiebelCommsReqABCSImpl service.

10. The CreatePaymentSiebelCommsReqABCSImpl service transforms the CreatePaymentListRespMsg EBM into a CreatePaymentSiebelRespMsg message, and returns it to the PaymentSiebelCommsReqABCS service.

11. The PaymentSiebelCommsReqABCS service returns the CreatePaymentSiebelRespMsg message to the CMUCreatePayment service as a CMUCreatePaymentResponseMessage message.

12. The system writes the CMUCreatePaymentResponseMessage to Siebel CRM or the payment channel and displays the payment confirmation number.

QueryReceivedPaymentList Integration Flow

The QueryReceivedPaymentList integration flow uses the following interfaces:

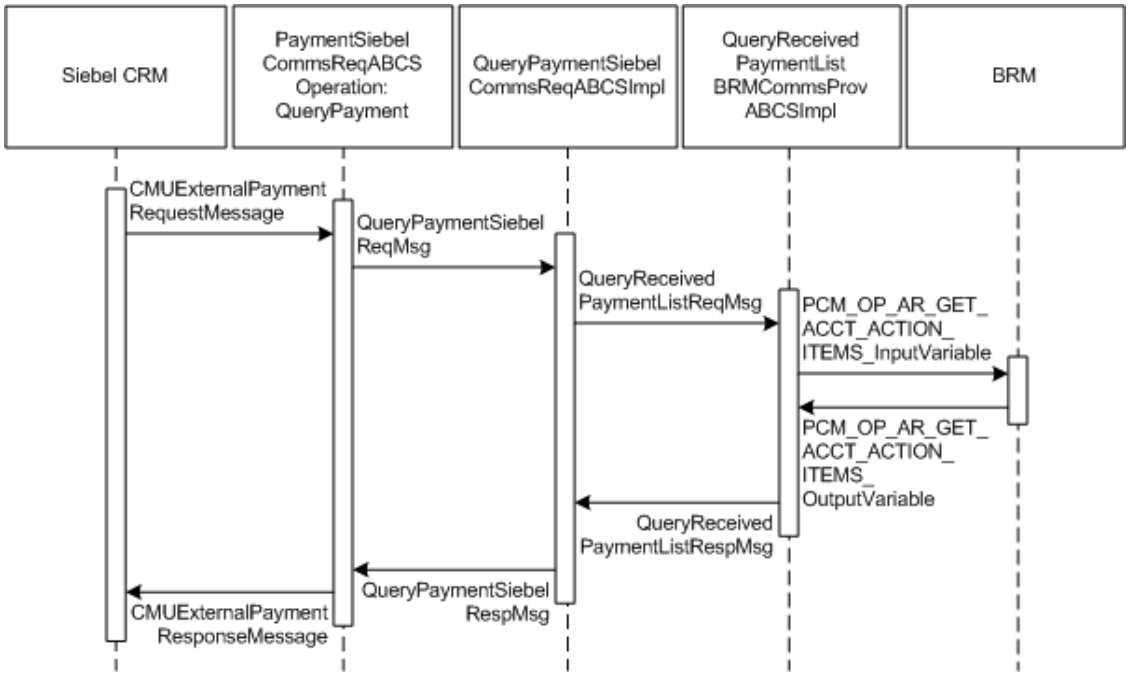
- ReceivedPaymentSiebelCommsReqABCS Interface with operations QueryPayment, SearchPayment, and QueryInvoicePayment
- QueryPaymentSiebelCommsReqABCSImpl

- QueryInvoicePaymentSiebelCommsReqABCSImpl
- SearchPaymentSiebelCommsReqABCSImpl
- QueryReceivedPaymentListBRMCommsProvABCSImpl

> **Note**
>
> The QueryInvoicePayment integration flow is similar to the QueryPayment integration flow except that a different ABCSImpl is used.

Figure 2-22 illustrates the QueryReceivedPaymentList integration scenario.

Figure 2-22 QueryReceivedPaymentList Integration Flow Sequence Diagram



When you initiate the QueryReceivedPaymentList process, the following events occur:

1. In Siebel CRM, navigate to the Billing Profile screen.
2. Navigate to the Accounts screen, query an account, and click a billing profile for the account.

   On the Billing Profile screen, click the **Payments** tab for the billing profile. The Payments screen has two buttons, one to create a payment and one to search for payments. Navigating to the **Payments** tab invokes the PaymentSiebelCommsReqABCS web service, which in turn calls PaymentSiebelCommsReqABCS with operation QueryPayment.
3. Invoking PaymentSiebelCommsReqABCS with operation QueryPayment routes the Siebel QueryPaymentReqMsg to the QueryPaymentSiebelCommsReqABCSImpl.
4. The QueryPaymentSiebelCommsReqABCSImpl transforms the QueryPaymentReqMsg into QueryReceivedPaymentListEBM and routes the QueryReceivedPaymentListEBM to the appropriate billing system.

As delivered, QueryReceivedPaymentListEBM is routed to QueryReceivedPaymentListBRMCommsProvABCSImpl.

5. QueryReceivedPaymentListBRMCommsProvABCSImpl transforms QueryReceivedPaymentListReqMsg into the input of PCM_OP_AR_GET_ACCT_ACTION_ITEMS and calls BRMARService with operation PCM_OP_AR_GET_ACCT_ACTION_ITEMS.
6. Invoking BRMARService with operation PCM_OP_AR_GET_ACCT_ACTION_ITEMS invokes the API PCM_OP_AR_GET_ACCT_ACTION_ITEMS and returns the payment object output to QueryReceivedPaymentListBRMCommsProvABCSImpl.
7. QueryReceivedPaymentListBRMCommsProvABCSImpl then transforms the BRM API output into QueryReceivedPaymentListResponseEBM and returns it to QueryReceivedPaymentListSiebelABCSImpl.
8. QueryReceivedPaymentListSiebelABCSImpl transforms the QueryReceivedPaymentListResponseEBM into QueryPaymentRespMsg, which is returned to PaymentSiebelCommsReqABCS.
9. ReceivedPaymentSiebelCommsReqABCS returns the QueryPaymentRespMsg to CMUQueryPayment as CMUQueryPaymentResponseMessage.
10. The system writes the CMUQueryPaymentResponseMessage to the Siebel Unbilled Details VBC for the subscriber.

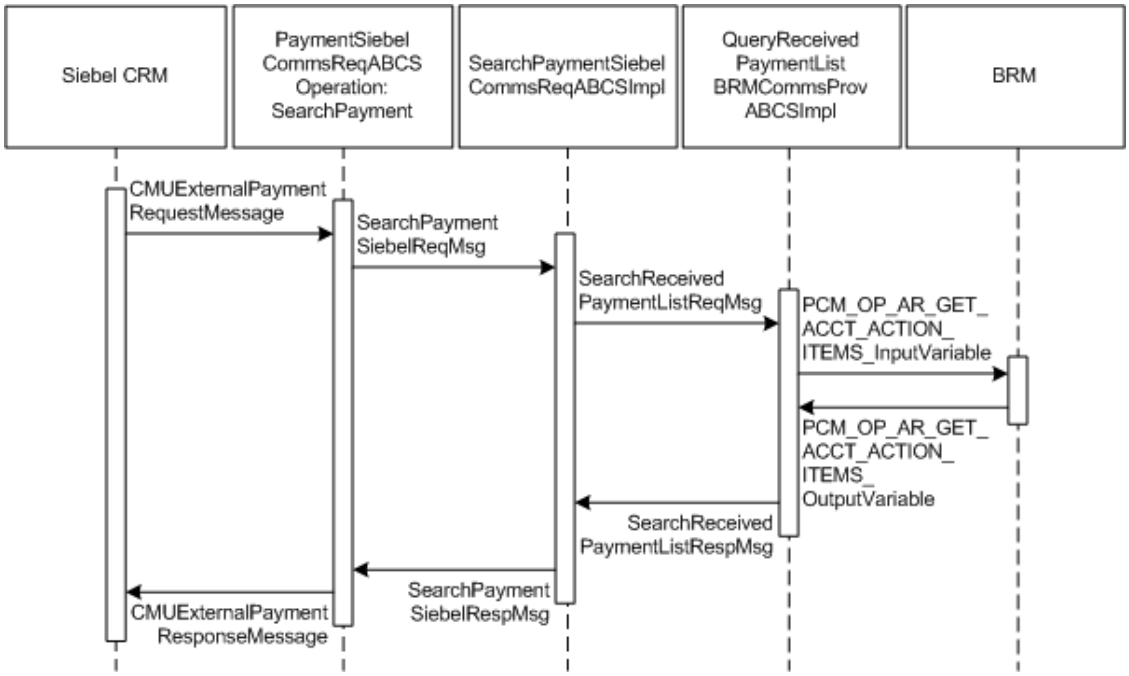
SearchPayment Integration Flow

This integration flow uses the following interfaces:

- ReceivedPaymentSiebelCommsReqABCSInterface with the following operations:
  - QueryPayment
  - SearchPayment
  - QueryInvoicePayment
- SearchPaymentSiebelCommsReqABCSImpl
- QueryReceivedPaymentListBRMCommsProvABCSImpl

Figure 2-23 illustrates the SearchPayment integration scenario.

Figure 2-23 SearchPayment Integration Flow Sequence Diagram



When you initiate the SearchPayment process, the following events occur:

1. In Siebel CRM, the user navigates to the Accounts screen, queries an account, and clicks a billing profile for the account. This displays the Billing Profile BRM screen. Click the **Payments** tab for the billing profile to access the payments screen. The Payments screen has two tabs, one to create a payment and one to search for payments. When you click **Search Payment**, a search applet is opened. After entering the search criteria click Go. This invokes the outbound web service PaymentSiebelCommsReqABCS with operation SearchPayment to fetch the payment records in the billing system.

   The following scenarios exist in which the same functionality is required in Siebel CRM. The integration flow is similar in both of these cases, but they have a separate SiebelABCSImpl.

   Once the CMUGetPayments is invoked, it fetches all records. You can search based on certain search criteria such as dates and amount range. After entering inputs, click Search to initiate this flow.

   From the **Invoice** tab, you can search for payments that are made against a specific invoice.

2. Navigating to the **Payments** tab invokes the PaymentSiebelCommsReqABCS web service, which calls PaymentSiebelCommsReqABCS with the operation SearchPayment.

   PaymentSiebelCommsReqABCS is a generic Siebel Payments interface service with several operations defined on the ReceivedPayment enterprise business object (EBO).

3. Invoking the PaymentSiebelCommsReqABCS with the operation SearchPayment routes the Siebel SearchPaymentReqMsg to SearchPaymentSiebelCommsReqABCSImpl.

4. The SearchPaymentSiebelCommsReqABCSImpl first transforms the SearchPaymentReqMsg into the SearchReceivedPaymentEBM and routes the SearchReceivedPaymentEBM to the appropriate billing system.

As delivered, the SearchReceivedPaymentEBM is routed to the QueryReceivedPaymentListBRMCommsProvABCSImpl.

5. The QueryReceivedPaymentListBRMCommsProvABCSImpl first transforms the SearchReceivedPaymentListReqMsg into the input of PCM_OP_AR_GET_ACCT_ACTION_ITEMS and calls the BRM web service BRMARServices with the operation PCM_OP_AR_GET_ACCT_ACTION_ITEMS.
6. Invoking the BRMARServices with the operation PCM_OP_AR_GET_ACCT_ACTION_ITEMS invokes the BRM API PCM_OP_AR_GET_ACCT_ACTION_ITEMS and returns the payment object output to QueryReceivedPaymentListBRMCommsProvABCSImpl.
7. The QueryReceivedPaymentListBRMCommsProvABCSImpl then transforms the BRM API output into the SearchReceivedPaymentResponseEBM and returns it to SearchReceivedPaymentListSiebelABCSImpl.
8. The SearchReceivedPaymentListSiebelABCSImpl transforms the SearchReceivedPaymentResponseEBM into the SearchPaymentRespMsg, which is returned to the PaymentSiebelCommsReqABCS.
9. The ReceivedPaymentSiebelCommsReqABCS returns the SearchPaymentRespMsg to the calling Siebel web service CMUSearchPayment as CMUSearchPaymentResponseMessage.
10. The system writes the CMUSearchPaymentResponseMessage to the Siebel Unbilled Details VBC for the subscriber.

BRM Interfaces

The CreateReceivedPayment integration flow uses this opcode:

- PCM_OP_PYMT_COLLECT

The QueryReceivedPaymentList integration flow uses this opcode:

- PCM_OP_AR_GET_ACCT_ACTION_ITEMS

See *Oracle Communications Billing and Revenue Management Opcode Flist Reference* for more information.

Siebel CRM Interfaces

The View and Capture Payments flow uses these Siebel CRM interfaces:

- For the CreateReceivedPayment flow: PaymentSiebelCommsReqABCS operation CreatePayment
- For the QueryReceivedPaymentList flow: PaymentSiebelCommsReqABCS operation QueryPayment

See *Siebel Order Management Guide Addendum for Communications* for more information about these web services.

Industry Oracle AIA Components

The View and Capture Payment flow uses the following delivered EBOs and EBMs:

- ReceivedPaymentEBO
- CreateReceivedPaymentEBM
- CreateReceivedPaymentResponseEBM

- QueryReceivedPaymentListEBM
- QueryReceivedPaymentListResponseEBM

The following directories contain the industry component files:

- Enterprise business object (EBO) and enterprise business message (EBM) XML schema files:

  COMMS_AIA_HOME/**comms_home/source/soainfra/apps/AIAMetaData/AIAComponents/EnterpriseObjectLibrary/Industry/Communications/EBO/**

- Enterprise business service (EBS) WSDL files:

  COMMS_AIA_HOME/**comms_home/source/soainfra/apps/AIAMetaData/AIAComponents/EnterpriseBusinessServiceLibrary/Industry/Communications/EBO/**

For detailed documentation of individual EBOs and EBMs, click the AIA Reference Doc link on EBO and EBM detail pages in the Oracle Enterprise Repository (OER).

EBOs can be extended, for instance, to add new data elements. These extensions are protected and remain intact after a patch or an upgrade, so long as the extensibility guidelines are followed.

See the discussion of Oracle AIA assets extensibility patterns in *Oracle Fusion Middleware Developer's Guide for Oracle SOA Core Extension* for more information about extending EBOs.

Integration Services

These services are delivered with the View and Capture Payments flow:

- PaymentSiebelCommsReqABCS
- CreatePaymentSiebelCommsReqABCSImpl
- CreateInvoicePaymentSiebelCommsReqABCSImpl
- CreateReceivedPaymentBRMCommsProvABCSImpl
- QueryPaymentSiebelCommsReqABCSImpl
- QueryInvoicePaymentSiebelCommsReqABCSImpl
- SearchPaymentSiebelCommsReqABCSImpl
- QueryReceivedPaymentListBRMCommsProvABCSImpl

PaymentSiebelCommsReqABCS

PaymentSiebelCommsReqABCS exposes the following operations:

- CreatePayment:
  - Routes CreatePaymentReqMsg to the requester implementation service
  - Routes CreatePaymentSiebelRespMsg to the requester
- CreateInvoicePayment:
  - Routes CreateInvoicePaymentReqMsg to the requester implementation service
  - Routes CreateInvoicePaymentRespMsg to the requester
- QueryPayment:
  - Routes QueryPaymentReqMsg to the requester implementation service

  - Routes QueryPaymentRespMsg to the requester
- SearchPayment:
  - Routes SearchPaymentReqMsg to the requester implementation service
  - Routes SearchPaymentRespMsg to the requester
- QueryInvoicePayment:
  - Routes QueryInvoicePaymentReqMsg to the requester implementation service.
  - Routes QueryInvoicePaymentRespMsg to the requester

CreatePaymentSiebelCommsReqABCSImpl

The CreatePaymentSiebelCommsReqABCSImpl transforms the CreatePaymentReqMsg into a Payment EBM.

CreateInvoicePaymentSiebelCommsReqABCSImpl

CreateInvoicePaymentSiebelCommsReqABCSImpl transforms the CreateReceivedPaymentResponseEBM into CreateInvoicePaymentSiebelRespMsg, which is returned to the Siebel Account ABC interface service.

CreateReceivedPaymentBRMCommsProvABCSImpl

CreateReceivedPaymentBRMCommsProvABCSImpl transforms the ReceivedPayment EBM into BRM API input format and calls the API to Create Payment output from the billing system. It then transforms the output from the API back to a ReceivedPayment EBM message and returns it to the calling requestor.

QueryPaymentSiebelCommsReqABCSImpl

The QueryPaymentSiebelCommsReqABCSImpl transforms the QueryPaymentReqMsg into QueryReceivedPaymentListEBM.

QueryInvoicePaymentSiebelCommsReqABCSImpl

The QueryInvoicePaymentSiebelCommsReqABCSImpl transforms the QueryInvoicePaymentReqMsg into ReceivedPaymentEBM.

SearchPaymentSiebelCommsReqABCSImpl

The SearchPaymentSiebelCommsReqABCSImpl transforms the SearchPaymentReqMsg into ReceivedPaymentEBM.

QueryReceivedPaymentListBRMCommsProvABCSImpl

QueryReceivedPaymentListBRMCommsProvABCSImpl transforms the BRM API output into QueryReceivedPaymentListResponseEBM.

Configuring Request-to-Answer Business Process

This section describes how to configure the request-to-answer business process.

About Configuring Billing Management

This section discusses how to set up Oracle Communications Billing and Revenue Management (BRM) and Siebel customer relationship management (Siebel CRM) for integrated billing management.

In addition, it discusses how to work with domain value maps (DVMs) and cross-references, how to handle errors, and how to configure the billing management process flow.

Setting Up BRM for Integrated Billing Management

To set up BRM for integrated billing management:

- Configure the BRM JCA adapter. See Deploying and Configuring JCA Resource Adapter on Oracle WebLogic Server in *Oracle Communications Billing and Revenue Management JCA Resource Adapter* for more information.
- To ensure that infinitely effective resource balances show a null date (instead of 31-Dec-1969/01-Jan-1970), set the ZeroEpochAsNull BRM JCA parameter (in JCA Resource Adapter connection factory) to **True**. See Configuring How to Represent Infinite Date Values in *Oracle Communications Billing and Revenue Management JCA Resource Adapter*.
- To display billing dates in Siebel CRM in the same time zone as the billing system server time, set the InteractionTimeZone parameter to the time zone of the BRM server. The InteractionTimeZone parameter in the JCA Adapter controls the time zone conversion for dates that are returned by BRM for billing queries. See Connecting JCA Resource Adapter to BRM from Oracle WebLogic Server in *Oracle Communications Billing and Revenue Management JCA Resource Adapter*.

Setting Up Siebel CRM for Integrated Billing Management

To set up Siebel CRM for integrated billing management:

- Set the UTCCanonical process property to *Y* for the Siebel CRM interfaces described in the instructions for ACR 474 and ACR 508 in *Siebel Maintenance Release Guide*.
- Configure the SWICreateAdjustment Siebel CRM outbound workflow to enqueue the Siebel messages in the AIA_CMUREQADJIOJMSQUEUE queue for the CreateAdjustment flow.

  See *Oracle Fusion Middleware Developer's Guide for Oracle SOA Core Extension* and *Siebel Application Integration for Oracle Fusion Middleware Guide* for more information about configuring queueing in Siebel CRM and Oracle AIA.

Working with DVMs

Domain value maps (DVMs) are a standard feature of the Oracle service-oriented architecture (SOA) Suite that enable you to equate lookup codes and other static values across applications, for example, FOOT and FT or US and USA.

DVMs are static in nature, though administrators can add maps as required. Transactional business processes never update DVMs-they only read from them. They are stored in XML files and cached in memory at run time.

DVM types are seeded for the Oracle Communications Billing and Revenue Management: Cash to Care flows. Administrators can extend the list of mapped values by adding more maps.

Table 2-2 lists the DVMs for the billing management flow.

Table 2-2 Billing Management Integration - DVMs

DVM	Description
CURRENCY_CODE	Currency codes.
RESOURCE	Nonmonetary resources (Free Minutes, Text Messages, and so on).
ACCOUNTBALANCEADJUSTMENT_REASON	Reason for adjustment.
ACCOUNTBALANCEADJUSTMENT_STATUS	Status of adjustment request (Posted, Not-Posted).
ACCOUNTBALANCEADJUSTMENT_TYPE	Type of adjustment (Credit, debit, and so on).
ACCOUNTBALANCEADJUSTMENT_TAXTREATMENT	Tax treatment on adjustment amount (Include, Exclude).
ACCOUNTBALANCEADJUSTMENT_USAGEALLOCATION_TAXTREATMENT	Tax treatment on CDR adjustment amount (Include, Exclude).
INSTALLEDPRODUCT_STATUS	Status of installed product (Active, Canceled, and so on).
RECIEVEDPAYMENT_TYPE	Type of payment (Credit, Direct Debit).
ACCOUNTBALANCEADJUSTMENT_SUBSTATUS	Sub-status of adjustment request.

See *Oracle Fusion Middleware Developer's Guide for Oracle SOA Core Extension* for more information about working with DVMS.

Working with Cross-References

Cross-references map and connect the records within the application network, and they enable these applications to communicate in the same language. The integration server stores the relationship in a persistent way so that others can refer to it.

Table 2-3 contains the billing management integration cross-references.

Table 2-3 Billing Management Integration - Cross-References

Name	Columns	Mapping Details	Description
CUSTOMERPARTY_ACCOUNTID	SEBL_01,COMMON,BRM_01	Set up as part of customer sync	Query
CUSTOMERPARTY_BILLPROFILEID	SEBL_01,COMMON,BRM_01	Set up as part of customer sync	Query

Handling Errors

Based on the roles defined for the services, e-mail notifications are sent if a service ends due to an error.

Table 2-4 lists the error messages provided by collections management for billing management.

Table 2-4 Billing Management Integration - Error Messages

Integration/Service Name	Error Code	Message Text
Account Balance / QueryBalanceSummarySiebel ReqABCSImpl Query Invoice List / QueryInvoiceListSiebelCommsReqAB CSImpl	AIA_ERR_AIACOMBMPI_0003	Billing Profile BPName for the account does not exist in the billing system. 1) To correct the error, submit a sales order with this billing profile. 2) Ensure that the sales order created with this billing profile is successfully submitted to the billing system.
Create Payment / CreateReceivedPaymentBRMComms ProvABCSImpl	AIA_ERR_AIACOMBMPI_0005	BRM Error Message (For example, *Service Unavailable*)

Describing Delivered Error Notification Roles and Users

The following roles and users are delivered as default values for issuing error notifications for the billing management flow:

- **Role:** AIAIntegrationAdmin
- **User:** AIAIntegrationAdminUser

See *Oracle Fusion Middleware Infrastructure Components and Utilities User's Guide for Oracle Application Integration Architecture Foundation Pack* for information about how to set up error notifications and trace and error logs using these values.

