Oracle Insurance Revenue Management and Billing

Version 5.1.0.0.0

Release Notes – Additional Features

Revision 1.2

F87552-01

October 2023

ORACLE®

Oracle Insurance Revenue Management and Billing Version 5.1.0.0.0 Release Notes – Additional Features

Note: To improve the content readability, the Oracle Insurance Revenue Management and Billing product is referred to as Oracle Revenue Management and Billing throughout this document.

F87552-01

Document Category: Public

Hazardous Applications Notice

This software or hardware is developed for general use in a variety of information management applications. It is not developed or intended for use in any inherently dangerous applications, including applications that may create a risk of personal injury. If you use this software or hardware in dangerous applications, then you shall be responsible to take all appropriate fail-safe, backup, redundancy, and other measures to ensure its safe use. Oracle Corporation and its affiliates disclaim any liability for any damages caused by use of this software or hardware in dangerous applications.

Trademark Notice

Oracle, Java, MySQL, and NetSuite are registered trademarks of Oracle and/or its affiliates. Other names may be trademarks of their respective owners.

Intel and Intel Inside are trademarks or registered trademarks of Intel Corporation. All SPARC trademarks are used under license and are trademarks or registered trademarks of SPARC International, Inc. AMD, Epyc, and the AMD logo are trademarks or registered trademarks of Advanced Micro Devices. UNIX is a registered trademark of The Open Group.

Third-Party Content, Products, and Services Disclaimer

This software or hardware and documentation may provide access to or information about content, products, and services from third parties. Oracle Corporation and its affiliates are not responsible for and expressly disclaim all warranties of any kind with respect to third-party content, products, and services unless otherwise set forth in an applicable agreement between you and Oracle. Oracle Corporation and its affiliates will not be responsible for any loss, costs, or damages incurred due to your access to or use of third-party content, products, or services, except as set forth in an applicable agreement between you and Oracle.

About this Document

This document describes the new features, enhancements, and user interface level changes that are introduced for the health insurance domain through the Release Notes – Additional Features.

This document does not describe the functionality of Oracle Revenue Management and Billing or technical know-how on how to install or upgrade Oracle Revenue Management and Billing. For more detailed information, you can refer to the following documents:

Document Name	Description
Oracle Revenue Management and Billing Licensing Guide	Lists different features which you can use when you acquire a license for the following products: • Oracle Financial Services Revenue Management and Billing • Oracle Insurance Revenue Management and Billing It also provides the licensing information about the third-party JARs and components which are included in the above mentioned products.
Oracle Revenue Management and Billing Quick Installation Guide	Provides information about the media packs which are available for the current release. It contains information about the supported platforms, space requirements, and release-specific documentation library. In addition, it contains high-level information on how to install ORMB and selected additional software.
Oracle Revenue Management and Billing Installation Guide	Provides an overview about the application architecture. It contains detailed information about the software and hardware requirements, supported platforms, application server and database space requirements, and application server pre-requisites for supported platforms. It explains the installation and configuration menu options which are available during installation and advanced menu options which are available post installation. It also explains how to configure WebSphere application server and how to install OUAF, OUAF rollup packs, ORMB, and ORMB patches or rollup packs. It contains appendices which lists ORMB rollup packs and notices about third-party JAR and other components.

Document Name	Description
Oracle Revenue Management and Billing Database Administrator's Guide	Provides information about the supported database platforms and explains how to install database with or without demo data. It contains the standards and conventions that should be followed while working with ORMB database. In addition, it lists various configurations and implementation guidelines with respect to tablespace, encryption, storage, compression, indexes, initialization parameters, listener, table partitioning, performance monitoring, memory resource management, optimizer statistics and segment advisors, etc. It contains various appendices which lists the new objects added in ORMB, application services configured for default user groups, new objects added in OUAF, information about OUAF system tables and ORMB rollup packs, notices about third-party JAR and other components.
Oracle Revenue Management and Billing Security Guide	Highlights what's new in security! Describes all the security features available in ORMB for authentication, authorization, user access, database access, LDAP and SSO integration, audit, encryption, web services, and JNDI access. It also describes how to provide support for data masking, Oracle cloud object storage, groovy, HTTP proxy, keystore, truststore, whitelist, federated architecture, and object erasure. In addition, it explains how to garble the customer information.
Oracle Revenue Management and Billing Server Administration Guide	Provides detailed information about the product architecture, native support, directory structures, environment variables, logs, etc. It explains various concepts related to the batch server and lists and describes the scripts required for installing and configuring ORMB. It explains how to monitor the system and configure Web browser, Web application server, business application server, inbound web services, and batch server. It explains how to provide Oracle Cloud Support and how to integrate and monitor Oracle Scheduler. It contains various appendices which lists and describes the parameters available in the ENVIRON.INI, Web.xml, spl.properties, hibernate.properties, submitbatch.properties, threadpoolworker, coherence-cache.config.xml, and tangosol-coherence-override.xml files.

Document Name	Description
Oracle Utilities Application Framework Business Process Guide	Explains how to get acquainted with the user interface. It explains the different types of pages or portals that you may come across in the application. It explains how to set the user preferences and how to create, manage, assign, and complete a To Do in the application. It also explains how to submit reports and view historic reports in the application.
Oracle Utilities Application Framework Administrative Guide	Explains the general, security, user, designing, developing, and scripting options available in Oracle Utilities Application Framework (OUAF). It describes the user interface, database, configuration, and reporting tools available in OUAF. In addition, it provides information about the application viewer, how to configure incoming and outgoing messages, and how to integrate Lightweight Directory Access Protocol (LDAP), Oracle Identity Manager (OIM), and Batch Scheduler with Oracle Revenue Management and Billing (ORMB).
Oracle Revenue Management and Billing Business Process Guide	Explains how to maintain the demographic, geographic, and financial objects (i.e. accounts) of a customer. It explains how to manage a customer's bills, payments, adjustments, credits, collections processing, statements and deposits in Oracle Revenue Management and Billing (ORMB). It also describes the financial transactions, case management, sales and marketing functions, rates engine, quotations, loans, how to monitor and execute job streams, and how to manage workflows, notifications, and overdue processing. In addition, it explains how to extract the data from the system using an extract template. The features listed and described in this document can be used in both financial services and health insurance domains.
Oracle Revenue Management and Billing Administrative Guide	Explains how to configure various features and functionalities in Oracle Revenue Management and Billing (ORMB). For example, billing, payments, adjustments, financial transactions, credits, collections processing, loans, service credits, background processes, quotations, case management, security, overdue processing, batch scheduler, workflow, and notifications, etc. The information available in this document can be used in both financial services and health insurance domains.

Document Name	Description
Oracle Revenue Management and Billing Banking User Guide	Describes various features which are available for the financial services business. For example, customer registration, customer 360° view, invoicing group, pricing management, multi-currency accounts, currency conversion, construct based billing and settlement, trial billing, product lifecycle management, subscription billing, mass pricing update, accrual, foreign exchange gain loss, transaction feed management, upload validated payment and adjustment data, freeze payments on notification, payment request, offset request, funding request, hold request, refund/write off request, dispute request, upload request, earnings credit rate, payment agreement request, invoice request, deal management, etc.
	It describes all screens related to these features and explains how to perform various tasks related to the feature in the application.
Oracle Revenue Management and Billing Insurance User Guide	Describes various features which are available for the following three lines of health insurance business – fully insured group, self-funded, and individual. For example, customer registration, customer 360° view, pricing management, trial billing, transaction feed management, upload validated payment and adjustment data, freeze payments on notification, payment request, offset request, funding request, hold request, refund/write off request, upload request, payment agreement request, invoice request, inbound message processing, reconciliation, entity audit, premium repricing, etc.
	It describes all screens related to these features and explains how to perform various tasks related to the feature in the application.
Oracle Revenue Management and Billing Upgrade Path Guide	Explains the path and pre-requisites for upgrading Oracle Revenue Management and Billing from one version to another.

Document Name	Description
Oracle Revenue Management and Billing Upgrade Guide	Explains how to upgrade the ORMB application server and database from one version to another. It also explains how to migrate the ORMB data from one version to another and describes the additional tasks that you need to perform after upgrading from one version to another. It includes various appendices that contain information about new tables introduced in the current release, existing tables which are modified in the current release, dropped algorithms and algorithm types, dropped characteristic types, dropped algorithm parameters, dropped option types in feature configurations, ORMB rollup packs, and SQL statements used for data migration.
Oracle Revenue Management and Billing Direct Database Upgrade Guide	Explains how to directly upgrade the ORMB database from 2.5.0.1.0 or any later version to the current release. It also highlights any known issues during direct database upgrade and how to handle these issues in the database.
Oracle Revenue Management and Billing Transaction Feed Management - Batch Execution Guide	Explains the sequence in which the batches should be executed while performing various tasks in the Transaction Feed Management (TFM) module. It provides detailed information about each TFM batch and its parameters. It also indicates the restart and multi-threading ability of each batch. In addition, it recommends values for various parameters which can be used for tuning batch performance as per the available hardware.
Oracle Revenue Management and Billing Batch Guide	Provides detailed information about various batches which are used in different modules, such as billing, payments, financial transaction, pricing management, funding request, offset request, hold request, upload request, inbound message, payment agreement request, accruals, earnings credit rate, ILM, deferred revenue recognition, reconciliation, garbling, repricing, entity audit, statements, etc. It also contains information about the batch parameters and the batch restart and multi-threading abilities.
Oracle Revenue Management and Billing Information Lifecycle Management (ILM) Implementation Guide	Provides an overview of the Information Lifecycle Management (ILM) feature. It describes how to implement ILM for the Transaction Feed Management (TFM) and Billing modules. It also provides detailed information about the ILM batches and their parameters.
Oracle Revenue Management and Billing FOP Reports Guide	Explains how to extract data from the system using various FOP reports in Oracle Revenue Management and Billing.
Oracle Revenue Management and Billing Chatbot Configuration Guide	Explains how to integrate Oracle Digital Assistant (ODA) with Oracle Revenue Management and Billing.

Document Name	Description
Oracle Revenue Management and Billing Chatbot User Guide	Explains how to use the menu based Chatbot introduced in Oracle Revenue Management and Billing.
Oracle Revenue Management and Billing ML Integration Guide	Explains how to integrate Machine Learning (ML) with Oracle Revenue Management and Billing for anomaly detection.

Change Log

Revision	Last Update	Updated Section	Comments
1.1	29-Nov-2023	Termination Date Rule Priority	Added Section
		Delinquency Termination Date Rule Business Rule	Added Section
		Delinquency Miscellaneous Options Business Rule	Added Section
		Canceling a Delinquency Process	Added Section
		Field Mapping	Added Section
		Business Rules	Added Section
		Parameter	Added Section
		Collection Class (New Features)	Added Section
		Types of Delinquency Process	Added Section
		Grace Period Configuration	Added Section
		Termination Date Rule Priority	Added Section
		Trigger Date Recalculation Configuration	Added Section
		Resuming a Delinquency Process	Added Section
		Collection Class (Enhancements)	Updated Information
		Delinquency Control	Updated Information
		Delinquency Event Type	Updated Information
		Delinquency Event Activation	Updated Information
		Delinquency Process Type Algorithms	Updated Information
		Termination and Reinstatement Configuration	Updated Information
		Approval Configuration	Updated Information
		Manual Actions at the Delinquency Process Level	Updated Information
		Delinquency Grace Period Business Rule	Updated Information
		Delinquency Event Attribute Business Rule	Updated Information
		Customer 360° View	Updated Information

Revision	Last Update	Updated Section	Comments
		Delinquency Management for Individual Health Insurance Business	Updated Information
		User Interface (UI) Level Changes	Updated Information
1.2	05-Dec-2023	Debt Notification Delinquency Process	Added Section
		Debt Collection Letters Delinquency Process	Added Section
		Membership Cancellation Delinquency Process	Added Section
		Membership Termination Delinquency Process	Added Section
		Resuming a Delinquency Process	Added Section
		Membership Cancellation Outbound Message	Added Section
		Membership Termination Outbound Message	Added Section
		Outbound Message Generation for Delinquency Management	Added Section
		Adjustments	Added Section
		Delinquency Management for Individual Health Insurance Business	Updated Information
		Collection Class	Updated Information
		Delinquency Event Activation	Updated Information
		Delinquency Event Completion	Updated Information
		Delinquency Process Type Algorithms	Updated Information

Access to Oracle Support

Oracle customers that have purchased support have access to electronic support through My Oracle Support. For information, visit http://www.oracle.com/pls/topic/lookup?ctx=acc&id=info or visit http://www.oracle.com/pls/topic/lookup?ctx=acc&id=trs if you are hearing impaired.