Configuring Billing Management

Configure these properties located in the *COMMS_AIA_HOME***/comms_home/source/soainfra/apps/config/AIAConfigurationProperties.xml** file. Entries in the **AIAConfigurationProperties.xml** file are case-sensitive.

See the discussion of building AIA integration flows in *Oracle Fusion Middleware Developer's Guide for Oracle SOA Core Extension* for more information about reloading updates to **AIAConfigurationProperties.xml** file.

The following Business Process Execution Language (BPEL) processes have entries listed in Table 2-5.

- QueryBalanceSummarySiebelCommsReqABCSImpl
- QueryCustomerPartyListBRMCommsProvABCSImpl
- QueryBalanceGroupListSiebelCommsReqABCSImpl
- QueryBalanceDetailsSiebelCommsReqABCSImpl
- QueryBalanceGroupServicesSiebelCommsReqABCSImpl
- QueryInstalledProductListBRMCommsProvABCSImpl
- QueryInvoiceListSiebelCommsReqABCSImpl
- QueryInvoiceListBRMCommsProvABCSImpl
- QueryInvoiceSiebelCommsReqABCSImpl
- QueryInvoiceEventDetailsSiebelCommsReqABCSImpl
- SearchInvoiceEventDetailsSiebelCommsReqABCSImpl

- QueryInvoiceBalanceDetailsSiebelCommsReqABCSImpl
- QueryUnbilledUsageSiebelCommsReqABCSImpl
- QueryServiceUsageBRMCommsProvABCSImpl
- QueryUnbilledEventDetailsSiebelCommsReqABCSImpl
- SearchUnbilledEventDetailsSiebelCommsReqABCSImpl
- QueryUnbilledBalanceDetailsSiebelCommsReqABCSImpl
- CreatePaymentSiebelCommsReqABCSImpl
- CreateInvoicePaymentSiebelCommsReqABCSImpl
- CreateReceivedPaymentBRMCommsProvABCSImpl
- QueryPaymentSiebelCommsReqABCSImpl
- QueryInvoicePaymentSiebelCommsReqABCSImpl
- SearchPaymentSiebelCommsReqABCSImpl
- QueryReceivedPaymentListBRMCommsProvABCSImpl
- QueryAccountBalanceAdjustmentSiebelCommsReqABCSImpl
- QueryAccountBalanceAdjustmentBRMCommsProvABCSImpl
- CreateAccountBalanceAdjustmentBRMCommsProvABCSImpl

Table 2-5 BPEL Process Property Values - 1

Property Name	Value/Default Value	Description
ABCSExtension.PreXform*ABM/EBM_name*TO*ABM/EBM_name*	true/false	Controls whether the extension point before transformation of application business message (ABM) to enterprise business message (EBM) is invoked during processing.
ABCSExtension.PreInvoke*PartnerLink_name*	true/false	Controls whether the extension point before invocation to enterprise business service (EBS) is invoked during processing.
ABCSExtension.PostXform*ABM/EBM_name*to*ABM/EBM_name*	true/false	Controls whether the extension point before transformation of EBM to ABM is invoked during processing.
ABCSExtension.PostInvokePartnerLink_name	true/false	Controls whether the extension point before invocation of callback service or response return is invoked during processing.
Routing.PartnerLink_Name.RouteToCAVS	true/false	Controls whether the Composite Application Validation System (CAVS) is used to handle the request.
Default.SystemID	Valid string	Specifies the name of the default systemID of the requester application.
Routing.PartnerLink_name.BRM_01.EnpointURI	eis/BRM	Specifies the JNDI entry for the partner link.
EBSOverride.*EBS_name*.*operation_name*.PortType	Valid string	PortType of the webservice that needs to be invoked dynamically. This value should be consistent with the EBSOverride.*EBS_name*.*operation_name*.Address property.

Table 2-5 (Cont.) BPEL Process Property Values - 1

Property Name	Value/Default Value	Description
EBSOverride.*EBS_name*.*operation_name*.ServiceName	Valid string	ServiceName of the webservice that needs to be invoked dynamically. This value should be consistent with the EBSOverride.*EBS_name*.*operation_name*.Address property.
EBSOverride.*EBS_name*.*operation_name*.Address	Valid string	This property is used to dynamically invoke any webservice from this service. This holds the address.endpoint URI of the webservice that needs to be invoked dynamically. To invoke CAVS or any other provider ABCS, this property needs to be updated accordingly.
BRM.Payment.Command	0	This property is specific to CreateReceivedPaymentBRMCommsProvABCSImpl.
Routing.CreateAccountBalanceAdjustmentListResponseBRMCommsJMSProducer.EndpointURI	Valid string	Endpoint URL of the CreateAccountBalanceAdjustmentListResponseBRMCommsJMSProducer. This property is specific to CreateAccountBalanceAdjustmentBRMCommsProvABCSImpl.

These BPEL processes have entries listed in Table 2-6.

- CreateAccountBalanceAdjustmentSiebelCommsReqABCSImpl
- UpdateAccountBalanceAdjustmentRespSiebelCommsProvABCSImpl

Table 2-6 BPEL Processes Property Values - 2

Property Name	Value/Default Value	Description
ABCSExtension.PreXform*ABM/EBM_name*TO*ABM/EBM_name*	true/false	Controls whether the extension point before transformation of ABM to EBM is invoked during processing.
ABCSExtension.PreInvoke*PartnerLink_name*	true/false	Controls whether the extension point before invocation to enterprise business service (EBS) is invoked during processing.
Routing.*PartnerLink_name*.RouteToCAVS	true/false	Controls whether the CAVS is used to handle the request.
Default.SystemID	Valid string	Specifies the name of the default systemID of the requester application.
EBSOverride.*EBS_name*.*operation_name*.PortType	Valid string	PortType of the webservice that needs to be invoked dynamically. This value should be consistent with the EBSOverride.*EBS_name*.*operation_name*.Address property.
EBSOverride.*EBS_name*.*operation_name*.ServiceName	Valid string	ServiceName of the webservice that needs to be invoked dynamically. This value should be consistent with the EBSOverride.*EBS_name*.*operation_name*.Address property.

Table 2-6 (Cont.) BPEL Processes Property Values - 2

Property Name	Value/Default Value	Description
EBSOverride.*EBS_name*.*operation_name*.Address	Valid string	This property is used to dynamically invoke any webservice from this service. This holds the address.endpoint URI of the webservice that needs to be invoked dynamically. To invoke CAVS or any other provider ABCS, this property needs to be updated accordingly.
Routing.SWIAdjustmentStatusUpdate.SEBL_01.EndpointURI	Valid string	Siebel endpoint URIL. This property is specific to UpdateAccountBalanceAdjustmentRespSiebelCommsProvABCSImpl.

Configuring Oracle HTTP Server for Billing Management

To integrate invoice header-level adjustments, you must configure Oracle HTTP Server (OHS) so that it recognizes the Oracle AIA mirror servlet that comes with the installation.

For more information about servlets, see *Oracle Fusion Middleware Developing Web Applications, Servlets, and JSPs for Oracle WebLogic Server*.

To configure OHS so that it recognizes the mirror servlet:

1. Open the following file:

   ```
   WebTier_HOME/instances/instance_name/config/OHS/component_name/mod_wl_ohs.conf
   ```

   where:

   - *WebTier_HOME* is the directory in which OHS Web Tier is installed.
   - *instance_name* is the OHS Web Tier instance defined during cluster set up.
   - *component_name* is the name for the OHS component defined during cluster set up. By default, this is **ohs1**.

2. Add the following code:

   ```
   <Location  /AIA>
       SetHandler weblogic-handler
       WebLogicCluster hostname1:port,hostname2:port
       WLLogFile  /tmp/web_log.log
   </Location>
   ```

   where:

   - *hostname1* and *hostname2* are the host names of the servers in the Oracle WebLogic server cluster.
   - *port* is the port where the host is listening for HTTP requests.

3. Save and close the file.
4. Restart OHS. See *Oracle Fusion Middleware Administrator's Guide for Oracle HTTP Server* for more information.

3
Complaint-to-Solution Business Process

This chapter describes the Complaint-to-Solution business process and its features.

Overview of the Complaint-to-Solution Business Process

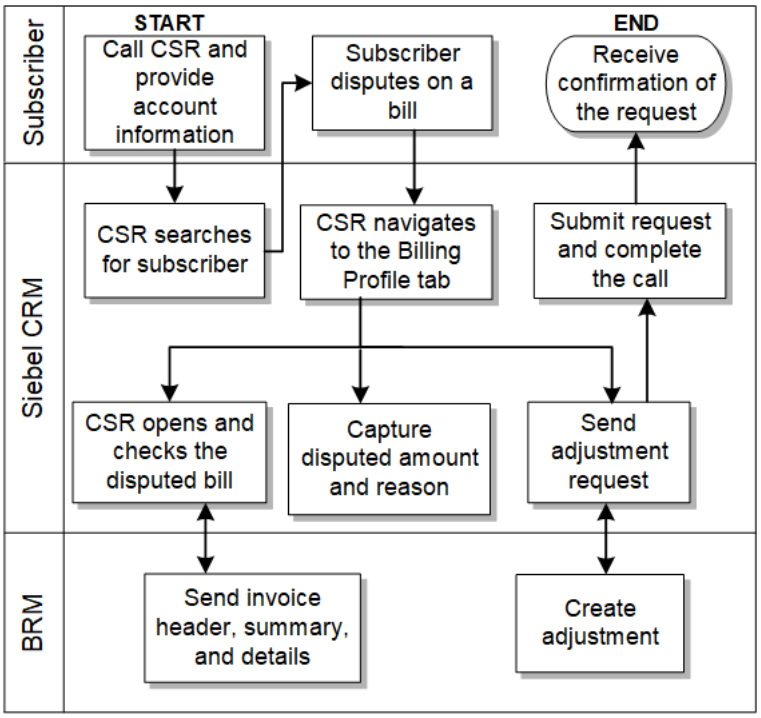
The Complaint-to-Solution business process comprises activities that deal with a complaint (problem) initiated by the subscriber, then the Customer Service Representative (CSR) analyzes it to identify the source of the issue, initiates resolution, monitors progress, and resolves the issue.

It supports the following features:

- View and capture adjustments
  - Credit/refund for bill adjustments
- Collections management
  - Run the collection process in BRM
  - Include the billing profile in collection
  - Sync credit alerts into Siebel
  - Update the credit alert
  - Manually run ODI for collection

Figure 3-1 illustrates a typical request to answer business process flow.

Figure 3-1 Typical Complaint to Solution Business Process Flow



The sections that follow describe the above mentioned features.

About View and Capture Adjustments

The view and capture adjustments feature enables a Customer Service Representative (CSR) to make adjustments at three levels in the invoice (header, item, and event) in Siebel CRM. When a subscriber calls to dispute an item or multiple items on a bill, CSRs first identify the bill, and then find the disputed items on the bill.

The integration supports adjustment requests that are both amount-based and percentage-based for adjustments at the bill/header and event level. Therefore, a CSR either captures an adjustment request for an absolute value (such as USD 2) or percentage value (such as 2%).

In addition to invoice adjustments, a CSR can use this integration feature to make unbilled service usage adjustments at the event level, also known as call detail record (CDR) level.

> **Note**
>
> In the billing system, taxes can only be applied to AR items after billing. Therefore, for Unbilled Event adjustments, the tax flag that is set in the Siebel CRM UI must be Exclude Tax.

CSRs request adjustments on a variety of levels, as appropriate to the situation. For instance, if a subscriber made a 10-minute call that was mistakenly billed as a 30-minute call, the CSR requests an adjustment for that specific call at the call-detail level. If, however, the subscribers plan provided 100 free minutes a month, but charges started accruing after only 30 minutes, the CSR requests an adjustment at the summary-level instead. CSRs can create an

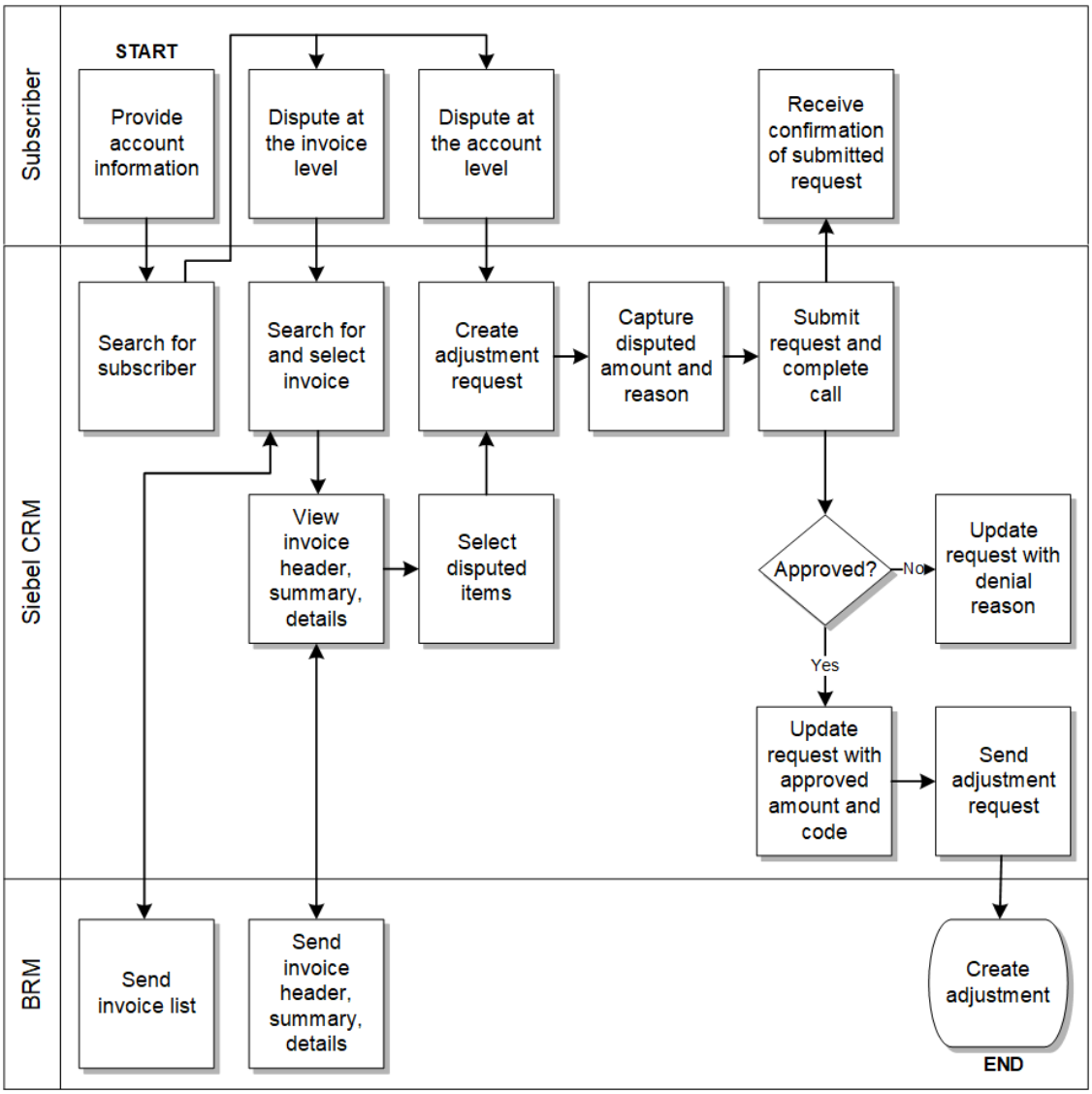
adjustment request for one or more lines on a single invoice. The adjustment request can include the following details per line:

- Account #
- Invoice #
- Request ID
- Requested Date
- Adjustment Amount Requested
- Adjustment Type (for example, credit, debit)
- Reason for Request
- Comments
- Status
- Amount Approved
- Approval Code
- Date Approved

When the CSR clicks the **Adjustments** tab of the Siebel Billing Profile screen, the adjustment history information appears. The adjustment records in this view are adjustment requests that originated from Siebel CRM and are stored in the Siebel database. The CSR must navigate to the **Invoice Detail** view to make adjustments at the header and item levels. For adjustments at the event level, the CSR navigates to the event details view. Each adjustment request triggers a separate web services call in the billing system. The adjustment, if approved, is created in BRM and is reflected in the subscribers' next bill. If the adjustment is not approved, the adjustment request is updated with the reason for denial.

Figure 3-2 illustrates the flow for View and Capture Adjustments.

Figure 3-2 The View and Capture Adjustments Flow



To view the adjustments for an invoice and create different levels of adjustments in Siebel CRM:

1. Navigate to the Accounts screen and query an account. Scroll down to the **Billing Profile** applet and click the **Billing Profile Name** link.
2. Click the **Bills** tab to view the list of bills under the account. Click the **Bill Number** link to open the Bill Details screen.
3. Click the **A/R Items** tab to view the adjustments for an invoice.
4. In the **Bills Detail** view, an adjustment request can be captured at the header, item, or event level. Each of these sections in the view has an **Adjust** button.
5. After creating a new adjustment for a particular level and getting the approval, click the **Submit** button.

See Mapping Siebel Billing Management UI Elements to BRM Customer Center for more information on mapping Siebel CRM elements to BRM elements.

See About View and Capture Adjustments for more information about implementing the view and capture adjustments feature.

About Collections Management

Collections management is the process of collecting payments from subscribers after the grace period for paying dues has ended. If subscribers do not make a payment after the grace period, service providers may choose to remind the subscribers at first with a letter or a phone call. If these contact methods fail, service providers may decide to take actions such as inactivating the service.

Collections management synchronizes collections actions based on specified collection scenarios between BRM and Siebel CRM, and administration of these collection actions and credit alerts.

Collections management covers the entire collections life cycle across BRM and Siebel CRM to define activities typically performed, such as:

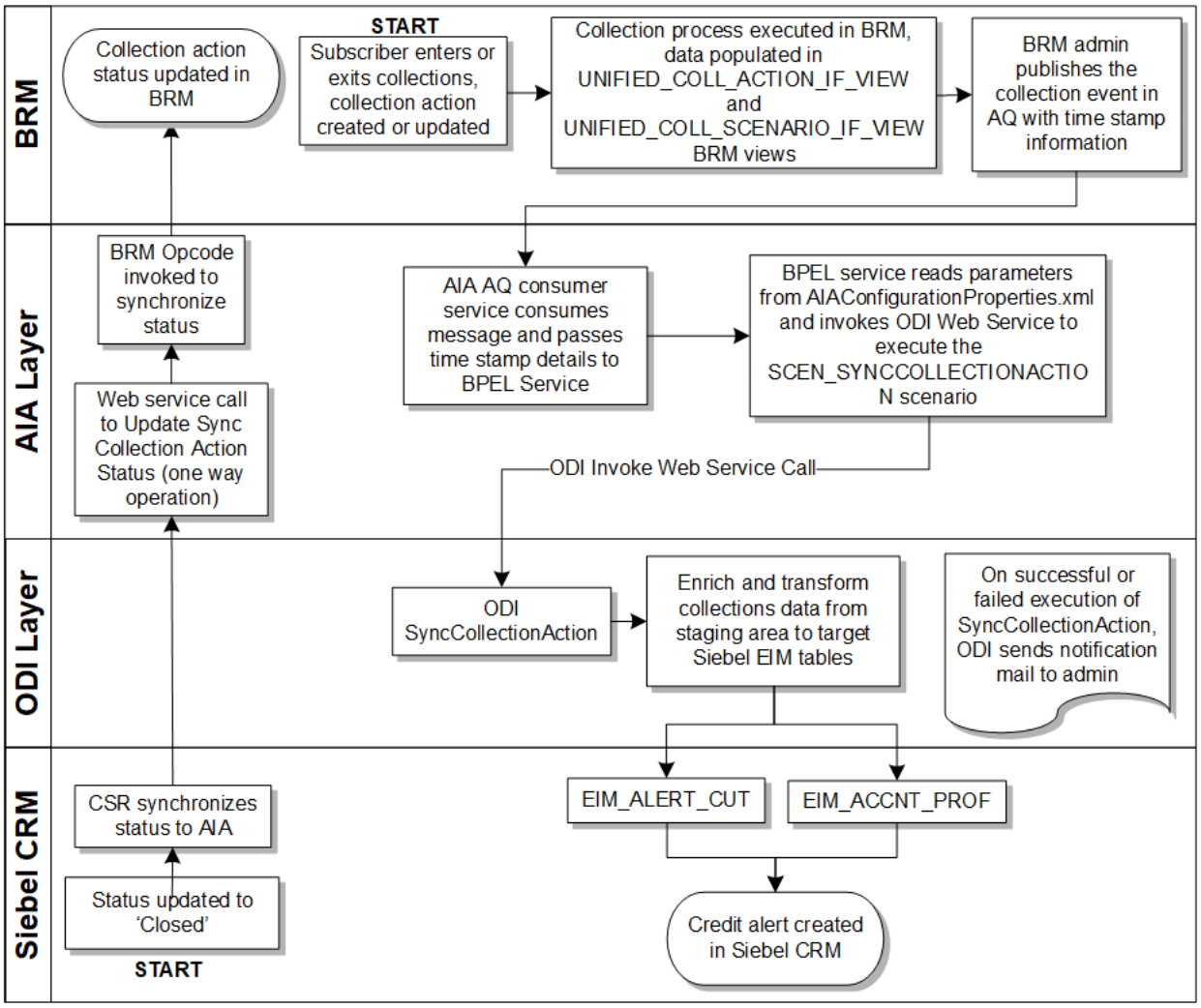
1. In BRM, a collections scenario is defined, created, and associated with a sequence of actions that must be performed.
2. Any bill units that move into collections and require an actionable event are passed to Siebel CRM as an action notification event in the form of a credit alert.
3. Siebel CRM capabilities can be accessed by the collections agent to create additional activities which can be associated with a credit alert.

See About Collections Management for more information about implementing collection management.

About the Collection Management Flow

Collection Management Flow illustrates the overall collection management process flow.

Figure 3-3 Collection Management Flow



Assumptions and Constraints for Working with Collection Management

These are the assumptions and constraints for working with collection management:

- Cross-reference data and domain value map (DVM) setup should be verified for a successful collections batch load run.
- To achieve and maintain high performance, the database memory area must be large enough to hold the frequently accessed data in the cache.
- If an amount due alert has not been sent by BRM a default 0 (zero) amount is set in the integration process.
- When creating a credit alert, a default agreement association is provided at the account level and not the credit alert level.
- BRM collection messages are stored in BRM tables, which can be queried from BRM database views. In a multischema scenario, a unified view is available on BRM to query the information of collections actions and scenarios.
- No cross-reference for the collections action ID exists. The BRM Portal Object (POID) for the collections action ID is sent to Siebel CRM as the integration ID. This POID is used for

all references to the action and when an Update Collection Action and Update Collection Action status is selected.

- In case of failure executions, there are two options available for the administrator:
  - Run the ODI scenario manually.
  - Start the SyncCollectionHeaderInfoBRMCommsReqImpl service using the timestamp details. This will start the ODI web service to begin the scenario.
- ODI uses SMTP port 25 to send notification emails. To use a different port you must configure the SMTP Port in ODI. See the discussion of defining Java options in ODI on the Oracle Support Web site:

  https://support.oracle.com/epmos/faces/DocumentDisplay?id=1317507.1

Implementing the Complaint-to-Solution Business Process

This section explains how the Oracle Application Integration Architecture (Oracle AIA) Oracle Communications Cash to Care business process implements the Complaint-to-Solution business process.

About View and Capture Adjustments

The View and Capture Adjustments integration between Siebel CRM and BRM supports the following integration scenarios:

- QueryAccountBalanceAdjustment Integration Flow enables a Customer Service Representative (CSR) to view the adjustments for an invoice in Siebel CRM.
- CreateAccountBalanceAdjustment Integration Flow enables a CSR to create different levels of adjustments (invoice, item, and event) for an invoice in Siebel CRM.

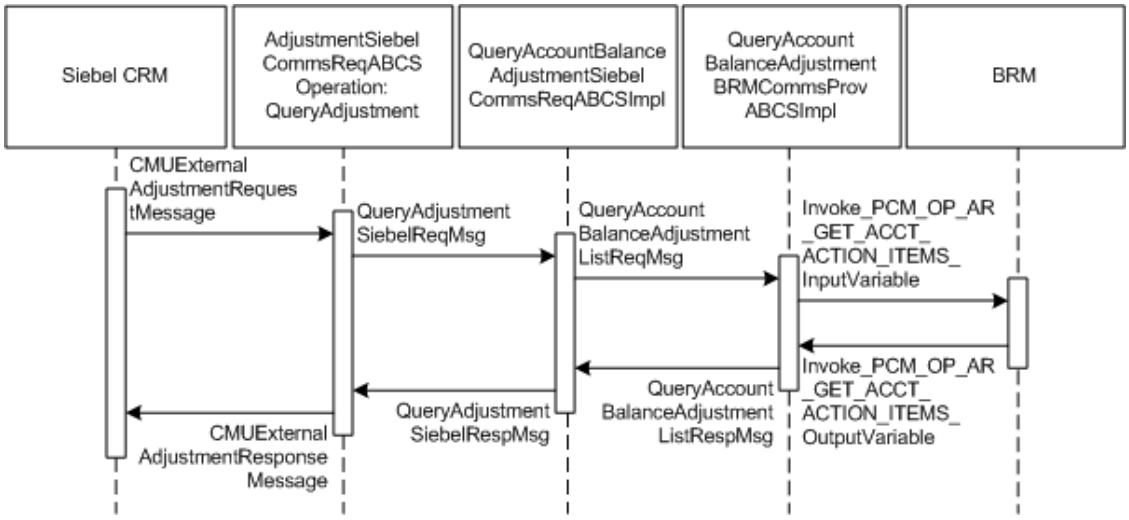
QueryAccountBalanceAdjustment Integration Flow

This integration flow uses the following interfaces:

- AdjustmentSiebelCommsReqABCS with operation QueryAdjustment
- QueryAccountBalanceAdjustmentSiebelCommsReqABCSImpl
- QueryAccountBalanceAdjustmentBRMCommsProvABCSImpl

Figure 3-4 illustrates the QueryAccountBalanceAdjustment integration scenario.

Figure 3-4 QueryAccountBalanceAdjustment Integration Flow Sequence Diagram



When you initiate the QueryAccountBalanceAdjustment process, the following events occur:

1. In Siebel CRM, navigate to the Billing Profile screen.
2. Navigate to Accounts, query an account, and click the billing profile for the account. On the Billing Profile screen, click the **Bill** tab to view the list of bills under the account. To open the Bill Detail View screen, click the **Bill Number** link. This opens the **Bill Details** view with the following information: bill summary, service charges, and item charges. Click the **A/R Items** tab and a web service call is made to get the adjustments specific to this bill for the account.
3. Open an invoice and select the **A/R Items** tab. A web service call is made to get the adjustment for that invoice.
4. Navigate to the **Bill** tab and open the **Bill Details** view.
5. Select the **A/R Items** tab. This invokes the AdjustmentSiebelCommsReqABCS web service, which in turn calls AdjustmentSiebelCommsReqABCS with operation QueryAdjustment.

   AdjustmentSiebelCommsReqABCS is a generic Siebel adjustment interface service with several operations defined on the AccountBalanceAdjustmentEBO.
6. Invoking AdjustmentSiebelCommsReqABCS with operation QueryAdjustment routes the QueryAdjustmentReqMsg to QueryAccountBalanceAdjustmentSiebelCommsReqABCSImpl.
7. The QueryAccountBalanceAdjustmentSiebelCommsReqABCSImpl transforms the QueryAdjustmentReqMsg into QueryAccountBalanceAdjustmentList_InputVariable and routes the QueryAccountBalanceAdjustmentListReqMsg to the appropriate billing system.

   As delivered, QueryAccountBalanceAdjustmentListReqMsg is routed to QueryAccountBalanceAdjustmentBRMCommsProvABCSImpl.
8. QueryAccountBalanceAdjustmentBRMCommsProvABCSImpl first transforms QueryAccountBalanceAdjustmentListReqMsg into the Invoke_PCM_OP_AR_GET_ACCT_ACTION_ITEMS_InputVariable as input of PCM_OP_AR_GET_ACCT_ACTION_ITEMS and calls BRMARServices with operation PCM_OP_AR_GET_ACCT_ACTION_ITEMS.

9. Invoking BRMARServices with operation PCM_OP_AR_GET_ACCT_ACTION_ITEMS invokes the BRM application programming interface (API) PCM_OP_AR_GET_ACCT_ACTION_ITEMS and returns the adjustment outputs Invoke_PCM_OP_AR_GET_ACCT_ACTION_ITEMS_OutputVariable to QueryAccountBalanceAdjustmentBRMCommsProvABCSImpl.
10. QueryAccountBalanceAdjustmentBRMCommsProvABCSImpl transforms the BRM API output Invoke_PCM_OP_AR_GET_ACCT_ACTION_ITEMS_OutputVariable into enterprise business message (EBM) output QueryAccountBalanceAdjustmentListRespMsg and returns it to QueryAccountBalanceAdjustmentSiebelCommsReqABCSImpl.
11. QueryAccountBalanceAdjustmentSiebelCommsReqABCSImpl then transforms the QueryAccountBalanceAdjustmentListRespMsg into QueryAdjustmentRespMsg, which is returned to AdjustmentSiebelCommsReqABCS.
12. AdjustmentSiebelCommsReqABCS returns the QueryAdjustmentRespMsg to the calling CMUExternalAdjustments as QueryAdjustmentRespMsg.
13. CMUAdjustmentResponseMessage is then written to the Siebel Balance Summary virtual business component (VBC) for the subscribers.

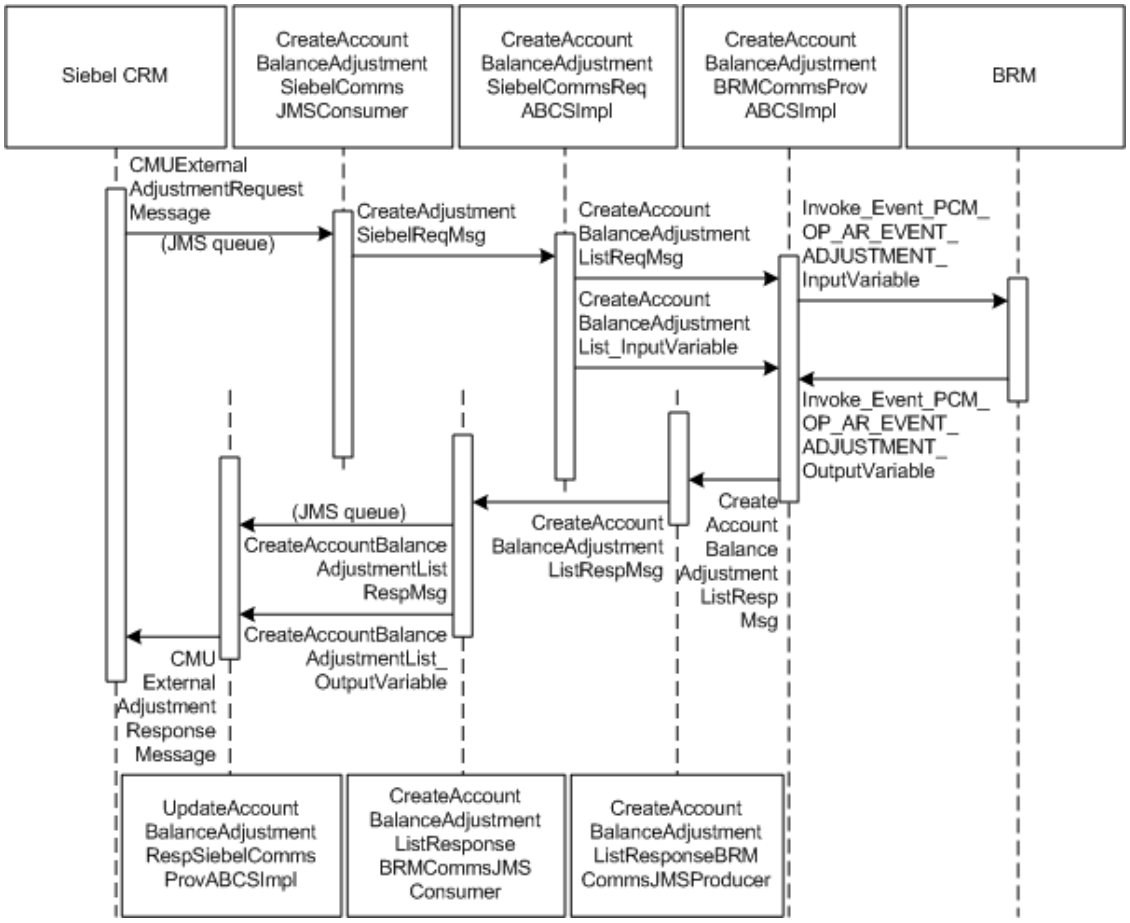
CreateAccountBalanceAdjustment Integration Flow

This integration flow uses the following interfaces:

- CreateAccountBalanceAdjustmentSiebelCommsReqABCSImpl
- CreateAccountBalanceAdjustmentBRMCommsProvABCSImpl
- UpdateAccountBalanceAdjustmentRespSiebelCommsProvABCSImpl
- CreateAccountBalanceAdjustmentListResponseBRMCommsJMSConsumer
- CreateAccountBalanceAdjustmentListResponseBRMCommsJMSProducer
- CreateAccountBalanceAdjustmentSiebelCommsJMSConsumer

Figure 3-5 illustrates the CreateAccountBalanceAdjustment integration scenario.

Figure 3-5 CreateAccountBalanceAdjustment Integration Flow Sequence Diagram



When you initiate the CreateAccountBalanceAdjustment process, the following events occur:

1. In Siebel CRM, navigate to the Billing Profile screen.
2. Navigate to the Accounts screen, query an account, and click a billing profile for the account. On the Billing Profile screen, the **Adjustment** tab displays all the adjustment requests.
3. To create an adjustment for an invoice, click the **Bills** tab.

   Select the bill against which an adjustment request must be created.
4. In the **Bill Details** view, an adjustment request can be captured at the header, item, or event level. Each of these sections in the view has an **Adjust** button.

   Clicking this button creates a new adjustment request at that level.
5. To create an adjustment at the event-level, click the **Net Amount** link for the required item charge.

   This opens the **Event Details** view to create the adjustment.
6. At the event-level, adjustments can be created for both monetary and nonmonetary resources such as free minutes.
7. You can also create adjustments for unbilled usage.

Unbilled adjustments are applicable only at the event-level for both monetary and nonmonetary resources.

8. The **Adjust** button on different screens calls the same web service to create the adjustment.

   But on the BRM side, it is based on the adjustment level. The adjustment type is set by Siebel CRM and the correct opcode is called based on the value in this column.

9. After creating the new adjustment for a particular level and getting the approval, click the **Submit** button.

   A web service call is made to SWICreateAdjustment, which in turn puts the message into the Queue AIA_CMUREQADJIOJMSQUEUE along with a Simple Object Access Protocol (SOAP) envelope. After submission, the adjustment status changes to **Submitted** and the adjustment record becomes read-only. Adjustments are persisted in Siebel CRM and when the adjustments are accepted, the status of the record changes to **Posted**. If the changes are not approved, the status changes to **Not Posted**.

10. CreateAccountBalanceAdjustmentSiebelCommsJMSConsumer dequeues the message and transforms it into the Siebel request application business message (ABM) and routes the CreateAdjustmentReqMsg to CreateAccountBalanceAdjustmentSiebelCommsABCSImpl.

11. The CreateAccountBalanceAdjustmentSiebelCommsABCSImpl transforms the CreateAdjustmentReqMsg into CreateAccountBalanceAdjustmentList_InputVariable and routes the CreateAccountBalanceAdjustmentList_InputVariable to the appropriate billing system.

    As delivered, CreateAccountBalanceAdjustmentList_InputVariable is routed to CreateAccountBalanceAdjustmentBRMCommsProvABCSImpl.

12. CreateAccountBalanceAdjustmentBRMCommsProvABCSImpl first checks the parameters of the Adjustment type coming from the request (Billed or Unbilled) and based on the parameter, PCM_OP_AR_EVENT_ADJUSTMENT is invoked with the appropriate data.

13. After checking the parameters of a particular service from request, the CreateAccountBalanceAdjustmentBRMCommsProvABCSImpl service transforms the CreateAccountBalanceAdjustmentListReqMsg into the BRM input flist message and invokes the opcode.

14. Invoking BRM API PCM_OP_AR_EVENT_ADJUSTMENT with account ID, billing profile ID, and event ID returns the list of events associated with items.

15. BRMARServices sends the response back to CreateAccountBalanceAdjustmentBRMCommsProvABCSImpl service with the list of adjustments and status for opcode calls.

    The **Status** field is mapped to the response and returned to the Siebel CRM user interface (UI).

16. After getting the response back from BRMARService, the status of the adjustment in CreateAccountBalanceAdjustmentBRMCommsProvABCSImpl is checked.

    If the status value returns as *Fail*, an error-handling framework service is called. This error-handling framework service calls the different error-handling services and logs the error for that particular failed adjustment request in the Admin Console.

17. Administrators can view the status, adjustment ID, and integration ID of a particular failed adjustment request.

    Administrators can also get the BRM description for a failed request.

18. CreateAccountBalanceAdjustmentBRMCommsProvABCSImpl then takes the response from the service, transforms

Invoke_Event_PCM_OP_AR_EVENT_ADJUSTMENT_OutputVariable to the CreateAccountBalanceAdjustmentListRespMsg, and routes it to the CreateAccountBalanceAdjustmentListResponseBRMCommsJMSProducer.

19. CreateAccountBalanceAdjustmentListResponseBRMCommsJMSProducer then puts the message into the queue AIA_CRTADJLSTRSPJMSQUEUE.
20. CreateAccountBalanceAdjustmentListResponseBRMCommsJMSConsumer picks the message from AIA_CRTADJLSTRSPJMSQUEUE and routes the CreateAccountBalanceAdjustmentListResponseMsg to UpdateAccountBalanceAdjustmentRespSiebelCommsProvABCSImpl.
21. UpdateAccountBalanceAdjustmentRespSiebelCommsProvABCSImpl transforms the CreateAccountBalanceAdjustmentListRespMsg into SWISIAAdjustmentIO and invokes the Siebel Update web service with this message.

> **Note**
>
> In the case of billed adjustments at the Bill and Item levels, the BRM provider calls PCM_OP_AR_BILL_ADJUSTMENT and PCM_OP_AR_ITEM_ADJUSTMENT opcodes.

BRM Interfaces

The QueryAdjustment integration flow uses:

- BRMARServices with operation PCM_OP_AR_GET_ACCT_ACTION_ITEMS

The CreateAdjustment integration flow uses:

- BRMARServices with operation PCM_OP_AR_EVENT_ADJUSTMENT

See *Oracle Communications Billing and Revenue Management Opcode Flist Reference* for more information.

Siebel CRM Interfaces

The View and Capture Adjustments integration flow uses these Siebel CRM interfaces:

- SWICreateAdjustment: To submit the adjustment
- SWIAdjustmentStatusUpdate: To update the adjustment

See *Siebel Order Management Guide Addendum for Communications* for more information about these web services.

Industry Oracle AIA Components

The View and Capture Adjustments integration uses the following delivered enterprise business objects (EBOs) and enterprise business messages (EBMs):

- AccountBalanceAdjustmentEBO
- QueryAccountBalanceAdjustmentListEBM
- QueryAccountBalanceAdjutsmentListResponseEBM
- CreateAccountBalanceAdjustmentListEBM
- CreateAccountBalanceAdjustmentListResponseEBM

The following directories contain the industry component files:

- Enterprise business object (EBO) and enterprise business message (EBM) XML schema files:

  COMMS_AIA_HOME/**comms_home/source/soainfra/apps/AIAMetaData/AIAComponents/EnterpriseObjectLibrary/Industry/Communications/EBO/**

- Enterprise business service (EBS) WSDL files:

  COMMS_AIA_HOME/**comms_home/source/soainfra/apps/AIAMetaData/AIAComponents/EnterpriseBusinessServiceLibrary/Industry/Communications/EBO/**

For detailed documentation of individual EBO and EBM, click the AIA Reference Doc link on EBO and EBM detail pages in the Oracle Enterprise Repository (OER).

EBOs can be extended, for instance, to add new data elements. These extensions are protected and remain intact after a patch or an upgrade, so long as the extensibility guidelines are followed.

See the discussion of Oracle AIA assets extensibility patterns in *Oracle Fusion Middleware Developer's Guide for Oracle SOA Core Extension* for more information about extending EBOs.

Integration Services

These services are delivered with the Adjustment Integration flow:

- AdjustmentSiebelCommsReqABCS
- QueryAccountBalanceAdjustmentSiebelCommsReqABCSImpl
- QueryAccountBalanceAdjustmentBRMCommsProvABCSImpl
- CreateAccountBalanceAdjustmentSiebelCommsABCSImpl
- CreateAccountBalanceAdjustmentBRMCommsProvABCSImpl
- UpdateAccountBalanceAdjustmentRespSiebelCommsProvABCSImpl
- CreateAccountBalanceAdjustmentListResponseBRMCommsJMSConsumer
- CreateAccountBalanceAdjustmentListResponseBRMCommsJMSProducer
- CreateAccountBalanceAdjustmentSiebelCommsJMSConsumer

Some of these services have been enabled to use Session Pool Manager.

See *Oracle Application Integration Architecture Pre-Built Integrations Utilities Guide* for more information about Session Pool Manager.

AdjustmentSiebelCommsReqABCS

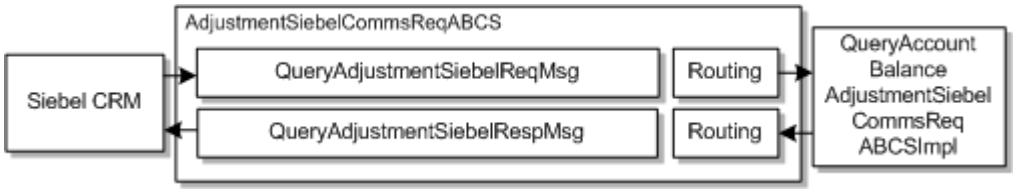
AdjustmentSiebelCommsReqABCS exposes the following operation related to Account Balance Adjustment on the Siebel ABM:

QueryAdjustment:

Routes QueryAdjustmentReqMsg to the provider implementation service
Routes QueryAdjustmentRespMsg to the requester

Figure 3-6 illustrates the relationship of AdjustmentSiebelCommsReqABCS with the other services in the integration flow.

Figure 3-6 AdjustmentSiebelCommsReqABCS



QueryAccountBalanceAdjustmentSiebelCommsReqABCSImpl

QueryAccountBalanceAdjustmentSiebelCommsReqABCSImpl transforms the Siebel message into the AccountBalanceAdjustmentEBM and calls the provider to get the Adjustment response from the billing system. It then transforms the AccountBalanceAdjustmentEBM response back to a Siebel message and returns it to the calling Siebel web service.

Figure 3-7 illustrates the relationship of QueryAccountBalanceAdjustmentSiebelCommsReqABCSImpl with other services in the integration flow.

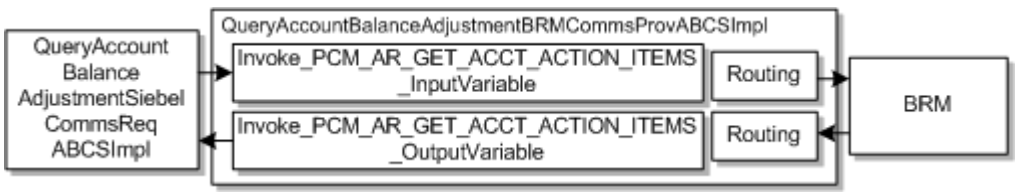
Figure 3-7 QueryAccountBalanceAdjustmentSiebelCommsReqABCSImpl

QueryAccountBalanceAdjustmentBRMCommsProvABCSImpl

QueryAccountBalanceAdjustmentBRMCommsProvABCSImpl transforms the AccountBalanceAdjustmentEBM message into a BRM API input format message and calls the API to get the adjustment details from the billing system. It then transforms the output from the API back to an AccountBalanceAdjustmentEBM message and returns it to the calling service.

Figure 3-8 illustrates the relationship of QueryAccountBalanceAdjustmentBRMCommsProvABCSImpl with other services in the integration flow:

Figure 3-8 QueryAccountBalanceAdjustmentBRMCommsProvABCSImpl



CreateAccountBalanceAdjustmentSiebelCommsABCSImpl

CreateAccountBalanceAdjustmentSiebelCommsABCSImpl transforms the Siebel message into an AccountBalanceAdjustmentEBM message format and calls the provider to create the

adjustment and get the response from the billing system. It then transforms the EBM response message to a Siebel message and routes it back to the AdjustmentSiebelCommsReqABCS service WSDL.