Contents

New Features (Specific to Insurance)

This section describes the following new features added in this release which can be used in the health insurance domain:

- [Automatic Offset of Debit and Credit Bill Line Items](#)
- [To Do Notification on Binder Payment Cancellation](#)
- [Transfer Binder Payment from General Suspense Account to Individual Membership Account](#)
- [Reconciliation of Individual Health Insurance Subsidy Payments](#)
- [Reconciliation Discrepancy Report of Individual Health Insurance Subsidy Payments](#)
- [Deferred Processing Mode for Discrepancy Report Creation](#)
- [Invoice Request for Individual Health Insurance Billing](#)
- [Delinquency Management for Individual Health Insurance Business](#)
- [Outbound Message Generation for Delinquency Management](#)
- [Delinquency Grace Period Business Rule](#)
- [Delinquency Event Attribute Business Rule](#)
- [Delinquency Termination Date Rule Business Rule](#)
- [Delinquency Miscellaneous Options Business Rule](#)

Automatic Offset of Debit and Credit Bill Line Items

The fully insured individual business do come across various scenarios wherein the customers would need a facility to automatically offset the debit and credit bill line items of an account. For example,

- An individual enrolls for a health plan. But the individual fails to pay the binder payment before the membership start date. As a result, the individual membership status is changed to **Canceled**. On canceling the individual membership, an adhoc invoice is generated in the system and the premium billed for the individual membership is reversed. In this scenario, you may want to automatically offset the debit and credit bill line items of the individual member's account.

- An individual enrolls for a health plan having subsidy. The subsidy fully covers the premium, and therefore a zero balance bill is generated. If you internally offset a zero balance bill, the system matches the debit and credit line items of a zero balance bill on the bill completion. But, in this case, the match events are not created at the bill line item level. In such scenario, you may want to automatically offset the debit and credit bill line items of the individual member's account such that match events are created at the bill line item level.

Oracle Revenue Management and Billing facilitates you with the auto maintenance process for accounts wherein the system will automatically offset the debit and credit bill line items of the accounts. The auto maintenance process is driven using the **Offset Request** feature ensuring that the match events are created at the bill line item level. Note that, at present, the auto maintenance process for accounts is only tested and certified for the different lines of business in the health insurance domain.

While running the auto maintenance process for an account, the system creates an auto maintenance offset request for the account when either of the following criteria is met:

- Total amount of the open and unmatched items of the account is equal to zero. For example, an A1 account has two bills – B1 for Jan 2023 with the open balance 200 and B2 for Feb 2023 with the open balance -200 and thereby the A1 account balance is zero.

- Total amount of the open and unmatched items of the account in a specific coverage period is equal to zero. For example, an A2 account has three bills – B1 for Jan 2023 with the open balance 300, B2 for Feb 2023 with the open balance 300, and B3 for Feb 2023 with the open balance -300. Here, the overall account balance of A2 is not zero, but the account balance of A2 for Feb 2023 is zero.

Note: While running the auto maintenance process for an account, the system does not consider the payments applied on the On Account, Excess Credit, and Suspense contracts of the account.

The system creates an auto maintenance offset request for an account using an offset request type where the offset category is set to **Auto Maintenance**. A new field named **Offset Category** is available while defining an offset request type. It helps to differentiate between the traditional (i.e. manual) offset requests and automatic offset requests. If you want to create a manual offset request using an offset request type, you must set the offset category in the offset request type to **Regular**. However, if you want to create an automatic offset request using an offset request type, you must set the offset category in the offset request type to **Auto Maintenance**. By default, the offset category is set to **Regular**. Note that you cannot edit the offset category in an offset request type when an offset request is already created using the offset request type.

The system enables you to skip a set of accounts for which auto maintenance is not required. A new field named **SKIP_AUTOM_SW** is added in the **CI_ACCT** table. It indicates whether you want to skip auto maintenance for the account. At present, you can provide the skip auto maintenance information for an account through a healthcare inbound message and not from the user interface. The system then displays the skip auto maintenance information of an account in the **Main** section of the **Account Information** zone on the **Account** tab of the **Customer 360° Information** screen. Note that the **Skip Auto Maintenance** field appears in the **Main** section for the account only when it is set to **Y**.

To use the auto maintenance feature for accounts, you need to do the following:

- Create an offset request type where the offset category is set to **Auto Maintenance**

Note: While creating an offset request type for auto maintenance of accounts, you need to ensure that the **Bill Line Items Level Offset** option is selected and the **Transfer Adjustment** option is not selected.

- Set the **Avoid internal match event creation for zero dollar bill** option type of the **C1_FINTRANOP** feature configuration to **Y**

A new batch named **C1-AUTOM** is introduced in this release. It considers those accounts for auto maintenance where the **Skip Auto Maintenance** field is set to **N**. It then checks whether either of the following criteria is met for the account:

- Total amount of the open and unmatched items of the account is equal to zero

- Total amount of the open and unmatched items of the account in a specific coverage period is equal to zero

Here, the system considers the eligible accounts with zero and non-zero balance. If the eligible account has zero balance, the system considers the bills with zero and non-zero balance where the open and unmatched debit and credit line items can be offset. If the eligible account has non-zero balance, the system considers the bills with zero and non-zero balance where the open and unmatched debit and credit line items of the same coverage period can be offset.

If either of the above two criteria is met, the system creates and processes an auto maintenance offset request. The auto maintenance offset request automatically closes the open and unmatched bill line items of the account that amounts to zero balance. To close an open and unmatched bill line item of the account, the system creates an offset request adjustment against each such debit or credit financial transaction. If the match event is present on any financial transaction, the same match event is stamped on the offset request adjustment. However, if the match event does not exist on any financial transaction, the system creates a new match event and stamps it against the financial transaction and its corresponding offset request adjustment.

This batch is a multi-threaded batch. The multi-threading is based on account ID and chunks for multi-threading are created based on numerical distribution of account ID. This batch contains the following parameters:

Parameter Name	Mandatory (Yes or No)	Description
Division	Yes (Conditional)	Used when you want to run the auto maintenance process for the accounts which belong to a particular division. **Note:** This parameter is required when you specify a customer class while executing this batch.
Customer Class	No	Used when you want to run the auto maintenance process for the accounts which belong to a particular customer class.
Offset Request Type	Yes	Used to indicate the offset request type using which you want to create auto maintenance offset requests for the accounts. **Note:** Here, you must specify an offset request type where the offset category is set to **Auto Maintenance**.
Freeze Number of Days	Yes	Used to specify the number of days. The system considers those accounts for auto maintenance whose financial transactions are frozen in the last specified number of days. **Note:** The parameter value must be greater than or equal to zero.

Parameter Name	Mandatory (Yes or No)	Description
Skip Auto Maintenance for Accounts with Open Refund or Write Off Request (Y/N)	Yes	Used to indicate whether you want to skip auto maintenance for those accounts for which the refund or write off request is still in the non-final status. The valid values are: • Y • N **Note:** By default, the parameter value is set to **Y**.
Skip Auto Maintenance for Accounts with Open Payment, Payment Transfer Request (Y/N)	Yes	Used to indicate whether you want to skip auto maintenance for those accounts for which the payment creation or payment transfer request is still in the non-final status. The valid values are: • Y • N **Note:** By default, the parameter value is set to **Y**.
Skip Auto Maintenance for Accounts with Open Offset Request (Y/N)	Yes	Used to indicate whether you want to skip auto maintenance for those accounts for which the manual offset request is still in the non-final status. The valid values are: • Y • N **Note:** By default, the parameter value is set to **Y**.
Account ID	No	Used when you want to run the auto maintenance process for a particular account.
Thread Pool Name	No	Used to specify the thread pool on which you want to execute the batch.

On successful completion of this batch, the system creates and processes an auto maintenance offset request for an account that meets the criteria. You can view the details of an auto maintenance offset request through the **Offset Request** screen.

While viewing the details of an auto maintenance offset request, the system displays the **Auto Maintenance Financial Transactions** zone instead of the **Selected Bills** and **Offset Request Adjustments** zones in the **Main** tab of the **Offset Request** screen. The system also displays the account information in the **Main** section of the **Offset Request** zone. The **Auto Maintenance Financial Transactions** zone lists the financial transactions of the account for which the corresponding offset request adjustments are created during the auto maintenance process.

To design the auto maintenance process for accounts, the following algorithms are enhanced:

- **C1-OFFSUBENT** – Here, some of the existing validations are skipped for an auto maintenance offset request and few new validations are introduced for an auto maintenance offset request.

- **C1-OFST-DIST** – Here, additional logic is added to handle the offset category and to create auto maintenance offset requests for accounts that meet the criteria. Note that the **Match Type**, **Match Type Entity**, and **Credit Bill Distribution Algorithm** parameters are not applicable while creating offset request adjustments for an auto maintenance offset request.

The system enables you to cancel an auto maintenance offset request, if required, through the **Unapply** button. While canceling an offset request, you need to specify a reason why you want to cancel an offset request. On canceling an auto maintenance offset request, the system cancels the offset request adjustments and opens the corresponding bill line items which were offset during auto maintenance of the account.

To Do Notification on Binder Payment Cancellation

Oracle Revenue Management and Billing enables you to generate a To Do notification when a binder payment for an individual membership is canceled. To generate the To Do notification when a binder payment is canceled, you need to do the following setup:

- Add the following value in the **MEMB_PERS_STAT_RSN_FLG** lookup field:

   o   Binder Payment Canceled

- Add the following status reason for the **Active** status of the **C1-IndMembership** business object:

   o   Membership is active but binder payment is canceled

- Add the following the status reason mapping in the preference which is defined using the **Membership Status Reason Mapping** (**C1-MemStatusReasonMapping**) business object:

Member Person Status	Member Person Status Reason	Membership Status Reason
Active	Binder Payment Canceled	Membership is active but binder payment is canceled

- Define the following attributes in the preference which is defined using the **C1-FieldMapping** (**C1-FieldMapping**) business object and where the preference category is set to **Binder Payment**:

Attribute	Value	Entity Type
Membership Status Reason when Binder Payment Cancelled	Membership Status Reason Code (of the **Membership is active but binder payment is canceled** status)	Membership
To Do Type for Binder Payment Cancelled	C1-BPCNC	Membership

A new To Do type named **C1-BPCNC** and algorithm type named **C1-BNDPYCNCL** are introduced in this release. To generate the To Do notification during binder payment cancelation, you need to attach an algorithm created using the **C1-BNDPYCNCL** algorithm type to the **Payment Cancellation** system event of the required customer class. If the **C1-BNDPYCNCL** algorithm is attached on the customer class, the system checks whether the cancel reason specified while canceling a payment is listed as a value of the **Binder Payment Cancel Reason List** parameter in the algorithm. If so, the system then validates whether the payment is a binder payment using the **Binder Payment Identification Characteristics Type** attribute of the binder payment preference. If the payment is a binder payment, the system then determines the individual membership whose binder payment is canceled.

Points to Note:

If a payment contains a membership identifier value characteristic that is specified in the **Binder Payment Identification Characteristics Type** attribute of the binder payment preference, the system considers the payment as a binder payment.

The system derives the individual membership using the membership identifier value characteristic on the binder payment and the **Binder Payment Identification ID Type** attribute of the binder payment preference.

Once the individual membership is derived, the system fetches the binder payment applicability flag for the individual membership. If the binder payment applicability flag of the individual membership is set to **No**, the system will not generate the To Do notification when the binder payment is canceled.

However, if the binder payment applicability flag of the individual membership is set to **Yes**, the system will derive the binder payment threshold amount for the respective membership. If the binder payment threshold amount is equal to zero, the system will not generate the To Do notification when the binder payment is canceled. If the binder payment threshold amount is greater than zero, the system will calculate the grace date (Membership Start Date + (Binder Payment Grace Days -1)) and the total binder payment amount until the grace date. If the **Consider Binder Liability Amount** attribute in the binder payment preference is set to **Y**, the system checks whether the total binder payment amount until the grace date is less than the threshold amount. If the total binder payment amount until the grace date is less than the threshold amount, the system does the following:

- Updates the member person status reason to **Binder Payment Canceled** and the membership status reason to **Membership is active but binder payment is canceled**
- A To Do using the **C1-BPCNC** To Do type is created

However, if the total binder payment amount until the grace date is greater than or equal to the threshold amount, the system neither changes the member person and membership status reasons nor creates a To Do notification when the binder payment is canceled.

If the **Consider Binder Liability Amount** attribute in the binder payment preference is set to **N**, the system checks whether the total binder payment amount until the grace date is equal to zero. If the total binder payment amount until the grace date is equal to zero, the system does the following:

- Updates the member person status reason to **Binder Payment Canceled** and the membership status reason to **Membership is active but binder payment is canceled**

- A To Do using the **C1-BPCNC** To Do type is created

However, if the total binder payment amount until the grace date is greater than zero, the system neither changes the member person and membership status reasons nor creates a To Do notification when the binder payment is canceled.

This algorithm contains the following parameters:

- **Membership Business Object** – Used to specify the individual membership business object. The system will then consider only those individual memberships which are created using the specified business object. Here, you must set the value to **C1-IndMembership**.
- **Algorithm for TO DO Creation** – Used to specify the algorithm using which you want to create a To Do notification during binder payment cancellation. Here, you must set the value to an algorithm which is created using the **C1-BNDTODO** algorithm type.
- **Binder Payment Cancel Reason List** – Used to specify a comma-separated list of payment cancel reasons that the system should consider while To Do generation on payment cancelation. You can specify maximum five comma-separated values for this parameter. Here, you must specify a payment cancel reason which is already defined in the system.

All the above parameters are mandatory.

Note that the system will update the membership status to **Pending Effectuation** only when the inbound message is received to change the membership status to **Pending Effectuation**. Until then, the member person and membership statuses will not change even if the respective binder payment is canceled.

Transfer Binder Payment from General Suspense Account to Individual Membership Account

Usually, in a fully insured individual business, the binder payments received before the individual membership enrollment are parked in the general suspense account. Oracle Revenue Management and Billing enables you to transfer the binder payments of an individual membership from the general suspense account to the individual member's account after the individual membership enrollment information is received through an inbound message.

To transfer the binder payment from general suspense account to individual membership account, you need to set the value for the following option type in the **C1_CMO** feature configuration:

- **Suspense Contract Type** – Used to indicate the suspense contract type whose suspense contracts' payments you want to transfer to the individual membership account.

A new batch control named **C1-PYTRS** is introduced in this release. It fetches the suspense contracts of the contract type which is specified in the **Suspense Contract Type** option type of the **C1_CMO** feature configuration. Once all suspense contracts are derived, the batch fetches the unmatched frozen payments which are made against the suspense contracts. The batch then checks whether the characteristic type specified in the **Payment Identifier Characteristic Type** parameter is defined on the payment. If the specified characteristic type is defined on the payment, the system considers the payment as the binder payment. If the specified characteristic type is not defined on the payment, the system does not consider the payment as the binder payment.

The batch then derives the individual membership for which the binder payment is made. While deriving the individual membership, the batch uses the membership identifier value characteristic defined on the binder payment and the value specified in the **Membership Identifier Type** parameter. You can specify a comma-separated list of membership identifier types in the **Membership Identifier Type** parameter. The batch then checks whether an individual membership exists with either of the membership identifier type and value combination in the system. Once the individual membership is derived, the batch fetches the individual member's account using the account identifier type and value characteristics which are defined on the individual membership.

> **Points to Note:**
>
> While deriving the account identifier type and value characteristics, the system considers the characteristic types which are specified in the **Account Identifier Type Char Type** and **Account Identifier Value Char Type** option types of the **C1-ASOBLLNG** feature configuration.
>
> If multiple individual memberships are derived with the membership identifier type and value combination, the system will not consider the binder payment for transfer and will accordingly add an appropriate log entry to such binder payments.

Once the individual member's account is derived, the batch transfers the binder payment from the suspense contract to either the unpaid bills or on account contract of the account depending on the following:

- If the match type is specified in the **Bill Match Type** parameter, the batch will transfer the binder payment from the suspense contract against the unpaid bills of the account. While transferring the binder payment against the unpaid bills of the account, the system will do either of the following:

    o If the **C1-MD-ACCT** manual distribution algorithm is attached to the match type, the system will distribute the binder payment against the unpaid bills of the account in the ascending order of the due date. If there are multiple bills with the same due date, the system will consider the unpaid bill in the ascending order of the unpaid amount. However, if there are multiple bills with the same due date and amount, the system will randomly consider the unpaid bill for payment. If there is any excess credit, the system will apply the excess credit amount on the on account contract of the individual member's account. Note that the on account contract is derived using the match type specified in the **On Account Match Type** parameter.

      o   If a manual distribution algorithm is not attached to the match type, the system will distribute the binder payment against the unpaid bills of the account in the ascending order of the due date. If there is any excess credit, the system will apply the excess credit amount on the on account contract of the individual member's account. Note that the on account contract is derived using the match type specified in the **On Account Match Type** parameter.

- If the match type is not specified in the **Bill Match Type** parameter, the batch will transfer the binder payment from the suspense contract to the on account contract of the individual member's account. Note that the on account contract is derived using the match type specified in the **On Account Match Type** parameter.

While transferring the binder payment from the suspense contract to either the unpaid bills or on account contract, the system cancels the old payment on the suspense contract and creates a new payment against the unpaid bills or on account contract of the individual member's account. While canceling the old payment, the system uses the cancel reason specified in the **Payment Cancel Reason for Transfer** parameter. While creating a new binder payment, the system copies all characteristics from the old payment before changing its status to **Frozen**.

This batch is a multi-threaded batch. The multi-threading is based on account ID and chunks for multi-threading are created based on numerical distribution of account ID. This batch contains the following parameters:

Parameter Name	Mandatory (Yes or No)	Description
Payment Identifier Characteristic Type	Yes	Used to specify the characteristic type which stores the membership identifier value. The system then considers a payment as a binder payment if the specified characteristic type is defined on the payment. **Note:** You must specify the characteristic type where the characteristic entity is set to **Payment**.
Membership Identifier Type	Yes	Used to indicate the membership identifier type using which you want to derive the individual membership. If you specify multiple membership identifier types separated by comma, the system will derive the individual membership using either of the membership identifier type and value combinations. **Points to Note:** You must specify a membership identifier type which is already defined in the system. You can specify maximum twenty comma-separated values for this parameter.

Parameter Name	Mandatory (Yes or No)	Description
Bill Match Type	No	Used to indicate the match type using which you want to transfer the binder payment from the suspense contract to the unpaid bills of the individual member's account. **Note:** You must specify a match type where the entity type is set to **Account**.
On Account Match Type	Yes	Used to indicate the match type using which you want to transfer the binder payment from the suspense contract to the on account contract of the individual member's account.
Payment Cancel Reason for Transfer	Yes	Used to indicate the payment cancel reason that you want to use while canceling the binder payment on the suspense contract. **Note:** You must specify a payment cancel reason which is already defined in the system.
Thread Pool Name	No	Used to specify the thread pool on which you want to execute the batch.

Reconciliation of Individual Health Insurance Subsidy Payments

Several states have established health insurance exchanges to offer health and dental insurance coverage options to their respective residents. Individuals within these states can compare and purchase coverage from a selection of health and dental plans, which are referred to as Qualified Health Plan (QHP).

Individuals who enroll for these QHPs may be eligible for Federal Tax Credits and Cost-Sharing Reductions (CSRs), which are subsidies based on the subscriber's income, to help pay for their health insurance and lower their out-of-pocket costs. The Centers for Medicare & Medicaid Services (CMS) makes monthly payments to insurers for the individuals who receive these credits.

CMS has established and instituted the State-Based Exchange (SBE) or Federally Facilitated Marketplace (FFM) reconciliation process to ensure that insurers enrollment information aligns with Marketplace enrollment information. The dispute process provides a mechanism for insurers to correct an FFM enrollment record or related payment information that the monthly reconciliation process cannot resolve.

Oracle Revenue Management and Billing (ORMB) may receive either the exact or different payment amount from the exchange for the sponsored charges due to various reasons. ORMB now enables you to reconcile the payments from the exchange against the bill line items for the fully insured individual business, thereby ensuring that the discrepancy is identified and resolved on time to quickly recover the money from CMS via the dispute process.

The reconciliation process in ORMB is enhanced to support both the fully insured group business and fully insured individual business. For the fully insured group business, the system stamped the policy ID and plan ID in the **C1_FT_EXT** table while freezing a financial transaction on the bill completion. These details were later used to reconcile the pay instructions against the bill line items. In this release, the **C1-STMPFTINF** algorithm type (which is attached on the **FT Freeze** system event of a customer class) is enhanced to support the fully insured individual business. It stamps the health plan code and price item in the **C1_FT_EXT** table while freezing a financial transaction on the bill completion for the fully insured individual business. These details are later used to reconcile the pay instructions received from the exchange against the bill line items of the exchange account.

A new field named **Reconciliation Category** is available while defining a reconciliation type. It helps to differentiate between the fully insured group and fully insured individual reconciliation objects. If you want to upload and process a pay instruction file for a fully insured group business using a reconciliation type, you must set the reconciliation category of the reconciliation type to **Group**. However, if you want to upload and process a pay instruction file for a fully insured individual business using a reconciliation type, you must set the reconciliation category of the reconciliation type to **Individual**.

A new algorithm type named **C1-PAYINSEXC** is introduced in this release. You need to attach an algorithm created using the **C1-PAYINSEXC** algorithm type to the **Upload Pay Instructions** system event of a reconciliation type where the reconciliation category is set to **Individual**. This algorithm parses a pay instruction file which is received from the exchange. It reads each record in the pay instruction file and creates one or more pay instructions in a reconciliation object depending on the number of exchange payment types received in the record. For example, if a record contains information about two exchange payment types (i.e. EP1 and EP2), the system will create two pay instructions for the record – one for EP1 and another for EP2. The exchange payment types (i.e. health insurance coverage) maintained in the CMS system might be different from the price items maintained in ORMB. Therefore, ORMB enables you to map an exchange payment type with a price item using the **C1-PayTypePrcItemMap** extendable lookup. Note that no values are shipped for this extendable lookup from the product. You need to create a value for the **C1-PayTypePrcItemMap** extendable lookup wherein each payment type received from the exchange is mapped to a price item. You can map a price item to one or more payment types based on the requirements.

While creating an algorithm using the **C1-PAYINSEXC** algorithm type, you need to specify the following parameters:

- **Pay Instruction Business Object** – Used to indicate the business object using which you want to create a pay instruction.
- **Date Format** – Used to indicate the format in which you want to store the coverage period start and end dates.

- **Payment Type – Price Item mapping** – Used to indicate the **C1-PayTypePrcItemMap** extendable lookup value using which you want the system to derive the price item for each payment type received from the exchange.

The system enables you to upload a pay instruction file received from the exchange in the CSV file format. You can upload pay instruction files in the CSV format from the specified location on the server using the **Pay Instruction CSV File Upload (C1-RECUP)** batch. While receiving a pay instruction file from the exchange, you need to ensure that each record in the pay instruction file contains the following information:

- Payor (i.e. exchange person) or its account from where the payment is received
- Subscriber (i.e. individual membership) for whom the payment is received
- Health plan and its coverage period for which the payment is received
- At least one exchange payment type (i.e. health insurance coverage) and its subsidy amount which is sponsored by the exchange

---

Points to Note:

The system assumes that the exchange person will have only one account. Therefore, if an exchange person has multiple accounts, the system will not allow you to upload and process a pay instruction file received from the respective exchange. Hence, if the exchange person has a single account in the system, you can either provide the payor or its account details in the record. However, if the exchange person has multiple accounts in the system, you must provide the payor's account details in the record.

You can create an exchange person and its account in the system through the health care inbound message. While creating an account for an exchange person, ensure that you set the **Eligible for Member Reconciliation (C1-RCELG)** characteristic of the account to **Y**.

You can specify either the health plan code or Health Insurance Oversight System (HIOS) ID in the record. If you are planning to use the Health Insurance Oversight System (HIOS) ID for a health plan, you need to attach an algorithm created using the **C1-STMPHLPN** algorithm type to the **Enter** system event (with the lowest sequence number) of the **Pending** status in the **C1-MemberReconciliation** business object. This algorithm will then derive the code of the health plan, to which the individual is enrolled for the coverage period, using the HIOS ID. Note that, while creating a pay instruction, the system will derive the health plan code when the HIOS ID is present in the record irrespective of whether the health plan code is present in the record or not. However, if the HIOS ID is not present in the record, the system will use the health plan code that is present in the record.

If a subscriber has enrolled for multiple insurance coverages of a health plan for the same coverage period and if the subscriber is eligible for subsidy for one or more insurance coverages, you will receive the subsidy amount for the respective insurance coverages from the CMS system. The subsidy information is received in the form of exchange payment type and amount. At a time, you can specify maximum 10 exchange payment types and their amounts in each record of a pay instruction file.