CreateAccountBalanceAdjustmentBRMCommsProvABCSImpl

CreateAccountBalanceAdjustmentBRMCommsProvABCSImpl transforms the AccountBalanceAdjustmentListEBM into a BRM API input format and calls the APIs to get the bill, item charge, and event details or Resource Impact Event adjustments from the billing system. It then transforms the output from the APIs back to an AccountBalanceAdjustmentListResponseEBM format that returns it to the calling service.

UpdateAccountBalanceAdjustmentRespSiebelCommsProvABCSImpl

UpdateAccountBalanceAdjustmentRespSiebelCommsProvABCSImpl transforms the AccountBalanceAdjustmentListResponseEBM into Siebel web service input formats and calls the web service to update the adjustment status.

This service is Session Pool Manager enabled.

See *Oracle Application Integration Architecture Pre-Built Integrations Utilities Guide* for more information about Session Pool Manager.

CreateAccountBalanceAdjustmentListResponseBRMCommsJMSConsumer

This service picks the message from the queue AIA_CRTADJLSTRSPJMSQUEUE and routes the CreateAccountBalanceAdjustmentListResponseMsg to UpdateAccountBalanceAdjustmentRespSiebelCommsProvABCSImpl.

CreateAccountBalanceAdjustmentListResponseBRMCommsJMSProducer

This process gets the CreateAccountBalanceAdjustmentListResponseEBM message from CreateAccountBalanceAdjustmentBRMCommsProvABCSImpl and puts the message into the Queue AIA_CMUREQADJIOJMSQUEUE.

CreateAccountBalanceAdjustmentSiebelCommsJMSConsumer

This process picks the message with a SOAP envelope from the queue AIA_CMUREQADJIOJMSQUEUE, transforms the message into a Siebel Requestor ABM by opening the SOAP envelope, and calls the CreateAccountBalanceAdjustmentSiebelCommsABCSImpl with the ListOfCmuRequestAdjustmentIO message.

About Collections Management

Collections management lets you synchronize collection actions (based on the specified collection scenarios) defined in BRM to Siebel CRM, and administer these collection actions and credit alerts.

It consists of the following integration flows:

- Synchronizing Collection Scenarios and Actions integration flow
- Synchronizing Credit Alerts integration flow

See About Collections Management for more information.

Synchronizing Collection Scenarios and Actions Flow - BRM to Siebel CRM

This section provides an overview of the Synchronizing Collection Scenarios and Actions flow and discusses how to:

- Configure the **AIAConfigurationProperties.xml** file in an Oracle AIA layer
- Receive and transform the data

For information about configuring BRM for integrated collections management, see:

- Setting Up BRM
- *Oracle Communications Billing and Revenue Management Concepts*
- *Oracle Communications Billing and Revenue Management Collections Manager*

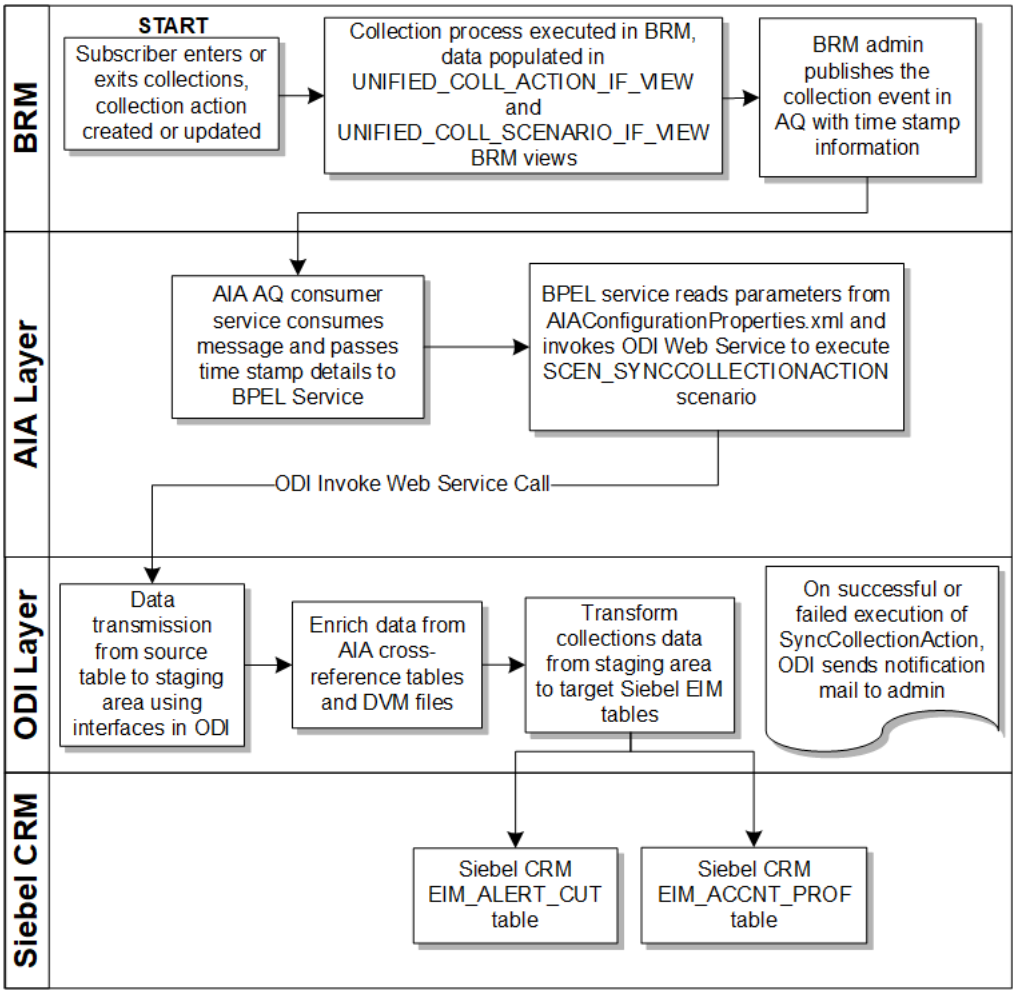
For information about managing collections in Siebel CRM and configuring Siebel CRM for integration, see *Siebel Communications Guide*.

About the Synchronizing Collection Scenarios and Actions Flow

The synchronizing collection scenarios and actions flow enables the transfer of collections data from BRM database views to Siebel Enterprise Integration Manager (EIM) tables. This integration uses ODI to transfer the data from BRM to Siebel CRM.

Figure 3-9 illustrates the synchronizing collection scenarios and actions flow.

Figure 3-9 Collections Data Synchronizing Activity Flow Diagram



When you initiate this process by creating a new collections action or updating an existing collections action, the following events occur:

1. A collections process is started in BRM and the actions data is populated into two database views:
   - UNIFIED_COLL_ACTION_IF_VIEW which contains all the collections action data (for example, to send a dunning letter)
   - UNIFIED_COLL_SCENARIO_IF_VIEW which contains the status of billing profiles (for example, Entered Collections or Exited Collections).
2. When the collections process is finished, BRM creates a CollectionsInfoChange business event and publishes it to an Advanced Queue (AQ) database queue with the start and end timestamp for the collection event. The timestamp information is in Unix format, such as 1319357706.
3. The SyncCollectionsInfoChangeBRMAQ service receives an AQ queue business event notification message and passes the timestamp details to SyncCollectionHeaderInfoBRMCommsReqImpl BPEL service.
4. The SyncCollectionHeaderInfoBRMCommsReqImpl BPEL service reads the required parameters from **AIAConfigurationProperties.xml** file and calls the ODI Web Service to start the SCEN_SYNCCOLLECTIONACTION scenario.

5. SCEN_SYNCCOLLECTIONACTION process starts ODI interfaces to perform the following:
   - Reads data from UNIFIED_COLL_ACTION_IF_VIEW and updates this data by performing a cross-reference and DVM lookup, and querying Siebel CRM base tables.
   - Reads data from the UNIFIED_COLL_SCENARIO_IF_VIEW view and updates this data by performing a cross-reference and DVM lookup, and querying Siebel CRM base tables.
   - Updates the AccountID and BillingProfileID columns using cross-reference data.
   - Updates the Currency Code, Action Name, Priority, and similar columns using DVMs.
6. The updated data is mapped to the EIM_ALERT_CUT table. For every row inserted into the EIM tables, the value of the IF_ROW_STAT column is set to FOR_IMPORT.
7. The Entered Collections and Exited Collections rows are extracted from the EIM_ALERT_CUT table and written to the EIM_ACCNT_PROF table. This data is responsible for changing the status of Billing Profile.
8. A success or failure notification mail is sent to the administrator. If a failure message is received the administrator must take the appropriate action. See Handling Errors for more information.
9. Siebel CRM runs a batch job and moves the data from the EIM table to the base table and creates credit alerts.

Table 3-1 lists various flows and steps while synchronizing collection scenarios and actions from BRM to Siebel CRM.

Table 3-1 Synchronizing Collection Scenarios and Actions Flow

Work Location	Step
BRM	1. Collections actions are generated and stored in a database view. 2. Publishes an event to the AQ queue with timestamp details.
Integration Process (ODI)	3. AQ Consumer Service receives the message and passes on the details to a BPEL Service. 4. BPEL Service calls the ODI Web service with a timestamp value as input parameter. 5. Integration flow reads the messages from the BRM database views. 6. Data is collected from BRM, updated by cross-references and DVM lookup, and inserted into the Siebel EIM tables. 7. After the ODI Scenario is processed, a success or failure notification message is sent to the Administrator.
Siebel CRM	8. Siebel CRM runs a batch job to move data from the EIM table to the Base table and creates credit alerts. 9. If required, updates the billing profile.

Time Zone Handling

Oracle AIA does not do a time-zone conversion when synchronizing credit alerts from BRM to Siebel CRM.

BRM publishes collection StartTime and EndTime timestamp details in BRM local server time in epoch format.

Configuring and Generating Data in BRM

You must configure BRM for integrated collections management as described in:

- Setting Up BRM
- See Integrating Collections with External CRM Applications in *Oracle Communications Billing and Revenue Management Developer's Guide* for more information about integrating collections with external CRM application.
- See About Integrating Collections Manager with Custom Client Applications in *Oracle Communications Billing and Revenue Management Collections Manager* for more information about integrating Collections Manager with custom client applications.

When you have configured BRM for integrated collections management, BRM generates collections data as follows:

1. A subscriber enters collections and one of the following collections activities occurs:
   - An account enters or exits collections.
   - A collections action object is created or updated.
   - A schedule object is created.
2. A collections process is started in BRM and data is populated to the UNIFIED_COLL_ACTION_IF_VIEW and UNIFIED_COLL_SCENARIO_IF_VIEW views.

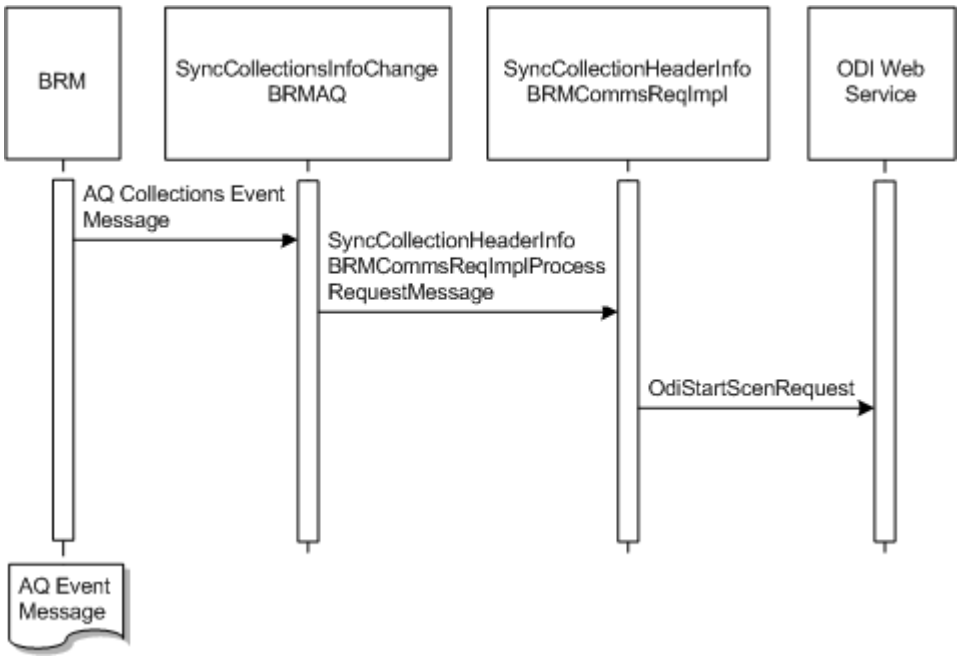
See *Oracle Communications Billing and Revenue Management Collections Manager* for more information about how BRM handles collections.

Receiving and Transforming the Data

The synchronizing collection scenarios and actions process flow uses ODI to receive, transform, and load the BRM data into the Siebel CRM EIM tables.

Figure 3-10 illustrates the synchronizing collection scenarios and actions from BRM to Siebel CRM flow.

Figure 3-10 Synchronizing Collection Scenarios and Actions from BRM to Siebel CRM Sequence Diagram



These tasks are performed as a part of the data load process:

1. In BRM the **pin_collections_process** utility generates the **/event/notification/collections/info_change** BRM business event.
2. The Data Manager AQ queue receives the payload data and publishes a new **CollectionsInfoChange** business event to the AQ queue. This includes the start and end timestamps, and the Account POID from the BRM event.
3. An Oracle AIA BPEL service SyncCollectionsInfoChangeBRMAQ receives a message from the AQ queue that new data is available and starts the SyncCollectionHeaderInfoBRMCommsReqImpl BPEL service.
4. The SyncCollectionHeaderInfoBRMCommsReqImpl BPEL service receives the timestamp parameters and reads the required parameters from the **AIAConfigurationProperties** file. Using the parameters it calls the ODI web service to start the SCEN_SYNCCOLLECTIONACTION process.
5. SCEN_SYNCCOLLECTIONACTION process starts ODI to perform the following:
   - Reads data from the UNIFIED_COLL_ACTION_IF_VIEW view and updates this data by performing a cross-reference and DVM lookup, and querying Siebel CRM base tables.
   - Reads data from the UNIFIED_COLL_SCENARIO_IF_VIEW view and updates this data by performing a cross-reference and DVM lookup, and querying Siebel CRM base tables.
   - Updates the AccountID and BillingProfileID columns using cross-reference data.
   - Updates the Currency Code, Action Name, Priority, and similar columns using DVMs.
6. The updated data is mapped to the Siebel EIM_ALERT_CUT table.

7. The Entered Collections and Exited Collections rows are extracted from the EIM_ALERT_CUT table and written to the EIM_ACCNT_PROF table. This data is responsible for changing the status of Billing Profile.
8. For every row inserted in the EIM tables, the value of the IF_ROW_STAT column is set to FOR_IMPORT.

> **Note**
>
> Before ODI loads the enriched collections data into the Siebel EIM tables, it performs data cleansing. All existing records that were successfully imported into the Siebel base table are deleted from the EIM tables.

Configuring AIAConfigurationProperties file to work with ODI

This section discusses how to configure the **AIAConfigurationProperties.xml** file to work with ODI.

Based on your requirements, after you have installed the collection management feature, you need to change the default values of the collections parameters to suit your implementation.

To configure the **AIAConfigurationProperties.xml** file to change the default values, do the following:

1. Open *COMMS_AIA_HOME***/comms_home/source/soainfra/apps/config/AIAConfigurationProperties.xml** file in a text editor.

   See the discussion of updating files in AIA MDS in *Oracle Communications Application Integration Architecture Cloud Native Deployment Guide.*
2. Locate <ModuleConfiguration moduleName="CollectionsParameters"> and update the following default values:
   - FromMailAddress
   - ToMailAddress
   - MailServer
   - ODI.USERNAME
   - ODI.PASSWORD
   - ODI.WORKREPNAME
   - BatchSize
   - Default.SystemID
   - AIAHome
   - IF_ROW_BATCH_NUM_CreatedStart
   - IF_ROW_BATCH_NUM_CreatedEnd
   - IF_ROW_BATCH_NUM_UpdateStart
   - IF_ROW_BATCH_NUM_UpdateEnd
   - IF_ROW_BATCH_NUM_Error
   - AL_SOURCE_CD
   - AL_TYPE_CD

See Configuring Collections Management for more information about these properties.

Synchronizing Credit Alerts Flow - Siebel CRM to BRM

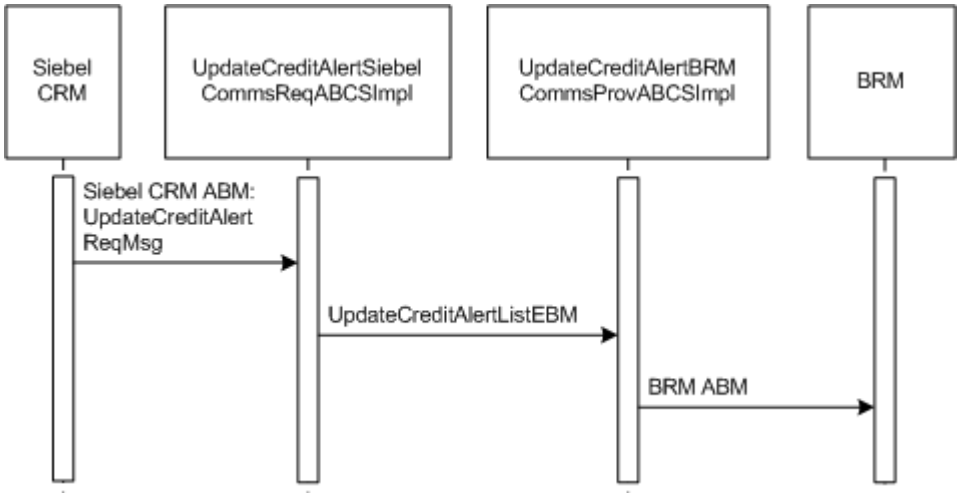
This flow synchronizes the credit alerts from Siebel CRM to BRM.

This flow uses the following services:

- UpdateCreditAlertSiebelCommsReqABCSImpl with operation UpdateCreditAlert
- UpdateCreditAlertBRMCommsProvABCSImpl with operation UpdateCreditAlert

Figure 3-11 illustrates the flow scenario for synchronizing credit alerts.

Figure 3-11 Synchronizing Credit Alerts from Siebel CRM to BRM Sequence Diagram



When you initiate the synchronizing credit alert process, the following events occur:

1. Siebel CRM starts Oracle AIA UpdateCreditAlertSiebelCommsReqABCSImpl service.
2. The UpdateCreditAlertSiebelCommsReqABCSImpl service transforms Credit Alert Status Siebel Application Business Message (ABM) ListOfSWICUTCreditManagementIO to the UpdateCreditAlertListEBM message. Based on the Billing Profile ID it sets the Target System ID in the EBM and calls the UpdateCreditAlertBRMCommsProvABCSImpl service.
3. The UpdateCreditAlertBRMCommsProvABCSImpl service searches for the Billing Profile ID, and sets the Target System ID and URL of target BRM instance.

   The UpdateCreditAlertBRMCommsProvABCSImpl service then transforms the UpdateCreditAlertListEBM message to an ABM and starts the PCM_OP_COLLECTIONS_SET_ACTION_STATUS BRM opcode.
4. The credit alert or action status is updated to **Closed** in BRM.

Table 3-2 lists various flows and steps while synchronizing credit alerts from Siebel CRM to BRM.

Table 3-2 Synchronizing Credit Alerts Flow

Work Location	Step
Siebel CRM	1. The action status for a billing profile is updated.
Integration Process (Oracle AIA)	2. Requester Application Business Connector Service (ABCS) receives the request from Siebel and transforms the application business specific message (ABM) to application neutral message (EBM). 3. Provider Application Business Connector Service (ABCS) transforms EBM to ABM and starts a BRM op-code.
BRM	4. The BRM op-code updates the status of the given collections action in BRM.

Handling Errors

Collection management handles BPEL services and ODI errors.

Handling BPEL Service Errors

For BPEL service errors, do the following:

SyncCollectionHeaderInfoBRMCommsReqImp

This service uses the default error handling capability of Oracle AIA.

- Faults: On error, the collections Application Business Message (ABM) is transformed to a fault message and returned to the calling process. The fault message contains the collections run timestamp details. The Error BPEL process is started. An exception is raised.
- Based on your Error Handling Notification configuration, an email is sent to the subscriber.

  According to the message received the subscriber must take the appropriate action.

UpdateCreditAlertSiebelCommsReqABCSImpl

This service uses the default error handling capability of Oracle AIA.

- Faults: On error, the Credit Alert ABM is transformed to a fault message and returned to the calling process. The fault message contains the Credit Alert details. The Error BPEL process is started. An exception is raised.
- Based on your Error Handling Notification configuration, an email is sent to the subscriber.

  According to the message received the subscriber must take the appropriate action.

UpdateCreditAlertBRMCommsProvABCSImpl

This service uses the default error handling capability of Oracle AIA.

- Faults: On error, the Credit Alert ABM is transformed to a fault message and returned to the calling process. The fault message contains the Credit Alert details. The error BPEL process is started. An exception is raised.
- Based on your Error Handling Notification configuration, an email is sent to the subscriber.

  According to the message received the subscriber must take the appropriate action.

Handling ODI Error Messages

After each scenario run, a notification email is sent to the subscriber's email address.

> **Note**
>
> The FromMailAddress and ToMailAddress must be set in the **AIAConfiguraionProperties.xml** file. See Configuring AIAConfigurationProperties file to work with ODI for more information.

On receiving a notification email, for failed jobs do the following:

1. Take the appropriate action based on the message received.
2. Re-start the scenario by doing one of the following:
   a. Manually restart the scenario. See Restarting Your Job Manually for more information.
   b. Re-run the job by passing the timestamp details to the SyncCollectionHeaderInfoBRMCommsReqImpl service.

Restarting Your Job Manually

The notification mail contains the start and end timestamps, for example:

```
Coll_StartTime: 1299794311
Coll_EndTime: 1299794315
```

To restart your job manually:

1. On the ODI system, navigate to the following file path:

   ODI_DOMAIN_HOME`/bin`

   where *ODI_DOMAIN_HOME* is the path to ODI Domain home.
2. Run the following command, where *Start_Time_Stamp* and *End_Time_Stamp* are the timestamp values shown in the notification email:
   - In a Linux environment:

     ```
     ./startscen.sh SYNCCOLLECTIONACTION 001 GLOBAL
     "GLOBAL.Coll_StartTime=Start_Time_Stamp"
     "GLOBAL.Coll_EndTime=End_Time_Stamp"
     ```

   - In a Windows environment:

     ```
     startscen.bat SYNCCOLLECTIONACTION 001 GLOBAL
     "GLOBAL.Coll_StartTime=Start_Time_Stamp"
     "GLOBAL.Coll_EndTime=End_Time_Stamp"
     ```

   For this command, ODI_HOME is the path of ODI home.