The **C1-RECUP** batch is enhanced to support the fully insured individual business. On executing the **C1-RECUP** batch, the system reads the pay instruction file from the specified location and validates it. Once a file is successfully validated, the reconciliation object is created for the file in the **Draft** status. The reconciliation object is immediately transitioned to the **Send Notification** status and the algorithms attached to the **Send Notification** status are executed. Once the To Do is created, the status of the reconciliation object is changed to **Pending**. One or more pay instructions are created for a record of a pay instruction file and its status is set to either **Pending** or **Error** depending on whether it is successfully validated or not. While creating a pay instruction, the system derives the health plan code (if not available), individual membership and the exchange account. For more information about the **C1-RECUP** batch, refer to *Oracle Revenue Management and Billing Batch Guide*.

You need to then specify the payment ID against which you want to reconcile the bill line items for which you have received the pay instructions. Once you specify the payment information and submit the pay instructions for reconciliation, the payment amount is distributed against the reconciliation contract of the exchange accounts for which you have received the pay instruction. The status of the reconciliation object is changed to **Pending Reconciliation**. On reconciling the pay instructions, the system finds the open and unmatched bill line item of the exchange account against which the pay instruction must be reconciled using the health plan, subscriber, price item, and coverage period combination. The system creates a reconciliation adjustment for a pay instruction which is successfully reconciled. While reconciling the pay instructions, the system sets the pay instruction matching level characteristic type of the reconciliation object to **MEMBER**.

You can manually reconcile the pay instructions of a reconciliation object, or you can execute the **Reconciliation Periodic Monitor (C1-RCNM)** or **Pay Instruction Processing Batch (C1-RCPM)** batch at regular intervals to reconcile the pay instructions.

> **Note:** While executing the **C1-RCPM** batch, you need to specify a member reconciliation preference whose attributes you want to use while reconciling all pay instructions of a file at the member level.

For more information about the above batches, refer to *Oracle Revenue Management and Billing Batch Guide*.

If you attach an algorithm created using the **C1-VALMRPYCN** algorithm type to the **Payment Cancellation** system event of the required customer class, the system automatically cancels the payment and reconciliation object associated with the payment when you cancel its corresponding payment tender. The status of the reconciliation object is changed to **Pending Cancelation** and the status of all pay instructions in the reconciliation object is changed to **Canceled**.

If required, you can also manually cancel a reconciliation object. However, you can cancel a reconciliation object only when it is in the **Open** or **Completed** status. On canceling a reconciliation object, the status of the reconciliation object is changed to **Pending Cancelation**. Before canceling a pay instruction, the reconciliation adjustment (if any) corresponding to the pay instruction is also canceled.

You need to configure the **Reconciliation Cancellation Periodic Monitor (C1-RCNMD)** batch to execute at the regular intervals. This batch monitors whether there are any reconciliation objects in the **Pending Cancelation** status. If there is a reconciliation object in the **Pending Cancelation** status, the status of the reconciliation object and the status of all pay instructions in the reconciliation object is changed to **Canceled**. For more information about the **C1-RCNMD** batch, refer to *Oracle Revenue Management and Billing Batch Guide*.

If all pay instructions in a reconciliation object are successfully reconciled, the system changes the status of the reconciliation object to **Completed**. However, if one or more pay instruction in a reconciliation object is not successfully reconciled, the system changes the status of the reconciliation object to **Open**. If you want the system to automatically reconcile the open pay instructions of the account when you bill the account each time, you need to attach an algorithm created using the **C1-RCLOPNRCN** algorithm type to the **Post Bill Completion** system event of the required customer class. The system will then trigger the reconciliation process to reconcile the open pay instruction against the open and unmatched bill line item of the exchange account and will accordingly update the status of the pay instruction and reconciliation object.

If you attach an algorithm created using the **C1-VLPYINACN** algorithm type to the **Adjustment Cancellation** system event of the adjustment type (using which the reconciliation adjustments are created), the system ensures that no one can cancel a reconciliation adjustment until it is linked to a non-canceled pay instruction.

Reconciliation Discrepancy Report of Individual Health Insurance Subsidy Payments

Until now, you were able to generate the reconciliation discrepancy report at the file or account level for the fully insured group business. Now, in addition, you can generate the reconciliation discrepancy report at the file or account level for the individual health insurance subsidy payments. If you generate the discrepancy report at the file level, the system lists the following:

- All pay instructions in the file which were partially reconciled due to rate variance
- All pay instructions in the file which could not be reconciled as the amount was paid, but not yet billed

However, if you generate the discrepancy report at the account level, the system lists the following:

- All bill line items which were partially reconciled against pay instructions due to rate variance
- All bill line items which could not be reconciled as the amount was billed, but not yet paid
- All pay instructions for the account which could not be reconciled as the amount was paid, but not yet billed

While generating a discrepancy report, you need to specify the discrepancy report type using which you want to create the discrepancy report. The system will create the discrepancy report in either **Pending** or **Deferred Processing** status. Once you review the discrepancy line items and finalize the discrepancy report, its status is changed to **Completed**.

At a time, you can only have one discrepancy report for a file or an account in the **Pending** or **Deferred Processing** status. Until you finalize a discrepancy report, you cannot create another discrepancy report for the respective file or account. Also, note that you can create a discrepancy report for a file only when its corresponding reconciliation object is in the **Open** status.

When you generate a discrepancy report for a pay instruction file which is received from the exchange or when you generate a discrepancy report for an exchange account, the system displays the following additional information for each discrepancy line item along with other details:

- Payor Identifier Type
- Payor Identifier
- Health Plan
- HIOS ID
- Payment Type

Note: Policy number and plan number are displayed instead of the above information when you generate discrepancy report for the fully insured group business.

Deferred Processing Mode for Discrepancy Report Creation

Until now, the reconciliation discrepancy report at the file or account level was only created in the online mode (i.e. immediately). Now, the system provides you with the ability to configure deferred processing mechanism for the reconciliation discrepancy report creation. A new status named **Deferred Processing** is introduced in the lifecycle of the **C1-DiscrepancyReport** business object. On generating a discrepancy report at the file or account level, the discrepancy report is created in the **Draft** status. A new algorithm named **C1-DISCDEF** is attached to the **Enter** system event of the **Draft** status. This algorithm checks whether the number of pay instructions in the non-final status exceeds the deferred count specified in the algorithm. If the number of pay instructions in the non-final status does not exceed the deferred count, the status of the discrepancy report is changed to **Pending**. The system then adds the discrepancy line items in the discrepancy report by comparing each non-final pay instruction against the open and unmatched bill line item of the exchange account.

However, if the number of pay instructions in the non-final status exceeds the deferred count, the status of the discrepancy report is changed to **Deferred Processing**. A new batch named **C1-DRMT** is introduced in this release. This batch is used to monitor or check whether there are any discrepancy reports in the **Deferred Processing** status. If there is a discrepancy report in the **Deferred Processing** status, the batch changes the status of the discrepancy report to **Pending**. Then, the system adds the discrepancy line items in the discrepancy report by comparing each non-final pay instruction against the open and unmatched bill line item of the exchange account.

This batch is a multi-threaded batch. The multi-threading is based on discrepancy request ID and chunks for multi-threading are created based on numerical distribution of discrepancy request ID. This batch contains the following parameters:

Parameter Name	Mandatory (Yes or No)	Description
Maintenance Object	Yes	Used to indicate that you want to monitor discrepancy reports which are created using the business objects of a particular maintenance object. **Note:** By default, the parameter value is set to **C1-DISCR-RPT**.
Restrict by Batch Code	No	Used when you want to monitor discrepancy reports whose current status is linked to the **Discrepancy Report Monitor (C1-DRMT)** batch. The valid value is **true**.
Restrict by Discrepancy Report Type	No	Used when you want to monitor the discrepancy reports which are created using a particular discrepancy report type.
Restrict by Business Object	No	Used when you want to monitor discrepancy reports which are created using a particular business object.
Restrict by Status Code	No	Used when you want to monitor discrepancy reports which are in a particular status. **Note:** This parameter is useful when this batch is invoked from more than one status in the lifecycle of the business object.
Override maximum errors	No	Used to override the maximum number of errors after which the batch must be terminated.
Threadpool	No	Used to specify the thread pool on which you want to execute the batch.

Invoice Request for Individual Health Insurance Billing

Until now, the system enabled you to generate an adhoc regular or trial bill for an account through an invoice request for the fully insured group business. Now, this feature is extended to support the adhoc billing requirements of the fully insured individual business.

The system enables you to manually or automatically create an invoice request for the fully insured individual business. It enables you to create an invoice request for an account to which an individual membership is billed. Before creating an invoice request for an account of an individual membership, the system validates whether the individual membership is eligible for the invoice request creation. You can create an invoice request only for an account of those individual memberships where the **Eligible For Invoice Request Creation (INVRACTE)** characteristic is set to **Y**. Whenever an invoice request is created for the account of an individual membership, the system adds the appropriate log entries for the individual membership in the **Log** tab of the **Membership** screen.

Note: The system validates whether the individual membership is eligible for invoice request creation using the characteristic type which is specified in the **Invoice Request Eligibility Char Type** option type of the **C1-INVREQ** feature configuration. By default, this option type is set to **INVRACTE**. You can change the characteristic type, if required.

The system enables you to create the following types of invoice request for the fully insured individual business:

- **Manual Regular Bill Generation Invoice Request** – It is used to generate an adhoc regular bill for an account. You can create a manual regular bill generation invoice request for an account from the user interface.
- **Manual Trial Bill Generation Invoice Request -** It is used to generate an adhoc trial bill for an account. You can create a manual trial bill generation invoice request for an account from the user interface.
- **Automatic Regular Bill Generation Invoice Request –** It is used to generate an adhoc regular bill for an account. You can configure the system such that an automatic regular bill generation invoice request is automatically created for an account when an individual membership is enrolled, terminated, reinstated, or canceled through a health care inbound message.

Points to Note:

The automatic invoice request creation feature is only designed to work for the fully insured group and individual businesses and not for the financial services or self-funded health insurance business.

At present, the system is not designed to create an automatic trial bill generation invoice request.

While creating an invoice request, you need to specify an invoice request type using which you want to create the invoice request. The system enables you to create the invoice request types with the following combinations:

Invoice Mode	Bill Generation Type	Approval Process
Manual	Regular	On
Manual	Regular	Off
Manual	Trial	Off
Automatic	Regular	Off

It is the invoice request type which helps the system to determine:

- The business object using which the invoice request should be created in the system

Note: If the **Bill Generation Type** field is set to **Regular**, you must select the **Invoice Request - Regular Bill Generation** business object from the **Invoice Request Business Object** list. However, if the **Bill Generation Type** field is set to **Trial**, you must select the **Invoice Request - Trial Bill Generation** business object from the **Invoice Request Business Object** list.

- Whether a manual or automatic invoice request should be created in the system
- Whether a regular or trial bill generation invoice request should be created in the system

- Whether a manual regular bill generation invoice request for an account should be processed in the online or deferred mode

Points to Note:

The system will process a manual regular bill generation invoice request for an account in the deferred mode when the number of billable charges of the account exceeds the defer processing billable charge count.

You can specify the defer processing billable charge count only when the invoice mode is set to **Manual** and bill generation type is set to **Regular**.

- Whether the approval is required for the invoice request

Note: You can set the **Approval Required** flag only when the invoice mode is set to **Manual** and bill generation type is set to **Regular**.

- The wait days using which the processing date is calculated for an invoice request

Note: If the invoice request is processed in the online mode, the system will process the invoice request when the processing date is earlier than or equal to the system date. However, if the invoice request is processed in the deferred mode, the system will process the invoice request when the processing date is earlier than or equal to the batch business date.

- The date calculation algorithm which is used to calculate the cutoff date, bill date, processing date, and accounting date for an automatic regular bill generation invoice request

Points to Note:

The **C1-INVRDTCIM** algorithm is shipped in this release for the fully insured individual business. It is used to calculate the cutoff date, bill date, processing date, and accounting date for an automatic regular bill generation invoice request. If required, you can configure the algorithm such that the processing date does not fall on a holiday. In that case, you need to specify the work calendar which helps to determine the working day. You can specify the work calendar in the **C1-INVRDTCIM** algorithm. Alternatively, if the work calendar is not specified in the **C1-INVRDTCIM** algorithm, the system will use the work calendar of the account's division to ensure that the processing date falls on a working day.

In addition, you can configure threshold days for the automatic invoice request creation. The system then determines whether the number of days between the bill cycle date and processing date is greater than the threshold days. If so, the automatic invoice request is created for the account. However, if the number of days between the bill cycle date and processing date is less than or equal to the threshold days, the automatic invoice request is created for the account such that cutoff date is calculated using the next window of the bill cycle.