> **Note**
>
> Make sure that Java home is set in the path before you run this command.

BRM Interfaces

The synchronizing credit alerts flow uses the following BRM interface:

- PCM_OP_COLLECTIONS_SET_ACTION_STATUS

See *Oracle Communications Billing and Revenue Management Opcode Flist Reference* for more information.

Siebel CRM Interfaces

The synchronizing credit alerts flow uses this Siebel CRM interface:

- Outbound web service UpdateCreditAlertSiebelCommsReqABCSImplServicePort operation UpdateCreditAlert

See *Siebel Order Management Guide Addendum for Communications* for more information about this web service.

Industry Oracle AIA Components

The synchronizing credit alerts flow uses the following delivered enterprise business objects (EBOs) and enterprise business messages (EBMs):

- CreditAlertEBO
- UpdateCreditAlertListEBM

The following directories contain the industry component files:

- Enterprise business object (EBO) and enterprise business message (EBM) XML schema files:

  *COMMS_AIA_HOME***/comms_home/source/soainfra/apps/AIAMetaData/AIAComponents/EnterpriseObjectLibrary/Industry/Communications/EBO/**

- Enterprise business service (EBS) WSDL files:

  *COMMS_AIA_HOME***/comms_home/source/soainfra/apps/AIAMetaData/AIAComponents/EnterpriseBusinessServiceLibrary/Industry/Communications/EBO/**

For detailed documentation of individual EBOs and EBMs, click the AIA Reference Doc link on EBO and EBM detail pages in the Oracle Enterprise Repository (OER).

EBOs can be extended, for instance, to add new data elements. These extensions are protected and remain intact after a patch or an upgrade, so long as the extensibility guidelines are followed.

See the discussion of Oracle AIA assets extensibility patterns in *Oracle Fusion Middleware Developer's Guide for Oracle SOA Core Extension* for more information about extending EBOs.

Integration Services

These services are delivered as part of the collection management flow:

- UpdateCreditAlertSiebelCommsReqABCSImpl
- UpdateCreditAlertBRMCommsProvABCSImpl
- SyncCollectionsInfoChangeBRMAQ

- SyncCollectionHeaderInfoBRMCommsReqImpl

UpdateCreditAlertSiebelCommsReqABCSImpl

UpdateCreditAlertSiebelCommsReqABCSImpl is a Business Process Execution Language (BPEL) process. It performs the following actions:

- Receives Siebel Credit Alert ABM containing billing profile ID, credit alert ID, integration ID, and status.
- Transforms the ABM to UpdateCreditAlertListEBM and starts the UpdateCreditAlertBRMCommsProvABCSImpl service.

UpdateCreditAlertBRMCommsProvABCSImpl

UpdateCreditAlertBRMCommsProvABCSImpl is a BPEL process. It performs the following actions:

- Receives UpdateCreditAlertBRMCommsProvABCSImpl from the requestor service.
- Transforms the UpdateCreditAlertListEBM to BRM opcode PCM_OP_COLLECTIONS_SET_ACTION_STATUS flist.
- Starts BRM PCM_OP_COLLECTIONS_SET_ACTION_STATUS and updates the status of the credit alert or action to **Closed**.

SyncCollectionsInfoChangeBRMAQ

SyncCollectionsInfoChangeBRMAQ is a mediator based service. It performs the following actions:

- Reads the message from AQ event message published by BRM.
- Starts SyncCollectionHeaderInfoBRMCommsReqImpl

SyncCollectionHeaderInfoBRMCommsReqImpl

SyncCollectionHeaderInfoBRMCommsReqImpl a BPEL process. It performs the following actions:

- Receives the timestamp values.
- Reads the properties from the **AIAConfigurationProperties.xml** file
- Transforms the input message to an ODI web service message type.
- Starts the operation of the ODI web service.

Integrating Collections in a Multischema Environment

In BRM, you can distribute your data amongst multiple schemas in the same database to increase scalability and support load balancing. To distribute BRM accounts and associated objects amongst multiple schemas, you can migrate them between schemas in batches, with each batch consisting of a list of accounts identified by their BRM Portal Objects (POID), source, and destination schema numbers.

See *Oracle Communications Billing and Revenue Management Concepts* and *Oracle Communications Billing and Revenue Management System Administrator's Guide* for more information about multischema architecture and account migration.

To communicate the correct account information between BRM and Siebel CRM, a common Oracle AIA identifier and an Oracle AIA cross-reference table is used to map accounts between BRM and Siebel CRM. The Oracle AIA cross-reference table uses BRM POIDs that include the BRM schema number. The BRM POID consists of a schema number, an object type, a unique object ID, and a revision number. For example:

```
0.0.0.2 /account 11599 4
```

where:

- **0.0.0.2** is the BRM schema number
- **/account** is the object type
- **11599** is the unique object ID
- **4** is the revision number

To integrate collections with Siebel CRM in a multischema environment, BRM populates the custom views UNIFIED_COLL_ACTION_IF_VIEW and UNIFIED_COLL_SCENARIO_IF_VIEW in BRM collections tables with the POID schema number, hard-coded as 0.0.0.1 for **/collections_action** regardless of the schema where the account resides. For example, for the account in the secondary schema shown in the example above, the value for a late fee would be **0.0.0.1/collections_action/late_fee 100845 0** in the UNIFIED_COLL_ACTION_IF_VIEW.

By including the hard-coded POID schema number, the schema numbers of the action objects being published to Siebel CRM do not change in the custom views, even if the action objects are migrated from one schema to another. Siebel CRM always reads the objects as residing in schema 0.0.0.1, whether it is a single-schema or a multischema environment. During the Synchronizing Credit Alerts integration flow, when synchronizing credit alerts from Siebel CRM to BRM, the integration reads the schema number for the account object POID to get the appropriate schema number for the /collections_action POIDs.

Configuring Complaint-to-Solution Business Process

This section describes how to configure the complaint-to-solution business process.

About Configuring Collections Management

This section discusses how to set up Oracle Communications Billing and Revenue Management (BRM) and Siebel customer relationship management (Siebel CRM). In addition, it discusses how to work with domain value maps (DVMs) and cross references, how to handle errors, and how to configure the collections management flow.

Setting Up BRM

To set up BRM:

1. Specify pay types to process.
2. Set the minimum overdue balance to process.
3. Set the number of bill units retrieved during step searches.
4. Set up invoice reminders.
5. Define collections features.
6. Configure how Collections Manager determines dates.

See the discussion of setting up Collections Manager in *BRM Collections Manager Guide* for more information.

7. Create views on BRM and configure pin_collections_process to record start and end time.

Setting up Collection Action Names in BRM

As stated in About Collections Management collection management synchronizes collection actions between BRM and Siebel CRM. The collection actions are mapped to Siebel CRM as credit alerts using the **DVM COLLECTION_ACTIONNAME.xml** file.

As delivered, DVM mappings exist for the following collections actions:

- Impose Late Fee
- Courtesy Phone Call
- Courtesy Email or SMS Reminder
- Courtesy Dunning Letter
- Demanding Phone Call
- Inactivate Services of Billinfo
- Harsh Dunning Letter
- Refer to outside agency
- Writeoff Billinfo
- Close Services of Billinfo
- Inactivate the Account

Of the 11 collections actions, these 4 collections actions are seeded in BRM and are available as part of the product installation.

- Inactivate Services of Billinfo
- Close Services of Billinfo
- Writeoff Billinfo
- Refer to outside agency

Before using collections management, the implementer can either add the remaining seven collection actions in BRM or run the collection integration process with the four seeded values listed previously. If the implementer's intent is to add these additional collection actions, then they must be entered manually in Collection Manager:

- Impose Late Fee
- Courtesy Phone Call
- Courtesy Email or SMS Reminder
- Courtesy Dunning Letter
- Demanding Phone Call
- Harsh Dunning Letter
- Inactivate the Account

> **Caution**
>
> Use caution when manually entering these action names in BRM because any change in spelling or case breaks the Sync Collection Action flow. The DVM lookup fails due to a mismatch of strings.

> **Note**
>
> If the intent of the implementer is to use new or different collection action names, then explicit changes must be made in the COLLECTION_ACTIONNAME DVM before you use collections management.

Setting Up Siebel CRM

For some Siebel CRM interfaces, you must set the UTCCanonical process property to **Y** in Siebel.

See instructions for ACR 474 and ACR 508 in *Siebel Maintenance Release Guide* for more information about which Siebel CRM interfaces require you to enable the UTCCanonical process property.

The credit alert appears in the subscriber service supervisor's queue by default, and the supervisor assigns the credit alert to a Customer Service Representative (CSR). Siebel CRM can also be customized to automatically assign these credit alerts to a CSR based on a set of criteria as defined by the service provider.

To set up Siebel CRM:

1. Extract and copy the .ifb files from the ACR 463/EIM folder.
2. Place the **ProcessAlertsAndBillingProf.ifb** file on the server path: Siebsrvr/Admin.

To configure Siebel CRM:

1. Make sure that the Siebel server is running.
2. Make sure that the Siebel Enterprise Integration Manager (EIM) component is online.
3. For creation of the EIM job:
   a. Go to **SiteMap, Administration - Server Management, Jobs**
   b. Click the **New** button and select *Enterprise Integration Manager* in the **Component/Job** tab.
   c. Complete the following parameters on the Job Parameters applet: **Configuration file = ProcessAlertsAndBillingProf.ifb**. Enter appropriate settings for **Error Flag, Trace Flag**, and **SQL Trace Flag**.
4. Click the **Submit Job** button.
5. Make sure that the status changes to **Success** for the job that you have run.
6. After the EIM job is done, query the IF_ROW_STAT column of EIM_ALERT_CUT table to ensure that all the records have been loaded in the Siebel base tables correctly.
7. Check the Siebel CRM user interface (UI) to ensure that the data appears in corresponding views of the Credit Management screen.

See the discussion of configuring EIM interfaces in *Configuring Siebel Business Applications* for more information.

For Oracle Data Integrator (ODI)-based collection flows, during the transfer of data from BRM to Siebel CRM, ODI creates some temporary tables in the Siebel database. Once the data is successfully written to the Siebel EIM table, these temporary tables are dropped. Therefore, the Siebel database administer must grant *create table* privilege so that ODI can create the temporary tables.

See *Oracle Fusion Middleware Developer's Guide for Oracle Data Integrator* for more information on using ODI.

Setting Up Split Cross-Reference Tables

To maintain performance levels when looking up values in large cross-reference tables, Oracle recommends splitting the cross-reference tables into multiple tables, with one table for each cross-reference object. You split cross-reference tables using the XREF Migration Utility and add the custom table names to the **AIAConfigurationProperties.xml** file.

To set up split cross-reference tables:

1. Split the collections cross-reference tables using the XREF Migration Utility.
2. Open **AIAConfigurationProperties.xml** in a text editor.
3. Add a comma-separated list of the names of your custom cross-reference tables.
4. Save and close the file.

Working with DVMs

Domain value maps (DVMs) are a standard feature of the Oracle service-oriented architecture (SOA) Suite that enables you to equate lookup codes and other static values across applications, for example, **FOOT** and **FT** or **US** and **USA**.

DVMs are static in nature, though administrators can add maps as required. Transactional business processes never update DVMs-they only read them. They are stored in XML files and cached in memory at run time.

DVM types are seeded for the Oracle Communications Billing and Revenue Management: Cash to Care business processes. Administrators can extend the list of mapped values by adding more maps.

Table 3-3 lists the DVMs for the collections management flow:

Table 3-3 Collections Management DVMs

DVM	Description
COLLECTION_ACTIONNAME.xml	DVM mapping for action name
COLLECTION_PRIORITY.xml	DVM mapping for priority
COLLECTION_STATUS.xml	DVM mapping for status
COLLECTION_SUBSTATUS.xml	DVM mapping for sub-status
CURRENCY_CODE.xml	DVM mapping for currency code

> **⚠ Caution**
>
> DVMs are stored in the Metadata Services (MDS) repository, which uses the database persistence, and are managed using tools provided by JDeveloper or Foundation Pack.

See the discussions of working with message transformations, DVMs, and cross references in *Oracle Fusion Middleware Developer's Guide for Oracle SOA Core Extension* for more information.

Working with Cross References

Cross references map and connect the records within the application network, and they enable these applications to communicate in the same language. The integration server stores the relationship in a persistent way so that others can refer to it.

Table 3-4 lists the collections management cross references:

Table 3-4 Collections Management Cross References

Cross Reference Table Name	Column	Description
CUSTOMERPARTY_ACCOUNTID	BRM/ SIEBEL	Used for cross-reference lookup for account ID.
CUSTOMERPARTY_BILLPROFILEID	BRM/ SIEBEL	Used for cross-reference lookup for billing profile ID.

Handling Errors

Handling BRM Errors

Error details are logged in the **CollectionsErrorLog.txt** file. This log file is available in the following directory:

User_Home/DISHome/CollectionsHome

where *User_Home* is the UNIX user home path.

Handling Siebel CRM Errors

After every EIM load, check the value of the IF_ROW_STAT column in the EIM_ALERT_CUT table. The status is imported for successful loading. If the status is not imported, errors occurred during data load from the Siebel EIM tables to the Siebel base tables. Perform the following actions for this type of error:

1. From the **Jobs** view in the Administration - Server Management screen, navigate to the **Tasks** view in the same screen.
2. Get the task number from the **Task** field.
3. Navigate to the `Siebsrvr\Log` directory in the server and locate the log file by querying with the same task number.
4. The log file contains details of every level of EIM processing and errors if any.
5. Run the EIM job with appropriate .ifb property values after modifying the values for failed records.

See the discussion of resolving import processing problems in *EIM Administration Guide* for more information about EIM Error handling.

For more information about the errors generated by BRM and Siebel CRM applications, see the documentation for that product.

For details on setting up error notifications using these values, see the discussions on error notifications and trace and error logs in the *Oracle Fusion Middleware Infrastructure Components and Utilities User's Guide* for Oracle Application Integration Architecture Foundation Pack.

Configuring the Collections Management Flow

Configure these properties in the **AIAConfigurationProperties.xml** file. The file is located in *COMMS_AIA_HOME***/comms_home/source/soainfra/apps/config**. Entries in the **AIAConfigurationProperties.xml** file are case-sensitive. See Table 3-5.

See the discussion of updating files in AIA MDS in *Oracle Communications Application Integration Architecture Cloud Native Deployment Guide.*

Table 3-5 AIAConfigurationProperties.xml - Property Values

Property Name	Value/Default Values	Description
FromMailAddress	user@oracle.com After installation, administrators or users can change this value.	This property is required to send a notification mail to user/admin.
ToMailAddress	user@oracle.com After installation, administrators or users can change this value	This property is required to send a notification mail to user/admin
MailServer	mail.oracle.com After installation, administrators or users can change this value.	This property is required to send a notification mail to user/admin.
ODI.USERNAME	odi.username After installation, administrators or users can change this value.	This property is read by SyncCollectionHeaderInfoBRMCommsReqImpl process to invoke ODI Web Service.
ODI.PASSWORD	odi.password	This property is read by SyncCollectionHeaderInfoBRMCommsReqImpl process to invoke ODI Web Service. The odi.password value for this property should be retained as is because at run-time an encrypted ODI password is read from **AIAInstallProperties.xml** file.
ODI.WORKREPNAME	odi.workrep.name After installation, administrators or users can change this value.	This property is read by SyncCollectionHeaderInfoBRMCommsReqImpl process to invoke ODI Web Service.
AIAHome	No default value. A value is entered during installation.	This property contains the absolute path to AIA Home.
BatchSize	Default value = 5000 After installation, administrators or users can change this value.	This property specifies the number of messages to dequeue in one batch. See Changing the BatchSize Default Value for information about changing the batch size default value.

Table 3-5 (Cont.) AIAConfigurationProperties.xml - Property Values

Property Name	Value/Default Values	Description
Default.SystemID	SEBL_01	This property gives the system ID of Siebel CRM.
IF_ROW_BATCH_NUM_Created Start	Default value = 100. After installation, administrators or users can change this value.	This property indicates the starting value of the batch number (IF_ROW_BATCH_NUM) for the rows for new collection actions created in the Siebel EIM table.
IF_ROW_BATCH_NUM_Created End	Default value = 199 After installation, administrators or users can change this value.	This property indicates the end value of the batch number (IF_ROW_BATCH_NUM) for the rows for new collection actions created in the Siebel EIM table.
IF_ROW_BATCH_NUM_Update Start	Default value = 200 After installation, administrators or users can change this value.	This property indicates the starting value of the batch number (IF_ROW_BATCH_NUM) for the updated collection actions in the Siebel EIM table.
IF_ROW_BATCH_NUM_Update End	Default value = 299 After installation, administrators or users can change this value	This property indicates the ending value of the batch number (IF_ROW_BATCH_NUM) for the updated collection actions in the Siebel EIM table.
IF_ROW_BATCH_NUM_Error	Default value = 50 After installation, administrators or users can change this value.	This property defines the value for the IF_ROW_BATCH_NUM for the unsuccessful collections actions during importing from the Siebel EIM table.
AL_SOURCE_CD	Customer	This property is used to update the AL_SOURCE_CD value in the Siebel EIM table.
AL_TYPE_CD	Credit	This property is used to update the AL_TYPE_CD value in the Siebel EIM table.

Changing the BatchSize Default Value

Changing the BatchSize default value has dependencies on the batch size that is defined in Siebel. Before you make any changes to this property, check Siebel documentation to find out the optimal batch size and accordingly make changes to this property.

For example, if the BatchSize property is changed to 5000 and the IF_ROW_BATCH_NUM_CreatedStart = 100 and IF_ROW_BATCH_NUM_CreatedEnd = 199, then ODI can simultaneously dequeue 5000*100 = 500000 records into Siebel EIM table. Siebel batch size for the EIM table load must match so that it can manage this data upload from ODI.

4

Usage-to-Payment Business Process

Learn about the Usage-to-Payment business process and its features available in the Oracle Communications Digital Business Experience Reference Solution package.

Overview of the Usage-to-Payment Business Process

The Usage-to-Payment business process is a foundational component of the Oracle Communications Digital Business Experience Reference Solution (Reference Solution), enabling telecom operators to convert customer service usage into revenue. This process begins when service consumption events, such as voice calls, data sessions, and messaging are captured from network elements and mediation systems. These usage records are then normalized, validated, and rated based on predefined tariffs and business rules to determine the corresponding charges.

The calculated charges are aggregated along with recurring fees and any applicable discounts or adjustments to produce comprehensive customer invoices within defined billing cycles. The Reference Solution supports a variety of payment methods, allowing customers to settle their invoices through channels, such as credit/debit card, bank transfer, or electronic payment gateways.

Throughout the process, the Reference Solution enforces compliance, data integrity, and auditability, supporting timely payment collection and enabling efficient revenue management. This end-to-end automation streamlines the operator’s revenue assurance activities, minimizes revenue leakage, and improves the overall customer billing experience.

This business process includes the following features:

- Manage Customer Billing
- Payment and Receivables
- Charging and Call Control
- Manage Collections

The sections that follow briefly describe each of the above features.

About Manage Customer Billing Process

This process supports telecom operators in accurately calculating, generating, and administering customer invoices for a wide range of telecommunication services. This process encompasses the consolidation of all customer account data, including service usage records, recurring charges, one-time fees, applied discounts, and applicable taxation.

At the conclusion of each billing cycle, this process compiles and verifies all relevant financial information for individual customers or accounts. Using configurable rules and billing schedules, the system produces detailed invoices that reflect every component of a customer's subscription and consumption profile. These invoices are then distributed through multiple channels, such as electronic delivery, customer self-service portals, or print, conforming to regulatory and contractual obligations.

This process ensures data integrity, compliance, and transparency throughout the billing lifecycle, supporting customer inquiries, adjustments, and dispute management as needed. By automating billing operations and integrating with downstream payment and finance systems, this process significantly enhances billing accuracy, reduces operational overhead, and strengthens customer trust in the operator's billing practices.

Implementing Billing

This feature enables telecom operators to generate comprehensive bill content objects from unbilled charges. This feature ensures each bill is accurate, compliant with local market regulations, and suitable for Business-to-Business (B2B) and Business-to-Customer (B2C) customer presentation. Bill runs are scheduled and performed as part of regular billing operations, based on predefined cycles, billing segments, or billing profile specifications.

The billing execution process consists of the following:

1. **Bill Run Scheduling**
   - **Frequency**: Bills are run one or several times per month, according to customer billing segments or Billing Date of Month (BDOM) specified in the Billing Profile.
   - **Billing windows**: Execution occurs on predefined days of the month.
   - **Applicable customers**:
     - B2B and B2C postpaid customers on monthly billing cycles.
     - B2B customers with quarterly or annual billing cycles when the billing cycle completes.
   - **Initiation**: Business Operation Center (BOC) schedules bill runs, potentially executing in multiple phases to account for delayed events (such as those from offline charging).
2. **Bill Generation Logic**
   - **Account processing**:
     - Service accounts are processed sequentially, respecting the billing hierarchy or other grouping dependencies.
     - Billing accounts aggregate accumulated service account charges for the cycle.
   - Each generated bill includes:
     - **Bill Number**: Consists of a fixed or dynamic prefix.
     - **Due Date**: Calculated based on payment terms, adjusted for calendar dates and any configured public holidays.
     - **Total Amount**: Rounded to the local currency precision.
     - **Cyclic Charges and Rollovers**: Cyclic fees, counter replenishments, and rollovers may be applied for the current and/or next bill/accounting period, with proration rules as per the configuration.
     - **Balance Forward**: Optionally includes previous unpaid bill amounts (Balance Forward configuration). When Balance Forward is enabled, previous unpaid totals are carried into the new bill's total due.
     - **Volume Tier Discounts**: Applied if volume-based discounting is enabled.
     - **Charge Breakdown**: Detailed charges by service and type (Non-Recurring Charges [NRC], Monthly Recurring Charges [MRC], and Usage).
     - **Bill Status**: Set to **Open**; once finalized, no additional charges can be added. Upon successful bill creation, subsequent invoicing and PDF rendering are

implemented via Oracle Communications Billing and Revenue Management (BRM) bulk utilities.

> **Note**
>
> If a subscriber is eligible and requests termination of the contract, the system must issue a final bill before contract termination.

Exception management

Bills that fail during the process are reported by BOC. These must be investigated, corrected, and reprocessed to ensure billing integrity. Re-invoicing scenarios are supported as per the configuration, and trial billing is available for non-posting invoice validation.