While creating a manual invoice request, you need to specify the following – invoice request type, account, cutoff date, accounting date, and bill date. Note that before manually or automatically creating an invoice request for an account, you need to ensure that the following algorithms are attached to the respective system event of the corresponding customer class:

System Event	Algorithm
Invoice Request Account Selection	C1-INVACTEL
Invoice Request Status Update	C1-INVRSTUP

Manual Regular Bill Generation Invoice Request

Once you manually create a regular bill generation invoice request for an account, the status of the invoice request is set to **Draft**. You can then submit the invoice request. You can optionally configure the approval process for the manual regular bill generation invoice request. If the **Approval Required** flag is set to **Yes** in the invoice request type, the status of the manual regular bill generation invoice request is changed to **Approval In Progress** when you submit the invoice request. The approver can then approve or reject the manual regular bill generation invoice request based on the observations.

On submitting or approving the invoice request, the system checks whether the number of the billable charges of the account exceeds the defer processing billable charge count. If the number of the billable charges of the account exceeds the defer processing billable charge count, the status of the invoice request is changed to **Defer Processing**. However, if the number of the billable charges of the account does not exceed the defer processing billable charge count, the system checks whether the processing date of the invoice request is earlier than or equal to the system date.

If the processing date is earlier than or equal to the system date, the system checks whether the pending bill already exists for the account. If the pending bill does not exist for the account, the system checks whether the cutoff date is later than the bill after date. If cutoff date is later than the bill after date, the system checks whether the accounting calendar is open for the accounting date. If the accounting calendar is open for the accounting date, the system creates the adhoc regular bill for the account in the online mode. Finally, the status of the invoice request is changed to **Processed**. But, if the pending bill exists for the account, or if the cutoff date is earlier than or equal to the bill after date, or if the accounting calendar is not open for the accounting date, or if an error occurs while creating adhoc regular bill or its bill segments, the status of the invoice request is changed to **Error**. The system enables you to move such erroneous invoice request back to the **Draft** status.

However, if the processing date is later than the system date, the status of the invoice request is changed to **Defer Processing**. On the **C1-INVRQ** batch execution, the system checks whether there are any invoice requests in the **Defer Processing** status. If there is an invoice request in the **Defer Processing** status, the system checks whether the processing date of the invoice request is earlier than or equal to the system date. If the processing date is earlier than or equal to the system date, the status of the invoice request is changed to **Processing** and the algorithm attached to the **Processing** status is invoked. However, if the processing date is later than the system date, the status of the invoice request remains as **Defer Processing**.

Note: The system enables you to cancel an invoice request which is in the **Defer Processing** status.

Manual Trial Bill Generation Invoice Request

The manual regular bill generation invoice request and manual trial bill generation invoice request are created and processed identically with the following exceptions:

- On submitting the manual trial bill generation invoice request for an account, the system does not check whether the number of the billable charges of the account exceed the defer processing billable charge count. The manual trial bill generation invoice request is always processed in the deferred mode.

- You need to specify the following parameters while executing the **BILLOPEN**, **BSGENREG**, and **POSTPROC** batches:

    - **Bill Generation Type** (set to **Trial**)

    - **Description for Trial Billing Batch Run**

    - **Off Cycle Switch** (set to **Y**)

- The **BSGENREG** batch will not freeze and complete a trail bill for the account.

- At present, a To Do is not generated for a trial bill review, and therefore the system does not check whether a To Do is generated for a trial bill review.

Automatic Regular Bill Generation Invoice Request

To enable the automatic invoice request creation feature for the fully insured individual business, you need to attach the following algorithms:

If you want to create an automatic invoice request for the account while…	Then attach the algorithm….			
	Business Object	**Status**	**System Event**	**Algorithm**
Activating an individual membership	C1-IndMembership	Active	Enter	C1-INVRQIMEN
Terminating an individual membership		Terminated	Enter	C1-INVRQIMTM
Reinstating a terminated individual membership		Active	Enter	C1-INVRQIMEN
Canceling an active individual membership		Canceled	Enter	C1-INVRQIMCN

You can create a single or different invoice request type for each individual membership event (such as, activation, termination, reinstatement, or cancelation) depending on the wait days you want to configure for the respective event. While attaching the above algorithms, you need to specify the following parameters:

Algorithm	Parameter	Parameter Description
C1-INVRQIMEN	Invoice Request Type for Individual Membership New Enrollment	Used to indicate the invoice request type using which you want the system to automatically create an invoice request when an individual membership is activated. **Note:** This parameter is mandatory.
	Invoice Request Type for Individual Membership Reinstatement	Used to indicate the invoice request type using which you want the system to automatically create an invoice request when an individual membership is reinstated. **Note:** This parameter is optional.
	Termination Reason	Used to specify the termination reason. The system will automatically create an invoice request while reinstating an individual membership which was terminated due to the specified reason. **Points to Note:** This parameter is valid only when you specify the value for the **Invoice Request Type for Individual Membership Reinstatement** parameter. This parameter is optional. If you do not specify the termination reason, the system will create the invoice request on the individual membership reinstatement irrespective of the termination reason. However, if you specify the termination reason, the system will create the invoice request on the individual membership reinstatement only when the individual membership was terminated due to the specified reason. You can specify maximum five comma-separated values for this parameter. You must specify a reason which is already defined for the **Terminated** status of the **C1-IndMembership** business object in the **Status Reason** screen.

Algorithm	Parameter	Parameter Description
C1-INVRQIMTM	Invoice Request Type	Used to indicate the invoice request type using which you want the system to automatically create an invoice request when an individual membership is terminated. **Note:** This parameter is mandatory.
	Termination Reason	Used to specify the termination reason. The system will automatically create an invoice request when the individual membership is terminated due to the specified reason. **Points to Note:** This parameter is optional. If you do not specify the termination reason, the system will create the invoice request on the individual membership termination irrespective of the termination reason. However, if you specify the termination reason, the system will create the invoice request on the individual membership termination only when the individual membership is terminated due to the specified reason. You can specify maximum five comma-separated values for this parameter. You must specify a reason which is already defined for the **Terminated** status of the **C1-IndMembership** business object in the **Status Reason** screen.
C1-INVRQIMCN	Invoice Request Type	Used to indicate the invoice request type using which you want the system to automatically create an invoice request when an individual membership is canceled. **Note:** This parameter is mandatory.
	Cancelation Reason	Used to specify the cancellation reason. The system will automatically create an invoice request when the individual membership is canceled due to the specified reason.

Algorithm	Parameter	Parameter Description
		Points to Note: This parameter is optional. If you do not specify the cancellation reason, the system will create the invoice request on the individual membership cancellation irrespective of the cancellation reason. However, if you specify the cancellation reason, the system will create the invoice request on the individual membership cancellation only when the individual membership is canceled due to the specified reason. You can specify maximum five comma-separated values for this parameter. You must specify a reason which is already defined for the **Canceled** status of the **C1-IndMembership** business object in the **Status Reason** screen.

The system determines the account to which the individual membership should be billed using the Account Identifier Type and Account Identifier Value characteristics which are defined on the individual membership. While fetching the characteristics, the system considers the characteristic types which are specified in the **Account Identifier Type Char Type** and **Account Identifier Value Char Type** option types of the **C1-ASOBLLNG** feature configuration. Before automatically creating an invoice request for the account of an individual membership, the system validates whether the individual membership is eligible for the invoice request creation. If the individual membership is eligible for the invoice request creation, the system checks whether any invoice request already exists for the account in the non-final status. If an invoice request exists for the account in the non-final status, a new invoice request is not created for the account. However, if an invoice request does not exist for the account in the non-final status, a new invoice request is created for the account in the **Draft** status. The system then transitions the status of the invoice request from **Draft** to **Defer Processing Batch**.

Points to Note:

The system validates whether the individual membership is eligible for the invoice request creation only while activating an individual membership and not while terminating, reinstating, or canceling an individual membership.

The system enables you to cancel an invoice request which is in the **Defer Processing Batch** status.

When you execute the **BILLOPEN** batch with the **Off Cycle Switch** parameter set to **Y**, the system checks whether the account that meets the search criteria is included in any invoice request which is in the **Defer Processing Batch** status. If the account is included in the invoice request, the system checks whether the processing date of the invoice request is earlier than or equal to the batch business date. If the processing date is earlier than or equal to the batch business date, the system checks whether the pending bill already exists for the account. If the pending bill does not exist for the account, the system checks whether the cutoff date is later than the bill after date. If cutoff date is later than the bill after date, the system checks whether the accounting calendar is open for the accounting date. If the accounting calendar is open for the accounting date, the system creates the adhoc regular bill for the account in the **Pending** status. When you execute the **BSGENREG** batch with the **Off Cycle Switch** parameter set to **Y**, the system creates the bill segments for the adhoc regular bill. When you execute the **POSTPROC** batch with the **Off Cycle Switch** parameter set to **Y**, the system creates the post-processing bill segments (if any) and completes the adhoc regular bill of the account. Finally, the status of the invoice request is changed to **Processed**.

However, if the pending bill exists for the account, or if the cutoff date is earlier than or equal to the bill after date, or if the accounting calendar is not open for the accounting date, or if an error occurs while creating adhoc regular bill or its bill segments, or if a To Do is generated for the bill review, the status of the invoice request is changed to **Error**. The system enables you to move such erroneous invoice request back to the **Draft** status.

Delinquency Management for Individual Health Insurance Business

Most often, the fully insured individual business come across scenarios wherein individual customers fail to pay premium for their health plans on time, resulting in delinquency. Oracle Revenue Management and Billing enables you to establish a delinquency process for an account to which one or more individual memberships are billed. A delinquency process signifies a set of procedures and policies that are in place to manage overdue or unpaid insurance premiums.

The system enables you to define an account level delinquency process for the fully insured individual business. In an account level delinquency process, the system monitors the debt of the account at predefined system events and commences the delinquency process whenever the account's debt is overdue. The system refers the collection class of the account to control its collection lifecycle. The system monitors the account's debt using the bills of the account as the collection objects. The system creates a delinquency process for an account only when at least one bill of the account is overdue.

A new batch named **C1-ACDLQ** is introduced in this release. It is used to create a delinquency process for an account when its overdue bills do not meet the tolerance limit. For more information on how the delinquency process is created for an account, refer to the Delinquency Process Creation section. The system can create a delinquency process for an account only when the following pre-requisites are configured:

- Collection method of the account's collection class is set to **Self-Control Delinquency**. For more information, refer to the Collection Class section.

- A delinquency control is defined for the account's collection class. For more information, refer to the Delinquency Control section.
- The required delinquency event types and delinquency process types are defined in the system. For more information about the delinquency event type and delinquency process type, refer to the Delinquency Event Type and Delinquency Process Type sections, respectively.
- A preference of the following preference categories is defined in the system:
    o A Delinquency Process
    o Delinquency Process Type
    o Membership Repricing Reasons
    o Routing Method – Contact Method Mapping

    For more information about the preferences, refer to the Field Mapping section.
- The parameters required to create the business rule criteria are defined in the system.
- The business rules of the following categories are defined in the system:
    o Delinquency Event Attributes Business Rule
    o Delinquency Grace Period Business Rule
    o Delinquency Miscellaneous Options Business Rule
    o Delinquency Termination Date Rule Business Rule
- Required rule types and rules are defined to determine whether the delinquency event is eligible to be included in the delinquency process of the account. For more information, refer to the Delinquency Event Eligibility Evaluation and Trigger Date Calculation section.
- The outbound message types are defined for the following business objects:
    o C1-MemberTermRequest
    o C1-MemberCancRequest

Once a delinquency process is created for an account, you can perform the following actions on the delinquency process from the user interface:

- Add additional information in the form of comments or characteristics
- Manually hold a delinquency process and edit the hold details (if required)
- Manually release a hold on the delinquency process
- Manually cancel a delinquency process
- Edit the trigger date of the delinquency events in the delinquency process
- Trigger or skip the delinquency events in the delinquency process

For more information about the manual actions, refer to the Delinquency Process Maintenance section. The system also enables you to do the following:

- Automatically hold a delinquency process when the certain conditions are met
- Automatically release a hold on the delinquency process when the certain conditions are met
- Automatically cancel a delinquency process when the certain conditions are met
- Automatically resume a cancelled delinquency process when the certain conditions are met

To enable the automatic process for the above actions, you need to configure the appropriate algorithms in the delinquency process type. The system also enables you to optionally configure the approval process for various manual actions that you can perform on the delinquency event or delinquency process. For more information, refer to the Delinquency Process Approval section.