About Generate Bill Invoice

The Generate Bill Invoice feature transforms previously generated bill objects into finalized, customer-ready invoices. This process applies to bills created through scheduled bill runs or corrective billing operations and ensures that invoices comply with formatting, legal, and regulatory requirements.

The invoice generation process involves the following steps:

1. **Standard Payload Generation**
   - The system creates an XML payload that represents the final invoice layout.
   - The **pin_inv_accts** script handles this step in bulk for efficiency and selects the appropriate invoice template key (for example, B2C, B2B, Credit Note, Re-billing, and so on) based on customer data and document reason.
   - The XML payload is enriched with customer and bill-specific data, including template selection based on customer type (for example, B2B, B2C, credit note), tax breakdowns, rate codes, and itemization aligned with invoice structure rules.
2. **PDF Rendering**
   - The **pin_inv_doc_gen** script converts the structured XML payloads into human-readable PDF invoices.
   - The Oracle Analytics Publisher (OAP) formatting system applies predefined templates and further enriches the output as required (for example, logos, contact numbers, payment instructions, optional marketing banners, and region-specific text).

The invoice contains the following sections, rendered to comply with local regulations and business requirements:

- **Summary**
  - Customer name and billing address
  - Invoice date and bill run date
  - Bill number
  - Due date (ensuring legal minimum delay between delivery and due date)
  - Total due amount
  - Payment method
- **Charge Aggregation**

  - Itemized periodic, one-time, and discount charges
  - Grouped by type of service
  - Rounded to local currency precision
- **Usage Details**
  - Optional and configurable by a billing preference flag
  - Details per usage type and service, with corresponding dates and tax information

The following additional formatting is done to the invoices:

- Application of logo, contact details, payment instructions, and optional marketing messages.
- Insertion of QR codes or barcodes for manual payment facilitation (using bill number and URI).
- Generation of QR codes for specific local regulatory requirements (for example, tax declarations), if applicable and supported.

Operational considerations

- Scheduling
  - Invoicing processes are triggered by the Business Operation Center (BOC) after successful bill execution for postpaid and hybrid customers.
  - The system ensures invoice delivery dates comply with legal requirements, typically enforcing a minimum notice period (for example, two weeks) before the due date.
- Distribution
  - Upon successful generation, invoice metadata is published to a queuing system for downstream processing and delivery.
  - Failed or incomplete invoice batches are monitored and retried through the BOC user interface.
- Re-invoicing
  - Re-invoicing occurs for various scenarios:
    * Address/contact changes: Regenerate the invoice PDF using the same bill object (no financial change).
    * Credit/Debit Notes: Issued as separate bill objects with dedicated prefix/series and templates; not a modification of the original invoice.
    * Other financial alterations require issuing a new corrective document

Exception management

- Any failures in the bulk invoicing process are reported and can be resolved by regenerating invoices using BOC tools.
- All invoice format changes or customer-driven alterations require the generation and delivery of an updated invoice.

About Invoice Template Customization

The Invoice Template Customization feature enables organizations to define and manage a range of invoice render templates tailored to various business scenarios and customer segments. This feature supports compliance, operational efficiency, and customer

communication requirements by allowing flexible and precise control over invoice formatting and presentation.

Invoice designers utilize the Oracle Analytics Publisher (OAP) rendering system to organize and maintain XSL/XSD-based templates, which format invoice XML payloads into final PDF documents. These templates accommodate different business purposes while ensuring information consistency and adherence to presentation standards, including dedicated templates for corrective documents (for example, Credit Notes).

Templates are specialized based on the following criteria:

- Product type
- Invoice reason (for example, final, hot, credit note, bill shock, and so on)
- Customer segment (for example, Business-to-Business (B2B), Business-to-Customer (B2C), and so on)
- Static messages (specific to business or regulatory needs)

Supported invoice render templates include, but are not limited to:

- Regular B2C invoice
- Regular B2B invoice
- Bill Now invoice
- Bill Threshold invoice
- Credit Note
- Re-billing (for B2B)

Note

- Select or modify templates according to business requirements and regulatory obligations.
- Incorporate static messaging and required invoice sections based on segment and context.
- Validate template output to ensure accuracy and compliance before mass deployment.
- Ensure numbering schemes and prefixes (for example, CN-xxxxx for Credit Notes) are configured before template.
- QR/barcode elements for manual payments may be included (bill number plus payment URI).
- Archive rendered PDFs to the designated Document Management System (DMS) upon successful generation.

About Trial Billing

The Trial Billing feature allows operators to validate the invoice output for newly introduced business products prior to full production rollout. This process enables targeted testing of invoice formats and billing calculations using a subset of customer accounts, without affecting production balances or financial ledger entries.

This feature:

- Enables operators to test and review invoice output for new product launches.
- Facilitates assessment of invoice structure and charge calculations for selected accounts in the production environment.
- Prevents any impact on current account balances or general ledger data during the trial process.

The Trial Billing process involves the following steps:

> **Note**
>
> Before starting the Trial Billing process, you must provide a list of customer accounts to be included in the trial billing run.

1. **Trial Bill Generation**
   - Launch the batch trial billing job using Business Operation Center (BOC) or a command-line interface (CLI).
   - Invoke the `pin_trial_bill_accts` script to create trial bill objects for specified accounts.
   - Trial bills include cycle fees and total amounts but do not alter existing account balances or General Ledger (GL) feeds.
2. **Trial Invoice Generation**
   - Run the `pin_trial_inv_accts` script to generate trial invoice files in XML format.
   - XML invoice files are generated for review and are not persisted in the database.
3. **Validation and Output**
   - The XML files can be further processed manually using the Oracle Analytics Publisher (OAP) system for invoice render and format validation.

Limitations

- Trial Billing is not intended for comprehensive billing simulation following major migrations or as a replacement for B2B proforma or pre-billing scenarios.
- Trial invoice output is for validation purposes only and does not trigger any operational or financial processing, no notifications, no queuing, no Document Management System (DMS) archiving, no tax commit.

Exception management

In case of processing failures for specific accounts, operators must re-run the utilities for failed account list; investigate configuration/data issues surfaced in logs.

About Correct Bill Invoice

The Correct Bill Invoice feature supports the issuance of credit notes to address adjustments on previously issued bills or bill items. This capability ensures compliance with legal and tax accounting requirements for Business-to-Business (B2B) and Business-to-Customer (B2C) customers by providing transparent and auditable corrective billing statements.

This process is triggered following the resolution of a bill inquiry, resulting in an approved adjustment on a prior bill.

A Customer Service Representative (CSR) initiates the corrective billing action by specifying the adjustment in the Siebel CRM, which triggers credit note generation within the billing system. The request is submitted to Billing and references the original bill/item adjustment.

This feature:

- Facilitates the generation of credit notes or corrective invoices resulting from bill inquiries and adjustments.
- Maintains alignment with legal, regulatory, and customer tax accounting standards, including General Ledger (GL) postings and adjustment.

This process involves the following steps:

1. **Credit Note Generation**
   - The system creates a new bill object for the affected billing account, using a predefined prefix series (for example, CN-XXXXX).
   - The credit note documents the amount credited or debited with explicit reference to the original bill number.
2. **Invoicing and Formatting**
   - The system applies a dedicated invoice template for credit notes to ensure clear differentiation from standard invoices.
   - Credit note is displayed as the heading on the invoice.
3. **Archiving and Distribution**
   - Invoicing, formatting, archiving, and distribution processes are invoked asynchronously after the credit note is generated, following standard operational procedures.
4. **Bill Now Execution**
   - After the item adjustment, the Bill Now process is used to immediately generate and issue the credit note, ensuring the update is reflected without waiting for the next regular billing cycle.

> **Note**
>
> - Only authorized and approved adjustments can trigger credit note issuance.
> - Each credit note references the original bill and carries a transaction/reference id for reconciliation.
> - All corrective actions are subject to existing audit trails and legal accounting requirements.

About Upload Invoice to DMS System

The Upload Invoice to DMS System feature manages the secure upload, archiving, and controlled retrieval of finalized invoice documents. This process ensures that all rendered invoices are stored in a Document Management System (DMS), supporting both compliance and operational access requirements across billing and customer service channels.

The upload process starts automatically after successful completion of the Generate Bill Invoice operation.

This feature:

- Provides secure, long-term storage for finalized invoices.
- Enables direct invoice access from Siebel CRM and self-care (unassisted) channels.
- Ensures systematic invoice indexing and archiving for regulatory compliance and operational needs.

This process involves the following steps:

1. **Event Creation**
   - Upon invoice rendering completion, an event containing bill metadata for content indexing is generated by the system.
   - This event is published into a queuing system for downstream processing.
2. **Acquisition and Upload**
   - An integration process (using AIA or OSB) consumes the event, retrieves the invoice from the Oracle Analytics Publisher (OAP) system, and uploads it to the designated DMS repository.
3. **Archiving and Access Control**
   - The DMS applies security and archiving policies.
   - Uploaded invoices are indexed for direct access via customer service applications and unassisted self-care portals.

> **Note**
>
> - DMS configuration is a field implementation and needs to be configured to upload invoices.
> - Ensure that integration endpoints between OAP, queuing, and DMS are monitored for reliability.
> - Verify that bill metadata content is complete and accurate for content indexing to enable effective search and retrieval.

Exception management

- In the event of upload failures or process interruptions, the system supports resumption of the upload process.
- Logic is in place to avoid gaps or duplicate entries during recovery or retries.

About Configure Billing

The Configure Billing feature allows Oracle Communications Billing and Revenue Management (BRM) specialists to define and manage key billing parameters and behaviors to meet specific business requirements. Configuration occurs during the design phase and is a prerequisite for achieving the expected operational outcomes from the billing system.

This feature:

- Establishes system-wide billing behavior aligned with organizational policies and business goals.
- Enables flexible control over billing cycles, accounting methods, customer segmentation, and integration points, including corrective billing and invoicing.

The system supports configuration of the following global parameters and extensions for billing:

- Primary Currency: Define the operational currency for all billing processes.
- Delayed Billing: Set rules for deferred bill runs or invoice generation.
- Open Item Accounting: Enable or disable open item accounting practices within the system.
- Item Assignment: Control itemization and groupings of charges.
- Customer Segment (Bill Suppression): Configure rules for bill suppression based on customer segment and minimal threshold amounts.
- Billing Segments: Organize billing cycles and groupings by segment.
- Journal and General Ledger (GL): Integrate billing events with accounting systems for revenue posting and reconciliation, and maintain GL Segments, GL Codes, and GLIDs.
- Rollover: Manage rollover logic for usage, balances, or charges across billing periods.
- Rating Engine and Pricing Designer: Specify engines for usage charging and pricing rule application.
- Tax Configuration: Set up tax rules, codes, and calculation parameters for compliance.
- Bill Number and Corrective Bill Prefix: Define numbering schemes for regular and corrective bills (for example, CN-xxxxx for Credit Notes and DN-xxxx for Debit Notes).
- Billing Event Notifications: Configure systems for notifying stakeholders of billing events and statuses.
- Invoice Template Configuration: Manage Oracle Analytics Publisher (OAP) templates (Business-to-Customer (B2C), Business-to-Business (B2B), Bill Now, Threshold, Credit Note, and Re-billing).
- Collections Configuration: Define profiles, entry/exit thresholds, grace days, and action scenarios.

Out-of-the-Box (OOTB) Validations

- The default system configuration supports Bill Day of Month (BDOM) values from 1 to 28.
- For customers configured with BDOM as 29, 30, or 31, the billing system automatically assigns a billing date of the 1st of the following month.

> **Note**
>
> - All configuration settings should be reviewed and managed by designated BRM specialists.
> - Validate global parameter changes in a controlled environment prior to production deployment.
> - Ensure all business and compliance requirements are captured in billing configuration before go-live.

About Payments and Receivables Process

This process governs the management of subscriber payments, application of those payments to outstanding invoices, and the tracking of receivable balances throughout the billing lifecycle.

This process ensures the secure and accurate collection of amounts due from subscribers through various payment channels, including electronic, manual, and third-party gateways.

The system records each payment transaction, automatically reconciles received funds with open invoices, and updates the subscriber's account status accordingly. It also monitors one-time payments (must carry distinct General Ledger IDs (GLIDs)), receivable balances, manages overpayments and underpayments, and supports adjustments, reversals, and chargebacks as required for both operational and regulatory compliance.

Additionally, this process enables timely cash application, supports financial reporting obligations, and provides the foundation for collections management by identifying overdue accounts and escalating outstanding amounts in accordance with organizational policies and market regulations. This process plays a critical role in maintaining the accuracy of financial records and ensuring the ongoing integrity of the revenue cycle with General Ledger (GL) segment/code/GLID mappings maintained in PDC/BRM for Accounts Receivable (A/R) and tax postings; adjustment reasons use LOVs mapped to GLIDs and apply proportional tax reversal when itemized tax is enabled.

Placing an Adjustment

This feature enables Customer Service Representatives (CSRs) and back-end support teams to process financial corrections to bills or bill items resulting from customer inquiries or disputes. The process upholds rigorous approval controls, ensures tax integrity, and maintains detailed audit trails in accordance with financial and regulatory requirements.

Adjustments are initiated following the resolution or approval of a bill inquiry or dispute through the ticketing system.

Only authorized personnel, such as CSRs or billing support staff, may process adjustments using the Siebel CRM UI or Billing Care UI.

The adjustment process works as follows:

- The user selects the billed item(s) requiring adjustment.
- The amount to adjust is specified in the transaction currency and must include all applicable taxes.
- A justification is required for each adjustment, selected from a predefined list of values (LOV).
- The system references configured GLID tracks for financial tracking.

Depending on the account and situation, adjustment posting can lead to the following:

- Application to next bill: The adjustment amount is carried forward and reflected on the subscriber's subsequent bill by default.
- Credit note generation: In cases requiring formal correction, a credit note is issued in accordance with credit note processes.
- Refund issuance: If the adjustment results in a negative subscriber balance and the contract is terminated, a refund process will be initiated.

Financial and Tax Handling

- The system automatically applies proportional tax reversals based on the adjusted itemization, ensuring tax reporting accuracy.
- For Business-to-Business (B2B) customers, adjustments are journalized, capturing all tax credit impacts and associating the reason (LOV) with the corresponding general ledger (GL) record for further financial analysis.

> **Note**
>
> - Currently, GLIDs are supported only for B2B customers.
> - All adjustments are recorded with reference to their reason code, approval chain, and financial impact.
> - Adjustment activities are available for finance teams to review and analyze through the configured GL identifiers.

About General Ledger Requirements

This feature ensures that all financial transactions impacting the organization's position are accurately reflected in the general ledger (GL). This capability enables consistent financial tracking, auditing, and reporting by associating every relevant transaction, such as charges, payments, bills, adjustments, and asset changes with dedicated GL codes and segments.

This feature:

- Guarantees complete and auditable representation of company financial activities in the GL.
- Supports detailed financial analysis for each product, service, or business activity.
- Complies with finance department and regulatory reporting requirements.

Configuration and Setup

- GL segments: Segments, such as Consumer and Business, are defined in a configuration file before upload to the Oracle Communications Billing and Revenue Management (BRM) system.
- GL codes:
  - Assigned based on the Communications Service Provider's (CSPs) chart of accounts.
  - Each transaction type (usage, payment, bill, adjustment, purchase, asset change) is mapped to appropriate GL codes.
- GL IDs: Unique identifiers are assigned for each Accounts Receivable (A/R) transaction and uploaded to the BRM system.
- Code association:
  - Each product, package, service, or discount is associated with the relevant GLID codes to ensure correct reporting.
  - Code-to-product associations and percentages for bundled offerings are defined and configured in the Oracle Communications Pricing Design Center (PDC) during product creation or amendment.

Financial Posting Process

- All transactions affecting the company’s balances are allocated to the appropriate GL segment and code at the time of event/rating, billing, payment, or adjustment.
- These allocations accumulate as transaction activity occurs and are periodically posted or exported to the GL system for consolidated financial reporting.

Usage and Audit

- Finance administrators and reporting teams can access detailed breakdowns of all GL-mapped transactions, including charges, credits, adjustments, and bills.

- This traceability supports the accurate reflection of financial performance for each service, product, or business activity in compliance with accounting standards.

> **Note**
>
> - Currently, this feature is applicable only for the Business-to-Business (B2B) customer segment.
> - Ensure all required GL codes and segments are loaded into the system before product or transaction configuration.
> - Maintain alignment between the chart of accounts and GL mappings in the billing platform.
> - Update GL assignments promptly when creating or modifying products, packages, or discounts in PDC.

About One-Time Payments

This feature provides the capability to process individual, unprompted subscriber payments from various internal and external sources. This includes payments for initial orders, settlements of open bills, and non-automated recurring payments, ensuring flexible and accurate revenue collection through multiple payment channels (assisted via Siebel; unassisted via digital; manual reconciliation for bank transfers).

The following are the supported payment scenarios:

- Acceptance of payments for single or multiple open bills.
- Handling of payments not set up as automatic or recurring debits.
- Processing of payments for both initial orders and outstanding invoices.

Payments may originate from the following channels and payment methods:

- Assisted channels (Siebel UI/Call Center/POS):
  - Credit card payments
  - Cash payments at point of sale (POS)
  - Manual check processing
  - Wire transfer reconciliations completed by back office staff
- Unassisted digital channels:
  - Online credit card processing
  - Third-party payment services integration
- Bank wire transfer:
  - Bank-to-bank account transfers, recorded and reconciled subsequently by finance personnel using Billing Care UI or Siebel CRM UI.

This process involves the following steps:

1. Payment details submission: Required details (payment type, bill reference, billing account, amount, payer name, date, method-specific data, such as check number) are submitted to the Oracle Communications Billing and Revenue Management (BRM) system.

2. Accounting requirements: Each payment method or configuration is associated with a distinct General Ledger Identifier (GLID), to ensure correct financial posting and reconciliation.
3. Payment allocation: BRM allocates the payment to the associated bill(s) or billing account according to business rules and configuration.
4. Customer notification: BRM generates and dispatches payment confirmation notifications to the subscriber upon successful processing, in accordance with notification guidelines.

The following are the integration interfaces available:

- Online processing: Credit card transactions and interactions with clearing houses.
- Manual reconciliation: Bank file uploads and wire transfer records managed by finance support teams.

> **Note**
>
> - Ensure all payment sources and types are mapped to the designated GLIDs for accurate accounting.
> - Maintain secure handling of sensitive payment data, especially for credit card transactions.
> - Coordinate with finance and customer support teams for effective management of manual and assisted channel payments.
> - Payment events should update collections status where applicable (for example, exit criteria and unbar triggers).

About Charging and Call Control

The Charging and Call Control process is essential for managing real-time rating, authorization, and control of customer telecommunications services. This process coordinates the assessment and enforcement of subscriber account balances and service entitlements at the time of service invocation, ensuring accurate charging and preventing unauthorized or over-quota usage.

During service usage, such as voice calls, data sessions, or value-added interactions, the process interacts with network and charging systems to verify subscriber balances, determine applicable tariffs, and authorize the requested usage based on business policies, product definitions, and usage quotas. For prepaid scenarios, the system performs real-time deduction from subscriber balances, while in postpaid environments, it accumulates charges for later billing by configuring service tariff plans in PDC and applied per Rating Group (RG).

The process is tightly integrated with policy management, rating engines, and network elements to enforce thresholds, apply discounts, and perform call barring or session termination when limits are reached. Comprehensive event logging and audit capabilities are maintained for revenue assurance and regulatory compliance. Ultimately, this process provides the foundation for reliable, secure, and transparent monetization of telecommunications services, supporting both customer experience and operator revenue objectives.

Performing Online Rating

This feature enables the real-time calculation and application of charges for subscriber service usage events. This feature determines the usage quantity, relevant zone, and correct rate

based on the subscriber's provisioned product and tariff plan, then applies the calculated charges to the appropriate account balances. The process supports both event-based and cumulative session-based rating scenarios, ensuring accurate and compliant monetization of telecommunications services.

Prerequisites:

- Subscriber is provisioned with an offer that includes an associated tariff plan.
- Zone definitions and service tariff plans are configured in the Oracle Communications Pricing Design Center (PDC).

The rating workflow involves the following processes:

- Subscriber identification
  - The subscriber is identified in the system.
  - The system receives a service usage request for rating via the defined network/ charging interface.
- Event-based rating
  - The system rates quantities grouped by Rating Group (RG) according to the service rate plan.
  - If applicable, the relevant zoning group for the service event is identified.
  - Both currency and non-currency balances are updated and discounts are applied.
  - Real-time taxation and general ledger (GL) impacts may be triggered, based on configuration.
  - The system generates and persists a rated event.
- Session-based rating
  - Update request
    - Receives Used-Service-Units (USU) with RG for accumulation in an ongoing charging session.
    - If sufficient balance exists, a new grant (GSU) is reserved and deducted. If not, and credit limits are enforced, the request is rejected and service redirection may occur.
    - Upon specific mid-trigger events, aggregated usage is attached to a bill item and persisted.
  - Terminate request
    - Upon session termination, final USUs and RG quantities are rated and charges are applied according to the subscriber’s rate plan.
    - Taxation and GL impacts are determined as configured.
    - The system persists a final rated event and closes the charging session.

Applying Usage Discounts

This feature enables the automated application of usage-based discounts during the final stage of the service rating process. This feature ensures that eligible subscribers receive contractually defined rebates, such as recurring or one-time allowances, directly impacting their rated usage charges and account balances.

Prerequisites

- The subscriber has purchased an offer that includes a cyclic (recurring) or one-time (top-up) usage allowance.
- The usage event being rated is eligible for discounting according to the terms of the subscriber's offer.
- There is a non-zero allowance balance available at the time of rating.

The discount application logic is as follows:

- Upon completion of the initial rating, the system checks for eligible usage units that qualify for allowance discounts.
- The Oracle Communications Elastic Charging Engine (ECE) evaluates the Convergent Usage Discount Filter to determine applicable discount rules.
- If conditions are met:
  - The monetary charge (in currency units) for eligible usage is partially or fully offset by the available allowance.
  - The corresponding allowance balance is reduced by the discounted usage units.
  - A detailed record of the discount application, including the allowance consumption and resulting balance, is recorded within the event data for audit and reference.

> **Note**
>
> - The discounting process is performed as the concluding step in the rating flow to accurately reflect any usage-based rebates.
> - If multiple discounts exist, their precedence/order is applied as per the configuration.
> - If the relevant bill item does not exist, the system auto-creates it so that the discounted event can be attached.
> - All adjustments to balances and event records are performed consistently to maintain financial integrity and support downstream billing and reporting activities.

Applying Events to Bill Items

This feature enables the association of rated service event records with appropriate bill items for each subscriber. This process ensures that all chargeable usage events are accurately reflected in the subscriber's invoice, supporting transparent and itemized billing.

During the rating process, the Online Charging System (OCS) generates and persists service event records according to predefined configuration rules.

Each event record contains usage-specific details required for billing and compliance.

Association with Bill Items

- Every rated event is linked to an existing bill item on the subscriber's account to enable invoicing.
- The system selects the applicable bill item based on configuration rules, ensuring alignment with the expected invoice structure.
- If a suitable bill item does not exist, the Oracle Communications Elastic Charging Engine (ECE) automatically creates a new bill item for the event.

Service event records capture the following usage attributes:

- Start and end time (local timestamp)
- Origin zoning information (for example, MCCMNC) to identify roaming activity
- B-Number (called number)
- Zoning destination for Voice or SMS
- User Equipment (UE) IP address (PDN)
- Serving Gateway (SGW), Mobility Management Entity (MME), Roaming Indicators, or Access and Mobility Management Function (AMF) address

> **Note**
>
> - Event-to-bill item assignment must be carefully configured to ensure the invoice structure remains accurate and compliant.
> - Automated creation of bill items streamlines billing operations and supports complete, auditable invoice records.

About Manage Collections Process

This process encompasses the systematic identification, monitoring, and resolution of overdue receivables resulting from unpaid subscriber invoices. This process begins when a subscriber's account becomes delinquent (meets entry criteria for its collection profile/scenario after grace days) and continues through a series of structured actions designed to recover outstanding balances while maintaining compliance with internal policies and external regulations.

Throughout the collections lifecycle, the system tracks aging receivables, evaluates account risk, and prioritizes actions based on configurable criteria, such as amount overdue, customer segment, and historical payment behavior. Automated notifications, reminders, and escalation procedures are generated to inform subscribers of their obligations and prompt timely payment. If necessary, additional measures, such as dunning, temporary service suspension, or referral to external collection agencies may be initiated in accordance with company policy.

By providing comprehensive oversight and control of collection activities, the Manage Collections process supports revenue assurance objectives, reduces bad debt exposure, and upholds the financial integrity of the organization with the following entry/exit rules:

- Entry: Move an account into collections when due amount exceeds the scenario's entry threshold and grace days have passed.
- Execution: On each action due date, if no or insufficient payment is received, Siebel executes automated steps (for example, send text, barring order) and posts status back to Oracle Communications Billing and Revenue Management (BRM).
- Exit: Upon sufficient payment or a valid Promise-to-Pay (PTP), cancel pending actions, unbar/restore services to pre-collections state, and update Siebel to remove workflows.
- Aging and reporting: Maintain aging buckets.

Qualifying Invoices for Collections

This feature automates the identification and initiation of collections treatment for accounts with overdue balances. This process is scheduled to run periodically, typically on a daily basis, following payment and billing cycles. It ensures timely escalation of delinquent accounts,

enabling effective risk management and minimization of outstanding balances, debtor days, and bad debt exposure.

This feature:

- Automatically detects accounts with overdue amounts exceeding thresholds.
- Initiates structured collections scenarios to recover outstanding balances.
- Supports dynamic assignment and update of collection profiles based on account payment behavior.

Accounts are assigned a collections profile upon creation, reflecting risk and payment history. The profile is adjustable over time (for example, relaxed criteria after sustained good payment behavior).

Each profile is linked to a collection path, specifying treatment steps and timing (for example, send SMS, initiate outbound call, restrict service).

Accounts are evaluated for collections entry using the following logic:

- The collections profile class for the billing account is valid.
- There is an associated collection scenario (treatment plan) for the profile.
- The account is not currently in collection status.
- The account's bill due amount exceeds the collection scenario's entry threshold.
- The designated grace period has elapsed (considering weekends/holidays as well).

Upon meeting the above criteria:

- The account's status is updated to **In collections**.
- The linked collection scenario is assigned.
- All predefined collections actions in the scenario are created and scheduled for implementation.
- Collections actions are synchronized with Siebel for further processing.

If an account is already in collection status:

- The process evaluates whether the overdue amount remains above the exit criteria value.
- If the overdue amount exceeds the threshold, the account remains in collections status without exiting the collections cycle, else exit processing is handled by the Exit Collections flow.

Implementing Collections B2B

This feature manages and automates the collections process for Business-to-Business (B2B) accounts identified as delinquent. Each account is associated with a defined collections path, which sequences the treatment steps and the intervals between them. The feature ensures prompt and systematic handling of overdue receivables, escalation of collection actions, and alignment between the Siebel and Oracle Communications Billing and Revenue Management (BRM) systems.

Each delinquent B2B account is linked to a specific collections path that prescribes:

- The sequence of actions to be performed.
- The duration between each action step.

Scheduling and Implementing an Action

> **Note**
>
> On each due date, actions are implemented only if no or insufficient payment has been received and exit criteria is not met.

- Actions are triggered on scheduled dates, calculated from the date the bill unit enters collections status.
- Example of an action schedule:
  - Action 1: Day 2
  - Action 2: Day 5
  - Action 3: Day 10
- Scheduled actions are mapped to calendar dates. For example, if a bill enters collections on May 1:
  - Action 1: May 3
  - Action 2: May 6
  - Action 3: May 11

Commonly implemented collections actions include:

- Sending reminder texts/emails
- Sending reminder letters
- Initiating outbound calls
- Barring or throttling services
- Terminating agreements and cancelling services
- Writing off debt (manual finance action)
- Sending debt to external collections agencies
- Notifying the legal department

The process flow is as follows:

- On the scheduled date, if no payment or an insufficient payment has been received:
  - Siebel automatically implements the next required action (for example, sending reminders, barring services).
  - For actions requiring manual involvement (such as outbound calling or management interventions), Siebel initiates the appropriate workflow.
  - Automated processes are triggered where possible, reducing manual effort for repetitive actions.
- Each action is recorded, and completion is communicated back to BRM for consolidated collections management and audit purposes.
- When sufficient payment or a valid Promise-to-Pay is recorded, remaining actions are cancelled and unbar/restore activities are initiated; credit profile and credit limit are reassessed as per the exit process.

> **Note**
>
> - Collections actions are synchronized from BRM to Siebel.
> - Siebel is responsible for implementing or initiating all required activities in accordance with the collections path.
> - Status and completion updates are sent back to BRM to maintain system alignment and data integrity.

Configuring Collections B2B

This feature defines the run-time parameters and workflow settings required to support automated and manual collections management for Business-to-Business (B2B) subscriber accounts. These configurations establish the credit risk framework, treatment paths, and operational processes necessary for effective and compliant handling of overdue receivables in the enterprise segment.

Credit limits are assigned to business accounts based on business rules, such as account tenure, product value, and payment history. For example:

- New business accounts with products valued at $500: credit limit of $1,000
- Accounts older than one year with good payment history: credit limit of $1,500
- Accounts averaging $750 per month: credit limit of $1,500

Collections paths are defined that detail the sequence and timing of actions performed during the collections process. For example:

- Path C1:
  - Action 1: Day 2
  - Action 2: Day 5
  - Action 3: Day 10
- Path C2:
  - Action 1: Day 2
  - Action 4: Day 3
  - Action 3: Day 15

> **Note**
>
> On each due date, actions are implemented only if no or insufficient payment has been received and exit criteria (including Promise-to-Pay) is not met.

Actions are configured that represent operational steps taken in each path. For example:

- Action 1: SMS notification/email (using format 'format1')
- Action 2: Letter dispatch
- Action 3: Barring order request to Siebel
- Action 4: Configurable additional actions as required

Credit profiles are set, which segment subscribers to guide risk and collection approaches. For example:

- Profile 1: New subscriber, value below threshold, no deposit required
- Profile 2: New subscriber, value above threshold, deposit required
- Profile 3: Established subscriber (more than 1 year), value below threshold, direct debit payer

Aging buckets are defined that categorize overdue balances based on the number of debtor days to support risk monitoring and reporting. Standard groups are as follows:

- 0 to 15 days
- 16 to 30 days
- 31 to 50 days
- Over 50 days

Implementing Collections B2C

This feature automates the collections lifecycle for Business-to-Customer (B2C) subscriber accounts identified as delinquent due to unpaid postpaid bills. The system applies standardized collections treatment paths tailored to customer segment, outstanding balance, product type, and delinquency age. Automation ensures efficient recovery, minimizes bad debt exposure, and maintains a consistent customer experience with entry and exit criteria enforced as per the configuration.

Each delinquent B2C subscriber account is automatically assigned a collections treatment path based on:

- Customer risk segment (low, medium, or high risk)
- Outstanding debt amount
- Days past due (DPD)/grace days elapsed from bill due
- Product type (for example, mobile, broadband, DTH)

Each treatment path specifies:

- The sequence of actions
- The interval between actions
- Escalation logic for non-payment (advance to next step only when no or insufficient payment and no valid Promise-to-Pay (PTP))

The following table provides example action schedules for collection treatment.

Table 4-1 Example Action Schedule

Action	Day from Bill Due
SMS reminder/emails	Day 2
IVR auto call	Day 5
Outgoing call center attempt	Day 8
Outgoing service restriction	Day 12
Incoming + Outgoing restriction	Day 18
Permanent disconnection	Day 30

Typical collection actions are as follows:

- Automated SMS/email payment reminders
- IVR auto-dial reminders
- Agent-assisted outbound calls
- Outgoing call/SMS bar
- Full service suspension
- Permanent account disconnection
- Write-off for low-value debt
- Transfer to external recovery agency
- Blacklist/KYC tagging
- Service reactivation upon payment

The workflow of this feature is as follows:

- Treatment actions for delinquent subscribers are synchronized from Oracle Communications Billing and Revenue Management (BRM) to Siebel.
- Payment status is updated in near real-time through digital channels (cards, apps) and via batch for offline/bank transfers.
- On reaching the scheduled action date, if payment remains insufficient or absent:
  - Siebel implements the required collection step.
  - For manual steps, Siebel triggers appropriate workflows (for example, outbound calls, management escalation).
  - Automated actions (for example, SMS, service barring) are system-initiated.
- Siebel communicates step completion and status updates back to BRM.

Upon receipt of full payment:

- All service restrictions (bars) are removed.
- Any future scheduled collections actions are cancelled.
- Full service is restored within the service-level agreement (SLA).
- Subscriber credit behavior profile is updated.

The integration for this feature happens as follows:

- BRM to Siebel: Synchronization of treatment actions.
- Siebel to BRM: Reporting of action status and step completion.
- Siebel to OSM: Fulfillment of service orders (bar/unbar/disconnect).
- Siebel to SMSC: Dispatch of messaging notifications (for example, SMS reminders).

Configuring Collections B2C

This feature establishes the parameter-driven controls and workflow definitions necessary to support automated and manual collection activities for Business-to-Customer (B2C) customers. This configuration enables operators to efficiently manage overdue accounts, enforce credit policies, and tailor collections strategies to subscriber profiles and payment behaviors.

Credit limits are assigned to subscriber accounts based on predefined rules reflecting account tenure, product value, and payment history. For example:

- New subscriber accounts with products valued at $100: credit limit of $200
- Accounts older than one year with good payment history: credit limit of $300
- Accounts with average monthly value of $150+: credit limit of $300

Collections paths are defined, which specify the sequence and timing of actions based on delinquency duration. For example:

- Path C1
  - Action 1: Day 2
  - Action 2: Day 5
  - Action 3: Day 10
- Path C2
  - Action 1: Day 2
  - Action 4: Day 3
  - Action 3: Day 15

> **Note**
>
> On each due date, actions are implemented only if no or insufficient payment has been received and exit criteria (including Promise-to-Pay) is not met.

Actions within each collection path are explicitly defined. For example:

- Action 1: SMS notification/email (using format 'format1')
- Action 2: Letter dispatch
- Action 3: Initiation of barring order request to Siebel
- Action 4: Configurable additional actions as required
- Termination: Generate customer notice; submit cancel/tear-down order

Credit Profiles are set that enable tailored collections and credit strategies based on subscriber status and payment characteristics. For example:

- Profile 1: New subscriber, value below threshold, no deposit required
- Profile 2: New subscriber, value above threshold, deposit taken
- Profile 3: Established subscriber (more than 1 year), value below threshold, pays via direct debit

Aging buckets are defined that segment debtor balances by the length of time past due for reporting and prioritization purposes. For example:

- 0 to 15 days
- 16 to 30 days
- 31 to 50 days
- Over 50 days

> **Note**
>
> - Maintain a systematic approach to defining and updating rules, actions, and profiles to align collections activities with evolving business policies.
> - Ensure collections configurations are consistent with regulatory and customer experience requirements.

About Analysis of Siebel Collections Processing

This feature details the Siebel system's role in performing collection activities initiated from the Oracle Communications Billing and Revenue Management (BRM) application. This integration ensures comprehensive handling of overdue accounts, customer communication, service management, and escalation actions throughout the collections lifecycle.

The following actions are triggered in Siebel as part of the collections process, either via synchronized requests from BRM or manual initiation:

- **Email or SMS Notification**:
  - Siebel sends an email or SMS reminder to subscribers regarding outstanding bills.
  - Example Email Content: This is a reminder that your communications bill of *$nn* on account *nnnnnn* is now due. Please pay as soon as possible to avoid service disruption. If you have already paid, please ignore this message.
  - Failed or bounced messages are flagged on the collections event.
  - All communication details are stored and accessible via the Customer 360 interface.
- **Physical Mail Generation**:
  - Siebel generates a formal letter for mailing using the Siebel document server or Oracle Analytics Publisher (OAP) system, as appropriate. The letter includes:
    - Account number
    - Debt amount
    - Due date
    - Payment instructions
    - Notice regarding consequences of non-payment and available support channels
  - All letters are stored for retrieval and displayed in Customer 360.
- **Call Initiation**:
  - Siebel schedules or triggers an outbound call, delivered through a Customer Service Representative (CSR) or automated recording.
  - Call metadata, recordings, and summaries are stored for CSR review within Customer 360.
- **Bar or Throttle Service**:
  - When sufficient payment or a valid Promise-to-Pay is recorded, Siebel generates the unbar/restore Modify order and submits it to OSM.
  - Upon a bar or throttle action, Siebel generates and submits orders to restrict service (outbound, total, or bandwidth throttling) for all products on the corresponding billing account.

  - Orders are automatically processed for provisioning via OSM.
- **Terminate Agreement and Cancel Services**:
  - Siebel produces and sends a termination notice to the subscriber (detailing account, debt, date, and affected services).
  - Generates orders to cancel (tear down) all services, quarantining resources for reuse.
  - Flags account as **Terminated for debt** and adds it to the bad debt register. This status syncs back to BRM.
- **Debt Write-off**: Manual action performed by finance personnel using Siebel or Billing Care, referencing the appropriate policy.
- **Send to Collections Agency**: Activity is either processed by internal collection agents or forwarded to finance for external collection action.
- **Legal Escalation**:
  - A service request or activity is generated for the legal department.
  - Account is flagged as **Legal**, ensuring any subscriber contact is routed appropriately.

> **Note**
>
> - All collections activities, communications, and escalations are recorded for retrieval and review within Customer 360.
> - SMS, Call, and Email are field configurations and not part of the reference solution. Hence, sending messages is subject to integration with underlying SMSC and email server.
> - Status changes and service actions are synchronized between Siebel and BRM to maintain data consistency and auditability.
> - Execution failures are not advanced unless policy allows.

About Collections Siebel Sends Message

This feature automates the process of notifying subscribers regarding overdue bills through electronic and physical communication channels. This feature leverages Siebel to implement message delivery actions triggered by the collections workflow synchronized from the Oracle Communications Billing and Revenue Management (BRM) system.

Siebel performs the following actions as part of the collections process:

- Send reminder text (Email/SMS)
- Send reminder letter (Physical Mail)

Electronic Messaging

> **Note**
>
> SMSC and Email are field configurations and not part of the reference solution. Hence, sending messages is subject to integration with underlying SMSC and email server.

- Siebel sends a reminder message to the customer via email or SMS.

- The message communicates the outstanding bill amount and account reference, for example:
  This is a reminder that your communications bill of *$nn* on account *nnnnnn* is now due. Please pay as soon as possible to avoid service disruption. If you have already paid, please ignore this message.
- If a message fails to deliver or is returned as undelivered (for example, incorrect email address), Siebel flags the event on the collections record for further review.
- All message details are recorded and made available for display and historical reference via the Customer 360 view.

> **Note**
>
> - Ensure message templates, addresses, and configurations are maintained for accuracy and compliance.
> - Regularly review flagged undelivered messages and update contact information as required.

About Collections Siebel Creates SR or Activity

This feature manages the initiation of manual activities and service requests (SRs) as part of the collections process. These actions are triggered by collections events synchronized from Oracle Communications Billing and Revenue Management (BRM) and are implemented within Siebel to ensure proper escalation and tracking of non-automated collections steps.

Siebel supports the following manual actions within the collections workflow:

- **Outbound call**
  - Siebel initiates an outbound call activity, which can be conducted by a Customer Service Representative (CSR) or as an automated recorded message. Both outbound calls and automated recorded messages are handled via approved dialer/IVR integrations as configured.
  - Each outbound call activity is assigned to a designated user or position in the system.
  - Details captured include the date and time of the call, call recording, and call summary.
  - All call activity data is stored and available for review within the Customer 360 view.
- **Write-off debt**
  - Write-off is a manual Accounts Receivables (A/R) adjustment performed by a finance department employee.
  - This action can be completed via Siebel or Billing Care.
  - The task is assigned to a finance team member or role responsible for debt adjustments.
- **Notify Legal department**
  - Siebel creates an SR or activity directed to the Legal department for accounts requiring escalation.
  - The account is flagged as **Legal**, ensuring that any future customer contact is directed appropriately.
  - The SR or activity is assigned to the Legal department or appropriate user/position for follow-up.

> **Note**
>
> - SR or Activities are field configurations and not part of the reference solution. Hence, sending messages or SR is subject to integration with underlying required systems.
> - Each manual activity or SR is explicitly assigned an owner (user or functional role) for clear responsibility and follow-through.
> - All activities and their status are tracked within Siebel and visible for operational oversight.

About Collections Automated Disconnect, Barring, and Unbarring

This feature orchestrates the execution of service restrictions and restorations in response to collections actions initiated in the Oracle Communications Billing and Revenue Management (BRM) system. Integration with Siebel ensures that all necessary provisioning and customer communication actions are automated, consistent, and auditable.

Bar, Unbar, or Throttle Service

- Upon receipt of a collections action from BRM, Siebel automatically generates an order to apply the designated service restriction to all or selected services/products linked to the billing account. Actions may include:
  - Outgoing bar (restrict outbound service)
  - Total service bar (restrict both inbound and outbound)
  - Bandwidth throttle (for applicable products)
- The order is automatically submitted for provisioning through the Oracle Communications Order and Service Management (OSM).

> **Note**
>
> Credit alert based throttling initiation from Siebel is considered as field configuration (based on Policy Control Function (PCF) or Policy and Charging Rules Function (PCRF) integration) and not part of the reference solution.

Integration and Synchronization

- All collections-generated service actions initiated in BRM are communicated to Siebel for fulfillment and status update.
- Siebel automates order generation, submission, and customer notification processes (for example, SMS/email notices for barring/termination as required), ensuring alignment with collections policies and provisioning requirements.
- Updates to account statuses and service orders are reflected back to BRM, maintaining consistent financial and operational records.

> **Note**
>
> - Ensure all actions and status changes are logged for compliance and auditability.
> - Maintain clear communication protocols between BRM, Siebel, and OSM to prevent service recovery or restriction errors.

About Exit Collections

This feature manages the process of removing subscriber accounts from collections treatment when designated exit conditions are met. Its primary objective is to allow timely restoration of services and update account credit status, thereby balancing revenue recovery with a reduction in bad debt exposure.

This feature:

- Facilitates the rapid reinstatement of services for subscribers who have resolved their outstanding balances or made acceptable payment arrangements.
- Ensures accurate updating of credit limits, collections profiles, and service statuses in accordance with company policy.

The Exit criteria is as follows:

- Full Payment: The account is eligible for collections exit when full payment of outstanding debt is received. Exit criteria can be configured to accept partial payments above a defined limit/threshold.
- Promise-to-Pay (PtP) Arrangement: Accounts with actively maintained PtP agreements may also qualify for collections exit, depending on configuration.

Depending on the mode of payment, the account exits collections as follows:

- **Online Payment**
  - When an online payment meeting or exceeding the exit threshold is received, the account is immediately removed from collections status.
  - No further collections actions or treatments are scheduled.
  - The account’s collections profile and credit limit are reassessed and may be updated.
  - Any service restrictions imposed by collections activities are lifted, and service is restored to its pre-collections state (including any previously active voluntary suspensions).
  - Siebel is updated to cancel associated collections workflows and actions.
- **Offline (Batch) Payment**
  - When a batch-processed offline payment meets the exit threshold, the collections exit execution process identifies eligible accounts.
  - The account is taken out of collections status, and no further treatment actions are initiated.
  - The account’s collections profile and credit limit are reassessed and updated as needed.
  - Service restrictions are addressed, with restoration to the prior status before collections actions.
  - Siebel is correspondingly updated to remove collections workflows or actions.

This feature uses Oracle Communications Billing and Revenue Management (BRM) to Siebel interfaces to ensure synchronization for cancellation of collections actions and restoration of account status (weekends/holidays are considered).

A

Mapping Siebel Billing Management UI Elements to BRM Customer Center

This appendix provides a mapping of fields on the Siebel customer relationship management (Siebel CRM) Billing Management UI to fields in Oracle Communications Billing and Revenue Management (BRM) Customer Center. This appendix can be used as a reference to explain the Cash to Care integration.

Billing Profile and Account Balance

Navigate to the **Account Summary, Billing Profile** applet, **Billing Profile Name** link.

Siebel Screen: Billing Profile Portal screen

Siebel View: Billing Invoice

UI Component: Billing Profile, as described in Table A-1.

Table A-1 Billing Profile Mapping

Siebel Applet	Siebel Field	BRM Tab	BRM Form	BRM Field	Comments
Billing Profile Form	Account	Summary	Contact Information	Company or First & Last Name	Account/Customer name
--	Primary	NA	NA	NA	--
--	Profile Name	Payments	Billing Payment Method	Bill Unit	Account/Customer Billing Profile name
--	Bill Type	NA	NA	NA	Bill/Invoice type requested by customer (summary or detail)
--	Service Account	Summary	Contact Information	Company or First & Last Name	In cases where the billing account and service account are different. In such scenarios it results in a parent-child hierarchy in BRM. The service account is the nonpaying child account.
--	Bill Media	Payments	Billing Payment Method - Payment Options	Delivery Method	Delivery method for invoice. For example, delivery of invoices by *email, paper*, and so on.
--	Payment Method	Payments	Billing Payment Method	Payment Method	How customers pay their bills. Payment methods include *credit card, invoice, debit card*, and so on.
--	Billing Profile Status	NA	NA	NA	Current status of the customer's billing profile (*active* or *inactive*).