The following new screens are introduced in this release for the Delinquency Management feature:

- Delinquency Control
- Delinquency Event Type
- Delinquency Process Type
- Delinquency Process Search
- Delinquency Process

A delinquency event and process goes through various statuses in its lifecycle. The delinquency process status also accordingly changes when the individual membership status is changed to **Terminated** or **Active** (on membership reinstatement).

Types of Delinquency Processes

You can design different types of delinquency processes based on the business requirements. Some may be used to just send warning or harsh letters and/or notifications, some may be used to initiate the termination request for the individual memberships which are billed to the delinquent account, and some may be used to initiate the cancellation request for the individual memberships which are billed to the delinquent account. The system enables you to send different warning letters or To Do notifications or initiate the termination or cancellation request by triggering different delinquency events at predefined intervals in the delinquency process. The system creates a delinquency process for an account using a delinquency process type. The system determines the delinquency process type using which the delinquency process should be created for an account through the delinquency control. Note that the system considers the delinquency control that is defined for the collection class to which the account belongs. For more information about the delinquency control and delinquency process type, refer to the Delinquency Control and Delinquency Process Type sections, respectively.

The system enables you to configure the events that should be triggered in a delinquency process by adding appropriate delinquency event types in a delinquency process type. For more information about the delinquency event type, refer to the Delinquency Event Type section. The different types of events are triggered in different types of delinquency processes. The system enables you to design the following types of delinquency processes for different set of individual customers:

- A delinquency process which sends one or more To Do notifications to the main customer of the account.
- A delinquency process which sends one or more warning letters via customer contact to all the persons that are associated with the account.
- A delinquency process which sends an outbound message to the enrollment system after repetitive warning letters requesting to terminate the individual memberships billed to the delinquent account.

- A delinquency process which sends an outbound message to the enrollment system after repetitive warning letters requesting to cancel the individual memberships billed to the delinquent account.

Debt Notification Delinquency Process

There might be scenarios wherein the individual customers fail to pay premium for their health plan coverage. In such cases, you may want to send reminder notifications to the individual customers to pay their premium immediately without any delay. Oracle Revenue Management and Billing enables you to send one or more To Do notifications at regular intervals to the main customer of the delinquent account.

To create a delinquency process to send To Do notifications, you need to design the delinquency process type (using which you want to create the delinquency process) in the following manner:

Delinquency Process Type	Delinquency Event Type	System Event	Algorithm	Algorithm Purpose
DP1	DE1	Event Activation	C1-DLQCRTODO1	Send a To Do notification
	DE2	Event Activation	C1-DLQCRTODO2	Send a To Do notification
	DE3	Event Activation	C1-DLQCRTODO3	Send a To Do notification

Note that the **C1-DLQCRTODO** algorithm is shipped with the product. However, you can create different algorithms (for example, **C1-DLQCRTODO1**, **C1-DLQCRTODO2**, and **C1-DLQCRTODO3**) using the **C1-DLQCRTODO** algorithm type when you want to send different notifications at regular intervals to the main customer of the account. For more information about the **C1-DLQCRTODO** algorithm, refer to the Delinquency Event Type section.

Once a delinquency process is created to send To Do notifications, the status of the delinquency process is set to **Initiated**. Once the first event in the delinquency process is triggered, the status of the delinquency process is changed to **Delinquency In Progress**. Once a To Do is created for the main customer of the delinquent account, the status of the delinquency event is changed to **Completed**. Once all the delinquency events are triggered successfully, the status of the delinquency process is changed to **Completed**.

The system may receive a payment or an adjustment against the overdue bill for which a delinquency process is created to send To Do notifications. In such case, the system behaves in the following manner depending on the status of the delinquency process:

- If a payment or adjustment is made when a delinquency process is in the **Initiated**, **Delinquency In Progress**, or **On Hold** status, the status of the delinquency process is changed to **Canceled**. For more information on how the delinquency process is automatically canceled, refer to the Canceling a Delinquency Process section.
- If a payment or adjustment is made when a delinquency process is in the **Completed** status, no action takes place on the debt notification delinquency process.

However, if the payment or adjustment is canceled due to any reason, the system executes the **C1-CDPPAYCAN** or **C1-DPADJCAN** algorithm attached to the customer class or adjustment type, respectively, to resume the delinquency process. For more information on how the delinquency process is automatically resumed, refer to the Resuming a Delinquency Process section.

> **Note:** If a payment or adjustment is canceled when a debt notification delinquency process is in the **Completed** status, no action takes place on the debt notification delinquency process.

Debt Collection Letters Delinquency Process

There might be scenarios wherein the individual customers fail to pay premium for their health plan coverage even after sending continuous reminders through To Do notifications. In such scenarios, you may want to send them dunning or warning letters (i.e. delinquency notices). Oracle Revenue Management and Billing enables you to send one or more warning or dunning letters via customer contact to all the persons that are associated with the delinquent account.

To create a delinquency process to send warning or dunning letters, you need to design the delinquency process type (using which you want to create the delinquency process) in the following manner:

Delinquency Process Type	Delinquency Event Type	System Event	Algorithm	Algorithm Purpose
DP1	DE1	Event Activation	C1-DLQCRTODO	Send a To Do notification
	DE2	Event Activation	C1-DLQSENDCC1	Send a Warning Letter
	DE3	Event Activation	C1-DLQSENDCC2	Send a Dunning Letter
	DE4	Monitor Completed Event	C1-RCLTRGDMD	Recalculate Trigger Dates Based on Document Mail Date

The **C1-DLQSENDCC** algorithm is shipped with the product. You can create different algorithms (for example, **C1-DLQSENDCC1** and **C1-DLQSENDCC2**) using the **C1-DLQSENDCC** algorithm type when you want to send different debt collection letters at regular intervals to all the persons that are associated with the delinquent account. For more information about the **C1-DLQSENDCC** algorithm, refer to the Delinquency Event Type section.

Note that sending a To Do notification before sending debt collection letters through a delinquency process is optional. You may or may not include these To Do notification event types while designing a delinquency process type to send debt collection letters. However, you need to attach the **C1-DLQSENDCC** and **C1-RCLTRGDMD** algorithms while designing a delinquency process type to send debt collection letters. For more information about these algorithms, refer to the Delinquency Event Type section.

Once a delinquency process is created to send debt collection letters, the status of the delinquency process is set to **Initiated**. Once the first event in the delinquency process is triggered, the status of the delinquency process is changed to **Delinquency In Progress**. Once a warning or dunning letter is sent via a customer contact to all the persons associated with the delinquent account, the status of the delinquency event is changed to **Completed**. Once all the delinquency events are triggered successfully, the status of the delinquency process is changed to **Completed**.

The system may receive a payment or an adjustment against the overdue bill for which a delinquency process is created to send debt collection letters. In such case, the system behaves in the following manner depending on the status of the delinquency process:

- If a payment or adjustment is made when a delinquency process is in the **Initiated**, **Delinquency In Progress**, or **On Hold** status, the status of the delinquency process is changed to **Canceled**. For more information on how the delinquency process is automatically canceled, refer to the Canceling a Delinquency Process section.
- If a payment or adjustment is made when a delinquency process is in the **Completed** status, no action takes place on the debt collection letters delinquency process.

However, if the payment or adjustment is canceled due to any reason, the system executes the **C1-CDPPAYCAN** or **C1-DPADJCAN** algorithm attached to the customer class or adjustment type, respectively, to resume the delinquency process. For more information on how the delinquency process is automatically resumed, refer to the Resuming a Delinquency Process section.

Note: If a payment or adjustment is canceled when a debt collection letters delinquency process is in the **Completed** status, no action takes place on the debt collection letters delinquency process.

Membership Cancellation Delinquency Process

There might be scenarios wherein the individual customers fail to pay the binder payment for their health plan coverage. As a result, their individual memberships cannot be activated in the system. Such individual memberships should be canceled in the system in a timely manner. To adhere to this requirement, you can now configure the system such that a delinquency process is created for the membership's account to send a cancellation outbound message to the enrollment system when the payment for the first bill (i.e. binder payment) is not made within the due date. This enables you to initiate the membership cancellation process through the **Delinquency Management** feature. Only when an inbound message is received from the enrollment system to cancel the membership, the system changes the status of the individual membership to **Canceled**.

To create a delinquency process to send a cancellation outbound message to the enrollment system, you need to create a delinquency process type using which you want to create a delinquency process to initiate cancellation request through an outbound message. You can design the delinquency process type in the following manner:

Delinquency Process Type	Delinquency Event Type	System Event	Algorithm	Algorithm Purpose
DP1	DE1	Event Activation	C1-DLQCRTODO	Send a To Do notification

Delinquency Process Type	Delinquency Event Type	System Event	Algorithm	Algorithm Purpose
	DE2	Event Activation	C1-DLQSENDCC	Send a Warning letter
	DE3	Event Activation	C1-DLEVMBCNC	Update the status reason of the individual memberships.

For more information about these algorithms, refer to the Delinquency Event Type section. Note that sending a To Do notification and a warning or dunning letter before initiating the cancellation request through a delinquency process is optional. You may or may not include these event types while designing a delinquency process type to initiate cancellation request. However, you need to attach the **C1-DLEVMBCNC** algorithm to initiate cancellation request. This algorithm updates the status reason of the individual memberships which are billed to the delinquent account. It considers the status reason which is specified in the **Awaiting Membership Cancellation Reason** attribute of the delinquency process preference. Before updating the status reason of the individual membership, the system validates whether the status reason is defined for the **Active** status of the **C1-IndMembership** business object.

Once a delinquency process is created to initiate cancellation request for the delinquent account, the status of the delinquency process is set to **Initiated**. Once the first event in the delinquency process is triggered, the status of the delinquency process is changed to **Delinquency In Progress**. Once the status reason of the active individual memberships is updated, the status of the delinquency event and delinquency process is changed to **Completed**.

You need to then execute the **C1-MTOMT** batch to generate the cancellation outbound messages for the individual memberships. For more information about the batch, refer to the Outbound Message Generation for Delinquency Management section. Before creating the cancellation outbound messages for the individual memberships, you need to setup some pre-requisites for the cancellation outbound message generation. For more information about the pre-requisites, refer to the Membership Cancellation Outbound Message section.

Once the cancellation outbound message is sent to the enrollment system, you need to wait for the enrollment system to send the inbound message request to cancel the individual membership in ORMB. Once the inbound message to cancel an individual membership is processed, the status of the individual membership is changed to **Canceled**.

The system may receive a payment or an adjustment against the overdue bill for which a delinquency process is created to initiate the cancellation request through an outbound message. In such case, the system behaves in the following manner depending on the status of the delinquency process:

- If a payment or adjustment is made when a delinquency process is in the **Initiated**, **Delinquency In Progress**, or **On Hold** status, the status of the delinquency process is changed to **Canceled**. For more information on how the delinquency process is automatically canceled, refer to the Canceling a Delinquency Process section.
- If a payment or adjustment is made when a delinquency process is in the **Completed** status, no action takes place on the membership cancellation delinquency process.

However, if the payment or adjustment is canceled due to any reason, the system executes the **C1-CDPPAYCAN** or **C1-DPADJCAN** algorithm attached to the customer class or adjustment type, respectively, to resume the delinquency process. For more information on how the delinquency process is automatically resumed, refer to the [Resuming a Delinquency Process](#) section.

> **Note:** If a payment or adjustment is canceled when a membership cancellation delinquency process is in the **Completed** status, no action takes place on the membership cancellation delinquency process.

Membership Termination Delinquency Process

There might be scenarios wherein the individual customers fail to pay premium for their health plan coverage even after sending continuous reminders. In such scenarios, you may want to terminate their individual memberships in the system. Oracle Revenue Management and Billing enables you to initiate the membership termination process through the **Delinquency Management** feature. You can configure the system such that a delinquency process is created for the membership's account to send a termination outbound message to the enrollment system when the overdue bills do not meet the tolerance limit. Only when an inbound message is received from the enrollment system to terminate the membership, the system changes the status of the individual membership to **Terminated**.