UI Component: Balance Summary, as described in Table A-2.

Table A-2 Balance Summary Mapping

Siebel Applet	Siebel Field	BRM Tab	BRM Form	BRM Field	Comments
Balance Summary Form	Balance	Balance	Balance Summary	Amount due for all bills	Total billed amount that has not been paid. This includes the balance impact of any unresolved dispute.
--	Due Now	Balance	Balance Summary	Due Now	Billed amount the customer currently owes. This is calculated as *Amount due for all bills* minus *Adjustments/Payments* that are not yet applied.
--	Pending Payments/ Adjustments	Balance	Balance Summary	Adjustments/ Payments not applied	Total of unallocated payments and unallocated account adjustments.
--	Currency	NA	NA	NA	--
--	Unresolved Disputes	Balance	Balance Summary	Unresolved Disputes	Total of disputed amounts that have been removed from the *Due Now* amount before the settlement of the dispute.
--	Total	Balance	Balance Summary	Total	Sum of the *Due Now* amount and the *Bill in Progress* (estimate) amount.
--	Unbilled Usage	Balance	Balance Summary	Bills in Progress	The *Bill in Progress* shows the current balance of the upcoming bill, including unbilled item charges, cycle forward arrears fees, and A/R actions on those charges and fees.

Bills

Navigate to the **Account Summary, Billing Profile** applet, **Billing Profile Name** link, and **Bills** tab.

Siebel View: Billing Invoice

UI Component: Bills, as described in Table A-3.

Table A-3 Bills Mapping

Siebel Applet	Siebel Field	BRM Tab	BRM Form	BRM Field	Comments
Invoice List Applet	Bill Number	Balance	Bills	Number	A unique number that identifies a specific bill. Each invoice contains a bill number. A bill is an object in the Oracle Communications BRM database that stores the balance impacts in the bill items of a subscriber's account during one billing cycle. Bills contain information about the subscriber's account, the account's billing cycle, and the amount billed.

Table A-3 (Cont.) Bills Mapping

Siebel Applet	Siebel Field	BRM Tab	BRM Form	BRM Field	Comments
--	Bill Period	Balance	Bills	Billing Cycle	The time period during which charges accumulate in an account before a bill is finalized. One billing cycle can contain one or more accounting cycles.
--	Amount Due	Balance	Bills	Balance	The original bill amount minus *Payments and A/R Actions*.
--	Due Date	Balance	Bills	Due Date	The date on which the bills payment is due.
--	Previous Balance	NA	NA	NA	--
--	Payments and A/R Actions	Balance	Bills	Payments and A/R Actions	The total sum of all payments made for a bill plus the A/R actions such as *adjustments* or *refunds*.
--	Bill Payment	Payments	Payments Received	Paid	The total payment made against a bill.

Navigate to the **Account Summary, Billing Profile** applet, **Billing Profile Name** link, **Bills** tab, and **Bill Number** link.

Siebel View: Billed Usage Detail

UI Component: Bill Details, as described in Table A-4, Service Charges, as described in Table A-5, and Item Charges as described in Table A-6.

Table A-4 shows the mappings for Bill Details.

Table A-4 Bill Details Mapping

Siebel Applet	Siebel Field	BRM Tab	BRM Form	BRM Field	Comments
Invoice Form Applet	Account	Summary	Contact Information	Company or First & Last Name	Account/Subscriber name
--	Profile Name	Payments	Billing Payment Method	Bill Unit	Account/Subscriber Billing profile name
--	Bill Number	Balance	Bill Details	Number	A unique number that identifies a specific bill. Each invoice contains a bill number. A bill is an object in the Oracle Communications BRM database that stores the balance impacts in the bill items of a subscriber's account during one billing cycle. Bills contain information about the subscriber's account, the account's billing cycle, and the amount billed.

Table A-4 (Cont.) Bill Details Mapping

Siebel Applet	Siebel Field	BRM Tab	BRM Form	BRM Field	Comments
--	Bill Period	Balance	Bill Details	Billing Cycle	The time period during which charges accumulate in an account before a bill is finalized. One billing cycle can contain one or more accounting cycles.
--	Amount Due	Balance	Bill Details	Balance	The original bill amount minus the *Payments and A/R Actions*.
--	Due Date	Balance	Bill Details	Due Date	The date on which the bills payment is due.
--	Previous Balance	NA	NA	NA	--
--	Payments and A/R Actions	Balance	Bill Details	Payments and A/R Actions	The total sum of all payments made for the bill plus the A/R actions such as *adjustments* or *refunds*.
--	Bill Payment	Balance	Bill Details	Total Payments	The total payment made against a bill.

Table A-5 shows the mappings for Service Charges.

Table A-5 Service Charges Mapping

Siebel Applet	Siebel Field	BRM Tab	BRM Form	BRM Field	Comments
Invoice Service Charge List	Description	Balance	Bill Details, Item Charges	Description	Name of the service
--	Charge	Balance	Bill Details, Item Charges	Charge	Holds charges of all items under the service.
--	Discount	Balance	Bill Details, Item Charges	Discount	Holds the total of all discounts given under the service or item.
--	Net Amount	Balance	Bill Details, Item Charges	Net	The *Net Amount* is the amount after discounts, payments, and A/R actions have been applied to the service.

Table A-6 shows the mappings for Item Charges.

Table A-6 Item Charges Mapping

Siebel Applet	Siebel Field	BRM Tab	BRM Form	BRM Field	Comments
Invoice Item Charge List	Description	Balance	Bill Details, Item Charges	Description	Item is an entity that represents a group of charges. For example, a *Cycle Forward* charge for the service VoIP instance.
--	Charge	Balance	Bill Details, Item Charges	Charge	Charge indicates the total amount for the item.

Table A-6 (Cont.) Item Charges Mapping

Siebel Applet	Siebel Field	BRM Tab	BRM Form	BRM Field	Comments
--	Discount	Balance	Bill Details, Item Charges	Discount	The Discount column shows any discount that is applicable to the item.
--	Net Amount	Balance	Bill Details, Item Charges	Net Amount	The *Net Amount* is the amount after discounts, payments, and A/R actions have been applied to the item.

Navigate to the **Account Summary, Billing Profile** applet, **Billing Profile Name** link, **Bills** tab, **Bill, Item Charges, Net Amount** link

Siebel View: CDR Details

UI Component: Event Details as described in Table A-7.

Table A-7 Event Details Mapping

Siebel Applet	Siebel Field	BRM Tab	BRM Form	BRM Field	Comments
CDR Details List	Date	Balance	Bill Details, Item Charges, Description	Date & Time	Date and time when the call was made. The time zone of the call is in the time zone in which the call was originally made.
--	Charge	Balance	Bill Details, Item Charges, Description	Charge	*Charge* indicates the total amount for the call (CDR).
--	Discount	Balance	Bill Details, Item Charges, Description	Discount	The *Discount* column shows any discount that is applicable.
--	Net Amount	Balance	Bill Details, Item Charges, Description	Net	Actual amount due after any discounts are applied to the charge.
--	Duration	Balance	Bill Details, Item Charges, Description	Quantity	Total time of the call.
--	Number Called	Balance	Bill Details, Item Charges, Description	Called No	Telephone number to which the call was made.
--	Non-Currency	Balance	Bill Details, Item Charges, Description	Non-Currency	If the event is of type *nonmonetary*, the column is checked.

Navigate to the **Account Summary, Billing Profile** applet, **Billing Profile Name** link, **Bills** tab, **Bill Number** link, **Payments** tab

Siebel View: Billed Usage Payments

UI Component: Bill Payments as described in Table A-8.

Table A-8 Bill Payments Mapping

Siebel Applet	Siebel Field	BRM Tab	BRM Form	BRM Field	Comments
Billed Usage Payments List	Payment Number	Balance	Bill Details, Payment Details	Payment Number	The payment item number to identify a payment.
--	Payment Date	Balance	Bill Details, Payment Details	Posted	The date on which the payment was posted.
--	Payment Method	Balance	Bill Details, Payment Details	Payment Type	The *Payment Method* identifies how subscribers paid their bill; for example, by credit card or direct deposit.
--	Payment Amount	Balance	Bill Details, Payment Details	Amount	The total amount that was paid by the subscriber as part of the payment.
--	Confirmation Number	Balance	Bill Details, Payment Details	Payment Number	The payment item number to identify a payment.
--	Allocated	Balance	Bill Details, Payment Details	Allocated	The payment amount that has been allocated to the bill.
--	Unallocated	Balance	Bill Details, Payment Details	Unallocated	The total amount from the payment made that is not yet applied.
--	Reversed	Balance	Bill Details, Payment Details	Reversed	If a particular payment has been reversed, then this column displays the value *Y*.

Navigate to the **Account Summary, Billing Profile** applet, **Billing Profile Name** link, **Bills** tab, **Bill Number** link, **A/R Items** tab

Siebel View: A/R items Details

UI Component: Bills A/R items as described in Table A-9.

Table A-9 Bills A/R Items Mapping

Siebel Applet	Siebel Field	BRM Tab	BRM Form	BRM Field	Comments
A/R Items List	Type	Balance	Bill Details, A/R Items	Type	The type of A/R actions, such as *Adjustment, Dispute, Refund, Write-off*, and so on.
--	Date	Balance	Bill Details, A/R Items	Date	The date on which the adjustment was made.
--	Resource Name	Balance	Bill Details, A/R Items	Resource	This column indicates to which resource the adjustment was made. For example, *currency* resource or *noncurrency* resource such as free seconds.
--	Amount	Balance	Bill Details, A/R Items	Amount	The adjustment amount that was made against a resource.

Balance Group

Navigate to the **Account Summary, Billing Profile** applet, **Billing Profile Name** link, **Balance Group** tab

Siebel View: Balance Group

UI Component: Balance Group Details (Balance Group, Balance, Balance Details and Services) as described in Table A-10.

Table A-10 Balance Group Details Mapping

Siebel Applet	Siebel Field	BRM Tab	BRM Form	BRM Field	Comments
Balance Group	Name	Balance	All Credit Limit, Currency Credit Limit	Balance Group	The name of the balance group. For example, *Account Level Balance Group*.
Balance	Balance	--	All Credit Limit, Currency Credit Limit	Outstanding	Total balance under the balance group.
--	Unit of Measure	NA	NA	NA	In case of monetary resource, this column in Siebel CRM displays *Currency* and for nonmonetary resource this column is blank.
--	Ceiling Credit Limit	--	All Credit Limit, Currency Credit Limit	Amount or Unlimited	1. If there is a value under the Account in BRM, this indicates the credit limit of the balance group. 2. If the *Unlimited* column is selected, this indicates there is no credit limit for the balance group.
--	Floor Credit Limit	NA	NA	NA	--
--	Threshold Credit Limit	NA	NA	NA	--
Balance Details (for monetary resource)	Available	Plan	Product Detail	Outstanding	Total balance under the balance group.
--	Valid From	Plan	Product Detail	Purchase Start Date	Purchase start or valid from date.
--	Valid To	Plan	Product Detail	Purchase End Date	Purchase end or valid to date.
--	No End	Plan	Product Detail	Check box	This is used when the product has unlimited validity.
Balance Details for non-monetary resource)	Available	Non-Currency	Non-Currency Details	Available	Total balance under the balance group.
--	Valid From	Non-Currency	Non-Currency Details	Valid From	Resource start or valid from date
--	Valid To	Non-Currency	Non-Currency Details	Valid To	Resource end or valid to date.
--	No End	Non-Currency	Non-Currency Details	Check box	This is used when the resource has unlimited validity.

Table A-10 (Cont.) Balance Group Details Mapping

Siebel Applet	Siebel Field	BRM Tab	BRM Form	BRM Field	Comments
Balance Group Services	BRM Service ID	Balance	All Credit Limit, Currency Credit Limit	Balance Group	If the product is part of a service bundle, then this column displays the *BRM Service ID* of the product under the balance group.
--	Service Type	Plans	Plans	Service	Billing service type of the product.
--	Product Name	Plans	Plans	Product/ Discount	Product name.
--	Effective Date	Plans	Plans	Purchased	Product purchased date.
--	Status	Plans	Plans	Status	Current status of the product (*active* or *canceled*).

Unbilled Usage

Navigate to the **Account Summary, Billing Profile** applet, **Billing Profile Name** link, **Unbilled** tab

Siebel View: Unbilled Usage

UI Component: Bill Details, Service Charges, Item Charges as described in Table A-11.

Table A-11 Bill Details, Service Charges, and Item Charges Mapping

Siebel Applet	Siebel Field	BRM Tab	BRM Form	BRM Field	Comments
Unbilled Usage	Description	Balance	Bills in Progress, Item Charges	Description	Name of the service.
--	Charge	Balance	Bills in Progress, Item Charges	Charge	Holds unbilled charges of all items under the service.
--	Discount	Balance	Bills in Progress, Item Charges	Discount	Holds the total of all discounts given under the service or item.
--	Net Amount	Balance	Bills in Progress, Item Charges	Net	The *Net Amount* is the amount after discounts, payments, and A/R actions have been applied to the service.
Unbilled Item Charges	Description	Balance	Bills in Progress, Item Charges	Description	Item is an entity that represents a group of charges. For example, a *Cycle Forward* charge for the service VoIP instance.
--	Charge	Balance	Bills in Progress, Item Charges	Charge	*Charge* indicates the total amount for the item.
--	Discount	Balance	Bills in Progress, Item Charges	Discount	The *Discount* column shows any discount that is applicable.
--	Net Amount	Balance	Bills in Progress, Item Charges	Net Amount	The *Net Amount* is the amount after discounts, payments, and A/R actions have been applied to the item.

Payments

Navigate to the **Account Summary, Billing Profile** applet, **Billing Profile Name** link, **Payments** tab

Siebel View: Billing Profile Payment

UI Component: Payments as described in Table A-12.

Table A-12 Payments Mapping

Siebel Applet	Siebel Field	BRM Tab	BRM Form	BRM Field	Comments
Billing Profile Payments List	Payment Number	Payments	Payments Received	Number	The payment item number to identify a payment.
--	Payment Date	Payments	Payments Received	Date	The date on which the payment was posted.
--	Payment Method	Payments	Payments Received	Payment Method	The *Payment Method* identifies how customers paid their bill; for example, by credit card or direct deposit.
--	Payment Amount	Payments	Payments Received	Paid	The total amount that was paid by the subscriber as part of the payment.
--	Comments	Payments	Payments Received	NA	--
--	Confirmation Number	Payments	Payments Received	Number	The payment item number to identify a payment.
--	Allocated	Payments	Payments Received	Allocated	The payment amount that has been allocated to the bill.
--	Unallocated	Payments	Payments Received	Unallocated	The total amount from the payment made that is not yet applied.
--	Reversed	Payments	Payments Received	Reversed	Any payments that were reversed.

B

Reintroducing Enterprise Business Services

This appendix provides instructions for reintroducing enterprise business services (EBSs) into the Oracle Application Integration Architecture (Oracle AIA) deployment.

EBSs are used to help route to multiple Providers. If you are using one source and one target system for your integration flows then EBSs are unnecessary. However, if you must dynamically identify a Provider system during runtime (content-based routing) then you should reintroduce EBSs.

> **Note**
>
> With the deployment of the Fusion Middleware Foundation Pack, web service definition language (WSDL) files are provided for all EBSs.

To reintroduce EBSs:

1. Go to JDevloper and create a new composite for the EBS with an Oracle Mediator service. Use the EBS WSDL provided by Fusion Middleware Foundation Pack.
2. Create routing rules in Oracle Mediator to route to appropriate Provider connectors.
3. Save your changes.
4. Open the **AIAConfigurationProperties.xml** file, which is located in: *COMMS_AIA_HOME*/**comms_home/source/soainfra/apps/config**.

   > **Note**
   >
   > Entries in the **AIAConfigurationProperties.xml** file are case sensitive.

5. To invoke new EBS connectors you need to replace the Provider connector's name and address with the EBS name and address.

   This action tells the Requestor to invoke EBS instead of the Provider application business connector service (ABCS).
6. Save and close the file.
7. Upload the changed file to the Oracle Metadata Services repository as described in *Oracle AIA Installation Guide*.

C

Using the Oracle Mediator Resequencer Feature

This appendix provides details about the Oracle Mediator Resequencer feature, which is used by various integration flows to ensure that messages are processed in a particular sequence.

See the discussion of resequencing in Oracle Mediator in *Oracle Fusion Middleware Developer's Guide for Oracle SOA Suite* for more information about resequencer.

Queues and Flows Enabled for Sequencing

Table C-1 lists the queues and flows that are enabled for sequencing.

> **Note**
>
> OSM manages scenarios where multiple revisions for the same order are sent out of sequence. If you are using a different order management system it must have similar support.

Table C-1 Queues and Flows Enabled for Sequencing

Oracle AIA Queue	Flow	JMS Priority	Sequencing Criteria	Comments
AIA_UPDSO_OUT_JMSQ	Update order flow from OSM to Oracle AIA for Siebel CRM.	Not set	Group By: Account ID mentioned in the ObjectCrossReference section of the update message(/UpdateSalesOrderEBM/EBMHeader/Sender/ObjectCrossReference/SenderObjectIdentification/AlternateObjectKey/ID[@schemeID = 'CUSTOMERPARTY_ACCOUNTID' and @schemeAgencyID = 'COMMON']) Order of Processing: FIFO (First in First Out). Composite Name: UpdateSalesOrderOSMCFSCommsJMSConsumer.	**Note:** The subscriber in the Create Trouble Ticket for Order Fallout business flow is only a sample. The resequencer in this flow ensures that multiple updates for the same order are processed in the right sequence.

Table C-1 (Cont.) Queues and Flows Enabled for Sequencing

Oracle AIA Queue	Flow	JMS Priority	Sequencing Criteria	Comments
AIA_CRTCUST_OUT_JMSQ	Order flow from OSM to Oracle AIA for customer data creation in billing.	Set by OSM	Group By: Account ID on the message (this is either the Billing account or the Service account on the order line that must be created in billing) and the target system identifier. concat($in.SyncCustomerPartyListEBM/ns0:SyncCustomerPartyListEBM/ns0:DataArea/ns0:SyncCustomerPartyList/ns0:CustomerPartyAccount/corecom:Identification/corecom:ApplicationObjectKey/corecom:ID[@schemeID='AccountId'], $in.SyncCustomerPartyListEBM/ns0:SyncCustomerPartyListEBM/corecom:EBMHeader/corecom:Target/corecom:ID) Order of Processing: FIFO (First in First Out). Composite Name: CommunicationsCustomerPartyEBSV2Resequencer.	The resequencer in this flow ensures that the solution can successfully handle processing of concurrent orders for the same subscriber.
--	Sync customer flow from Siebel CRM system to Oracle Customer Hub.	Not Set	Group By: AccountID. Order of Processing: FIFO (First in First Out). Composite Name: SyncAcctSiebelAggrEventConsumer SyncContSiebelAggrEventConsumer.	Also available in the Cash to Care business processes. The resequencer in this flow ensures that multiple updates for the same subscriber are processed in the right sequence.
AIA_CRTFO_IN_JMSQ	Order flow from Oracle AIA to OSM.	Set by ProcessSalesOrderFulfillmentOSMCFSCommsJMSProducer	None. (Onus is on OSM.)	NA
AIA_CRTBO_OUT_JMSQ	Order flow from OSM to AIA for billing.	Set by OSM	None as delivered. You can use ProcessFulfillmentOrderBillingOSMCFSCommsJMSConsumer to implement custom sequencing.	NA
AIA_UPDBO_IN_JMSQ	Order flow from AIA (from billing) to OSM.	Set by ProcessFulfillmentOrderBillingResponseOSMCFSCommsJMSProducer	None. (Onus is on OSM.)	NA

Table C-1 (Cont.) Queues and Flows Enabled for Sequencing

Oracle AIA Queue	Flow	JMS Priority	Sequencing Criteria	Comments
AIA_UPDCUST_IN_JMSQ	Response of the customer creation in billing from AIA to OSM.	Set by ProcessFOBillingAccountListRespOSMCFSCommsJMSProducer	None. (Onus is on OSM).	NA
AIA_CRTFO_OUT_JMSQ	Create Fulfillment Order flow from OSM to Oracle AIA for the provisioning system.	Set by OSM.	None as delivered. Customer can use ProcessProvisioningOrderOSMCFSCommsJMSConsumer to implement custom sequencing.	NA
AIA_FOCFS_IN_JMSQ	Update Fulfillment Order flow from Oracle AIA (from the provisioning system) to OSM).	Set by ProcessFulfillmentOrderUpdateOSMCFSCommsJMSProducer	None. (Onus is on OSM).	NA
AIA_FOPROV_OUT_JMSQ	Update Fulfillment Order flow from the provisioning system to Oracle AIA (for OSM)	Set by provisioning system	None as delivered. Customer can use ProcessFulfillmentOrderUpdateOSMPROVCommsJMSConsumer to implement custom sequencing.	NA
AIA_FOPROV_IN_JMSQ	Create Fulfillment Order from Oracle AIA (from OSM) to the provisioning system.	Set by ProcessProvisioningOrderOSMPROVCommsJMSProducer	None. (Onus is on OSM).	NA

Resolving Errors in Flows with Resequencer

If an error occurs in the Oracle Communications Billing and Revenue Management (BRM) Subscriber provider, the message may be blocked in the CommunicationsCustomerPartyEBSV2Resequencer service and the error message may not propagate back to CommsProcessFulfillmentOrderBillingAccountListEBF. In these situations, fulfillment fallout specialists must take corrective action on the resequencer to move the flow. If the message fails due to a system error (for example, if the target system is unavailable), then fulfillment fallout specialists must retry the message from resequencer. If the message fails because of a business error, then the fulfillment fallout specialist must unblock the resequencer.

An error may occur in the Siebel CRM provider after it is consumed by UpdateSalesOrderOSMCFSCommsJMSConsumer and sent for processing. In this situation the messages are rolled back to the resequencer for this subscriber and any subsequent order updates for that particular order are not processed. If this occurs, the fulfillment fallout specialist must take corrective action on this resequencer to move the flow like the ones described above. If the message fails due to a system error (for example, if the target system is unavailable), then fulfillment fallout specialists must retry the message from resequencer. If

the message fails because of a business error, then the fulfillment fallout specialist must unblock the resequencer.

See the discussion of monitoring resequenced messages in *Oracle Fusion Middleware Administrator's Guide for Oracle SOA Suite and Oracle Business Process Management Suite* for more information on unblocking and retrying.