To create a delinquency process to send a termination outbound message to the enrollment system, you need to create a delinquency process type using which you want to create a delinquency process to initiate termination request through an outbound message. You can design the delinquency process type in the following two ways:

Delinquency Process Type	Delinquency Event Type	System Event	Algorithm	Algorithm Purpose
DP1	DE1	Event Activation	C1-DLQCRTODO	Send a To Do notification
	DE2	Event Activation	C1-DLQSENDCC	Send a Warning letter
	DE3	Event Activation	C1-DETTERMDT	Calculate the termination effective date
		Event Activation	C1-DETTRTGDT	Calculate the termination request date
		Monitor Completed Event	C1-TRNTERTRG	Update the status of the delinquency process on the termination request date
DP2	DE1	Event Activation	C1-DLQCRTODO	Send a To Do notification
	DE2	Event Activation	C1-DLQSENDCC	Send a Warning letter
	DE3	Event Activation	C1-DETTERMDT	Calculate the termination effective date

Delinquency Process Type	Delinquency Event Type	System Event	Algorithm	Algorithm Purpose
	DE4	Event Activation	C1-DETTRTGDT	Calculate the termination request date
		Monitor Completed Event	C1-TRNTERTRG	Update the status of the delinquency process on the termination request date

For more information about these algorithms, refer to the [Delinquency Event Type](#) section. Note that the sending a To Do notification and a warning or dunning letter before initiating the termination request through a delinquency process is optional. You may or may not include these event types while designing a delinquency process type to initiate termination request. However, you need to attach the **C1-DETTERMDT**, **C1-DETTRTGDT**, and **C1-TRNTERTRG** algorithms to the above mentioned system events to initiate termination request.

Once a delinquency process is created to initiate termination request for the delinquent account, the status of the delinquency process is set to **Initiated**. Once the first event in the delinquency process is triggered, the status of the delinquency process is changed to **Delinquency In Progress**. After calculating the termination effective date and termination request date, the status of the respective delinquency events is changed to **Completed**. However, the status of the delinquency process changes to a status which is specified in the **Delinquency Process Pending Termination Status** attribute of the delinquency process preference only on the termination request date when the **C1-DPMON** batch is executed. Once the status of the delinquency process is changed to **Pending Termination**, the system executes the **C1-MEMTRMRSN** algorithm attached to the **Pending Termination** status in the lifecycle of the **C1-DelinquencyProcess** business object.

The **MEMTRMRSN** algorithm updates the status reason of each active individual membership which is billed to the delinquent account. It considers the status reason which is specified in the **Membership Termination Reason for Delinquency** attribute of the delinquency process preference. Before updating the status reason of the individual membership, the system validates whether the status reason is defined for the **Active** status of the **C1-IndMembership** business object. In addition, it adds an appropriate log entry in the individual membership and delinquency process and stamps the **Delinquency Process** (**C1-DELPO**) characteristic on the individual membership.

Once the status reason of the individual memberships is changed, you need to execute the **C1-MTOMT** batch to generate the termination outbound messages for the individual memberships. For more information about the batch, refer to the [Outbound Message Generation for Delinquency Management](#) section. Before creating the termination outbound messages for the individual memberships, you need to setup some pre-requisites for the termination outbound message generation. For more information about the pre-requisites, refer to the [Membership Termination Outbound Message](#) section.

Once the termination outbound message is sent to the enrollment system, you need to wait for the enrollment system to send the inbound message request to terminate the individual membership in ORMB. Once the inbound message to terminate an individual membership is processed, the status of the individual membership is changed to **Terminated**. Once the membership is terminated in ORMB, the status of the delinquency process is changed to **Customer Terminated**.

The system creates a subsequent delinquency process which sends a harsh letter via customer contact when the status of the primary delinquency process requesting to terminate the individual membership is changed to **Customer Terminated**. The main objective of this harsh letter is to communicate any penalty or legal notice on defaulting or delaying payments. In ORMB, the primary delinquency process is referred to as the related delinquency process for the subsequent delinquency process. To create a subsequent delinquency process when an individual membership is terminated in the system, you need to configure the delinquency process type using which you want to create the subsequent delinquency process in the **C1-CRDPTERM** algorithm attached to the **Customer Terminated** status of the **C1-DelinquencyProcess** business object.

The system may receive a payment or an adjustment against the overdue bill for which a delinquency process is created to initiate the termination request through an outbound message. In such case, the system behaves in the following manner depending on the status of the delinquency process:

- If a payment or adjustment is made when a delinquency process is in the **Initiated**, **Delinquency In Progress**, **On Hold**, or **Pending Termination** status, the status of the delinquency process is changed to **Canceled**. For more information on how the delinquency process is automatically canceled, refer to the [Canceling a Delinquency Process](#) section.

  However, if the payment or adjustment is canceled due to any reason, the system executes the **C1-CDPPAYCAN** or **C1-DPADJCAN** algorithm attached to the customer class or adjustment type, respectively, to resume the delinquency process. For more information on how the delinquency process is automatically resumed, refer to the [Resuming a Delinquency Process](#) section.

- If a payment or adjustment is made when a delinquency process is in the **Customer Terminated** status, you need to wait till an inbound message request is received from the enrollment system to reinstate an individual membership. Once the status of the individual membership is changed to **Active**, the system does the following:

  o Changes the status of the primary delinquency process through which the termination request was initiated to **Customer Reinstated**

  o Changes the status of the subsequent delinquency process to **Canceled** if it meets the cancel criteria

  For more information on how the delinquency process is automatically canceled, refer to the [Canceling a Delinquency Process](#) section.

  If the payment or adjustment is canceled due to any reason, the system executes the **C1-RODPDUEPX** or **C1-RODPONADX** algorithm attached to the customer class or adjustment type, respectively, to resume the delinquency process. For more information on how the delinquency process is automatically resumed, refer to the [Resuming a Delinquency Process](#) section.

Collection Class

A collection method named **Self-Control Delinquency** is available while defining a collection class. If you do not want to initiate delinquency process for an account, ensure that the account belongs to a collection class where the collection method is set to **Not Eligible for Collection**. However, if you want to initiate delinquency process for an account, ensure that the account belongs to a collection class where the collection method is set to **Self-Control Delinquency**. The **Self-Control Delinquency** collection method is used to create a separate delinquency process for the accounts which belong to the collection class.

Delinquency Control

You need to define a delinquency control for each collection class where the collection method is set to **Self-Control Delinquency**. A new maintenance object named **C1-DELIN-CTR**, a new business object named **C1-DelinControl**, and a new screen named **Delinquency Control** are introduced in this release.

The **Delinquency Control** screen enables you to create, edit, and delete a delinquency control. Note that this screen enables you to define a delinquency control only for those collection classes where the collection method is set to **Self-Control Delinquency**.

A delinquency control enables the system to determine the following:

- A minimum credit review frequency at which the accounts belonging to the respective collection class should be monitored for delinquency. In other words, it indicates in how many days the system should review whether the bills of the accounts are overdue.

- A delinquency monitor rule algorithm that you want to invoke for the collection class while executing the **C1-ACDLQ** batch.

A new algorithm type named **C1-DLNQMNRL** is introduced in this release. This algorithm creates a new delinquency process for an account. If the delinquency process already exists for the account in any of the status specified in the **Delinquency Process Status For Existing Delinquency Process** parameter, then the system will add new overdue bills in the existing delinquency process and will not create a new delinquency process for the account. While adding a bill to an existing delinquency process, the system enables you to create a To Do for the delinquency process.

The **C1-DLNQMNRL** algorithm contains the following parameters:

- **Only Process Customers with Credit Rating Less than or Equal to This Value** – Used when you want to initiate the delinquency process for only those accounts where the credit rating is less than or equal to the specified value. This parameter is optional.

- **Which Date Determines Age ('D' - Due Date or 'B' - Bill Date)** – Used to indicate whether you want the system to determine the age of the invoice (i.e. bill) using the bill date or the due date. The valid values are **D** and **B**. This parameter is mandatory.

- **Use Current Revenue Period Billed For Latest Overdue Due Date - Y/N** – Used when you want to consider all those financial transactions of the account whose coverage start date is earlier than or equal to the bill due date while calculating the total billed (i.e. original) amount. The valid values are **Y** and **N**.

This option helps to handle the scenario wherein an individual short pays by just enough each month but still does not become delinquent. This is because, in this case, the system calculates the unpaid percentage or amount based on the total amount billed for the coverage period rather than the total amount billed for the overdue bills.

This parameter is optional. If you do not specify the parameter value, by default, it is set to **N**.

- **On Account Payment Contract Types** – Used to indicate the on account contract types that you want to consider while determining the total unpaid amount. You can specify maximum twenty comma-separated values for this parameter. This parameter is only required when the **Include On Account Payments In Threshold Evaluation - Y/N** parameter is set to **Y**.

- **Include On Account Payments In Threshold Evaluation - Y/N** – Used to indicate whether you want to consider the payments made against the on account contracts of the account while calculating the unpaid amount. The valid values are **Y** and **N**.

  This parameter is optional. If you do not specify the parameter value, by default, it is set to **N**.

- **Only Process Person/Account with an Unpaid Percentage Greater Than or Equal to This Value** – Used to specify the unpaid percentage value (i.e. unpaid amount with respect to original amount). The system then considers an account as delinquent only when the unpaid percentage of its overdue bills is greater than or equal to the specified value. This parameter is required.

- **Only Process Person/Account with an Unpaid Amount Greater Than or Equal to This Value** - Used to specify the flat unpaid amount. The system then considers an account as delinquent only when the unpaid amount of its overdue bills is greater than or equal to the specified value. This parameter is required.

- **Only Process Bills with an Age Greater than or Equal to This value** – Used to specify the number of days. The system then considers an account as delinquent only when the invoice age is greater than or equal to the specified number of days. This parameter is optional.

- **Person Relationship Type Exclusion List** – This parameter is not applicable for the fully insured individual business.

- **To Do Type for To Do Entry for Individual Delinquency** – Used to indicate the To Do type using which you want to create the To Do for the delinquency process. This parameter is only required when the **To Do Entry Required For Individual Delinquency** parameter is set to **Y**.

- **Delinquency Process Status For Existing Delinquency Process** – Used to specify the status of the delinquency process. The system then checks whether any delinquency process exists for the account in the specified status. If so, the system adds a new overdue bill of the account in the respective delinquency process and does not create a new delinquency process for the account. You can specify comma-separated values for this parameter. This parameter is mandatory.

- **Postpone Date Characteristic Type** – Used to indicate a characteristic type which is used to store the postpone credit review date. Here, you must specify a characteristic type where the characteristic entity is set to **Bill**.

  You have an option to set the specified characteristic type on the bill if you want to postpone the credit review date of the bill. If the specified characteristic type is defined on the unpaid bill and if the postpone credit review date is later than the batch business date, the system will not validate the unpaid bill for delinquency. This parameter is optional.

- **Unpaid Amount and Percentage Required ('Y' or 'N')** – Used to indicate whether you want to consider both the unpaid amount and unpaid percentage to check whether the account is delinquent. The valid values are:

    - **Y** - If you set the value to **Y**, the system considers both the unpaid amount and the unpaid percentage to check whether the account is delinquent.

    - **N** - If you set the value to **N**, the system considers either the unpaid amount or the unpaid percentage to check whether the account is delinquent.

    This parameter is mandatory.

- **To Do Entry Required For Individual Delinquency** – Used to indicate whether you want the system to create a To Do when a new overdue bill is added to an existing delinquency process. The valid values are **Y** and **N**. This parameter is mandatory.

- **Delinquency Process Type Field Mapping – Individual** – Used to specify a preference where the preference category is set to **Delinquency Process Type**. The system uses the preference to determine the delinquency process type using which the delinquency process should be created in the system. This parameter is mandatory.

- **Delinquency Process Type Field Mapping – Group** – This parameter is not applicable for the fully insured individual business.

Delinquency Event Type

Once you create a delinquency control for each collection class where the collection method is set to **Self-Control Delinquency**, you need to create the required delinquency event types using which you want the system to create delinquency events in the delinquency process. If an account is delinquent, you may want to send different kinds of letters such as dunning letter, warning letter, etc., send a To Do notification, or initiate the termination or cancellation request for the individual memberships which are billed to the delinquent account. The system enables you to send different kinds of letters, send To Do notification, or initiate the termination or cancellation request by triggering different delinquency events at predefined intervals in the delinquency process.

A new maintenance object named **C1-DLQ-EV-TP**, a new business object named **C1-DelinEventType**, and a new screen named **Delinquency Event Type** are introduced in this release. The **Delinquency Event Type** screen enables you to create, edit, delete, and copy a delinquency event type. A delinquency event type enables the system to determine the following:

- The primary actions that you want to perform when the delinquency event of the respective delinquency event type is triggered manually or automatically on the trigger date. For example, a primary action can be sending a letter, sending a To Do notification, initiate the termination request, etc. You can configure the system to perform the primary actions on the **Event Activation** system event. Once the algorithms attached to the **Event Activation** system event are executed, the status of the delinquency event is changed to **Completed**.

- The secondary actions that you want to perform when the primary actions are completed on the trigger date. For example, a secondary action can be changing the delinquency process status, recalculating the trigger dates of the remaining events in the delinquency process, etc. You can configure the system to perform the secondary actions on the **Monitor Completed Event** system event. If any algorithm is attached to the **Monitor Completed Event** system event, it is executed when the status of the delinquency event is changed to **Completed**.

Note the algorithms attached to the **Event Activation** and **Monitor Completed Event** system events are invoked in the **C1-DPMON** batch.

Delinquency Event Activation

The following table lists the primary actions that you can perform using the respective algorithm shipped with the product:

Action	Algorithm	Algorithm Description
Create a To Do Notification	C1-DLQCRTODO	This algorithm creates a To Do for the main customer of the account, for which the delinquency process is created. The system also creates a log entry for each To Do notification in the delinquency process.
		This algorithm is invoked when the event is triggered manually on or before the trigger date from the user interface or automatically on the trigger date through the **C1-DPMON** batch.
		You can use this algorithm in a delinquency process type using which you want to create a delinquency process to send To Do notifications, create warning or dunning letters, or initiate cancellation or termination request through an outbound message. It should be attached to the delinquency event type which sends a To Do notification to the main customer of the account.
		Note: Once a To Do notification is created, a record is added for the individual membership and delinquency event combination in the **C1_DELIN_PROC_EVT_NOTIF** table where the **DP_EVT_NOTIF_TYPE_FLG** column is set to To Do and the **DP_EVT_NOTIF_ID** column is set to To Do Entry ID.
		This algorithm contains the following parameters:
		• **To Do Role Account** – Used to indicate that users with the specified To Do role must receive the To Do notification. This parameter is optional. If you do not specify this parameter, the system considers the default To Do role associated with the To Do type.
		• **To Do Type Account** – Used to indicate the To Do type using which you want to create a To Do. The **C1-DLQAC** To Do type is shipped with the product for the account

Action	Algorithm	Algorithm Description
		level delinquency process. This parameter is mandatory.
		Note: All other parameters (not listed here) are not applicable for the fully insured individual business.
Send Letter via Customer Contact	C1-DLQSENDCC	This algorithm creates a customer contact to send a letter (for example, dunning letter, warning letter, etc.) to each person that is associated with the delinquent account. In addition, the system does the following: • Creates a log entry for each customer contact in the delinquency process • Stamps the delinquency process ID as a characteristic using the **C1-DELPO** characteristic type on the customer contact • Stamps the account ID as a characteristic using the given characteristic type on the customer contact • Stamps the individual membership ID as a characteristic using the given characteristic type on the customer contact This algorithm is invoked when the event is triggered manually on or before the trigger date from the user interface or automatically on the trigger date through the **C1-DPMON** batch. You can use this algorithm in a delinquency process type using which you want to create a delinquency process to send debt collection letters, or initiate cancellation or termination request through an outbound message. It should be attached to the delinquency event type which sends a warning or dunning letter. **Note:** Once a customer contact is created, a record is added for the individual membership and delinquency event combination in the **C1_DELIN_PROC_EVT_NOTIF** table where the **DP_EVT_NOTIF_TYPE_FLG** column is set to Customer Contact and the **DP_EVT_NOTIF_ID** column is set to customer contact ID. This algorithm derives the preferred contact method for the customer contact. While deriving the preferred contact method, the algorithm first fetches the main customer's bill route type on the account and then fetches the bill routing method of the bill route type. Once the bill routing method is derived, the system considers the preference which is defined using the **Routing Method – Contact Method Mapping** preference category.

Action	Algorithm	Algorithm Description
		It then checks whether any preferred contact method is mapped to the bill routing method in the preference. If the preferred contact method is mapped to the bill routing method in the preference, the system stamps it on the customer contact. However, if the preferred contact method is not mapped to the bill routing method in the preference, the system considers the default contact method as the preferred contact method and stamps it on the customer contact.
		This algorithm contains the following parameters:
		• **Customer Contact Type** – Used to indicate the customer contact type using which you want to create the customer contact. This parameter is mandatory. Here, you must specify a customer contact type with the appropriate letter template depending on the type of letter you want to send to the customer.
		• **Customer Contact Class** – Used to indicate the customer contact class to which the customer contact type belongs. This parameter is mandatory.
		• **Default Contact Method** – Used to indicate the contact method that you want to use when the preferred contact method cannot be derived for the routing method from the preference. This parameter is mandatory.
		• **Account Characteristic Type** – Used to indicate the characteristic type using which you want to store the account ID on the customer contact. Here, you must specify a characteristic type where the characteristic entity is set to **Customer Contact**.
		• **Membership Characteristic Type** – Used to indicate the characteristic type using which you want to store the individual membership ID on the customer contact. Here, you must specify a characteristic type where the characteristic entity is set to **Customer Contact**.
		• **Account Relationship Type** – Used to indicate the relationship type using which the person is associated with the account. The system will then create the customer contacts only for those persons which are associated with the account using the specified relationship type. This parameter is optional. You can specify maximum ten comma-separated values for this parameter. If you do not specify the parameter value, the system will create customer contacts for all persons that are associated with the account irrespective of their relationship type.

Action	Algorithm	Algorithm Description
		Note: All other parameters (not listed here) are not applicable for the fully insured individual business.
Calculate Termination Date	C1-DETTERMDT	This algorithm is used to calculate the termination effective date. This date is sent to the enrollment system through the termination outbound message. This algorithm is invoked when the termination event is triggered manually on or before the trigger date from the user interface or automatically on the trigger date through the **C1-DPMON** batch.

It calculates the termination effective date based on the termination date rule. The system derives the termination date rule using an algorithm which is attached to the **Termination Date Rule Derivation** system event of the delinquency process type. The system retrieves the business rules of the **Delinquency Miscellaneous Options Business Rule** category which are effective on the termination event trigger date. Note that the system considers only those effective business rules which are in the **Active** status.

Once the effective business rules are identified, the system executes these business rules in the order of their priority. Once an individual membership meets the criteria defined in the business rule, the business rule is stamped against the individual membership.

Note: The system derives the individual memberships using an algorithm which is attached to the **Memberships Derivation** system event of the delinquency process type.

Once the system derives the termination date rule and the delinquency miscellaneous options business rule, the system calculates the termination effective date for each membership in the following manner:

Termination Date Rule is set to…	Parameter which is used from the business rule…	Termination Effective Date is set to…

Action	Algorithm	Algorithm Description		
		Latest Billed Coverage End Date	Add Days to Coverage End Date	Latest Billed Coverage End Date (i.e. coverage end date that falls on or prior to the grace end date) + Add Days to Coverage End Date
		Latest Due Coverage End Date	Add Days to Coverage End Date	Latest Due Coverage End Date (i.e. coverage end date that falls prior to the latest due date) + Add Days to Coverage End Date
		X days After Paid Through Date	Add Days to Paid Through Date	Paid Through Date + Add Days to Paid Through Date
		X Months After Paid Through Date	Add Months to Paid Through Date	Paid Through Date + Add Months to Paid Through Date
		Month End of Grace Period Start	-	Last Day of Grace Period Start Month
		Month End of Termination Letter Creation	-	Last Day of Termination Letter Creation Event Trigger Month
		Month End of Termination Request Event	-	Last Day of Termination Event Trigger Month
		X days After Grace End Date	Add Days to Grace End Date	Grace End Date + Add Days to Grace End Date

Action	Algorithm	Algorithm Description		
		X days After Termination Request Event	Add Days to Date of Processing of Termination Request	Termination Event Trigger Date + Add Days to Date of Processing of Termination Request

		Points to Note: The system calculates the grace start date using the trigger date calculation mode and the grace period. The system derives the grace period from the delinquency process type or using an algorithm attached to the **Grace Period Derivation** system event depending on the grace period source that is specified in the delinquency process type.
		Once the termination effective date is calculated for each membership, the system checks whether the termination effective date is earlier than the respective membership start date. If the termination effective date is earlier than the respective membership start date, the system sets the termination effective date to the respective membership start date.
Send Termination Request	C1-DETTRTGDT	This algorithm is used to calculate the termination request date. It is invoked when the termination event is triggered manually on or before the trigger date from the user interface or automatically on the trigger date through the **C1-DPMON** batch. You must use this algorithm in a delinquency process type using which you want to create a delinquency process to initiate termination request through an outbound message. It should be attached to the delinquency event type which requests to terminate the individual memberships billed to the delinquent account. **Note:** The system changes the status of the delinquency process to **Pending Termination** and updates the status reason of the individual memberships (billed to the delinquent account) on the termination request date when you execute the **C1-DPMON** batch. This algorithm calculates the termination request date based on the termination trigger process and the termination wait days.

Action	Algorithm	Algorithm Description
		The following table explains how the termination request date is calculated for each membership when the trigger termination process is set to:

Trigger Termination Process is set to…	Termination Request Date is set…
Immediate	Termination Event Trigger Date
Month End	Last Day of Termination Event Trigger Month
After X Days	Termination Event Trigger Date + Termination Wait Days

Points to Note:

The termination event trigger date is calculated when you execute the **C1-DPEVL** batch.

The system derives the value for the Trigger **Termination Process** flag and the termination wait days from the delinquency process type or using an algorithm attached to the **Termination and Reinstatement Configuration Derivation** system event depending on the termination reinstatement configuration source that is specified in the delinquency process type.

Action	Algorithm	Algorithm Description
		The system checks whether the **Trigger Termination Process** flag is set to **Immediate** for all the active individual memberships which are billed to the delinquent account. If so, the system sets the termination request date of the delinquency event to the termination event trigger date and the **Trigger Termination Process** flag of the delinquency event to **Immediate**. However, if the **Trigger Termination Process** flag is set to **Month End** or **After X Days** for any or all the active memberships, the system calculates the termination request date for each active membership and then considers the latest termination request date of the individual membership. The system sets the termination request date of the delinquency event to the latest termination request date of an active membership and sets the **Trigger Termination Process** flag of the delinquency event to the value derived for the membership with the latest termination request date.
Send Cancellation Request	C1-DLEVMBCNC	This algorithm is used to update the status reason of an individual membership when its cancellation process is initiated through a delinquency process.

Action	Algorithm	Algorithm Description
		It is invoked when the cancellation event is triggered manually on or before the trigger date from the user interface or automatically on the trigger date through the **C1-DPMON** batch.
		You must use this algorithm in a delinquency process type using which you want to create a delinquency process to initiate cancellation request through an outbound message. It should be attached to the delinquency event type which requests to cancel the individual memberships billed to the delinquent account.
		Note: The system derives the individual memberships using an algorithm which is attached to the **Memberships Derivation** system event of the delinquency process type.
		The system considers the status reason which is specified in the **Awaiting Membership Cancellation Reason** attribute of the delinquency process preference. Before updating the status reason of the individual membership, the system validates whether the status reason is defined for the **Active** status of the **C1-IndMembership** business object.

Delinquency Event Completion

The following table lists the secondary actions that you can perform using the respective algorithm shipped with the product:

Action	Algorithm	Algorithm Description
Change the Delinquency Process Status	C1-TRNTERTRG	This algorithm updates the status of the delinquency process on the termination request date. It is invoked in the **C1-DPMON** batch. The system considers the status which is specified in the **Delinquency Process Pending Termination Status** attribute of the delinquency process preference.
		You must use this algorithm in a delinquency process type using which you want to create a delinquency process to initiate termination request through an outbound message. It should be attached to the delinquency event type which requests to terminate the individual memberships billed to the delinquent account.
Recalculate Trigger Dates Based on Document Mail Date	C1-RCLTRGDMD	This algorithm fetches the customer contact which is created in the delinquency event and then retrieves its document mail date. If multiple customer contacts are created in the delinquency event, the system derives the latest document mail date. It then calculates the additional days required to deliver the document to the individual customer by finding the difference between the document mail date and the customer contact date.

		Once the additional days are derived, the system adds the additional days to the remaining events' trigger date and thereby derives the new trigger date for each remaining event in the delinquency process. If the **Consider Work Days** option is selected in the respective delinquency process type, the system ensures that the new trigger date falls on a business day and not on a holiday.
		This algorithm is invoked in the **C1-DPMON** batch. You can use this algorithm in a delinquency process type using which you want to create a delinquency process to send debt collection letters. It should be attached to the delinquency event type which sends a warning or dunning letter.
		This algorithm contains the following parameters:
		• **Document Composition Date Characteristic Type** – Used to indicate the characteristic type which stores the document mail date on the customer contact. This parameter is mandatory. Here, you must specify a characteristic type where the characteristic entity is set to **Customer Contact**.
		• **Evaluate Document Mail Date For Primary Contact Only** – Used to indicate whether the document mail date should be derived only from the delinquent account's main customer's customer contact. The valid values are **Y** and **N**. If you set the parameter value to **N**, the system derives the document mail date from all persons' customer contacts who are associated with the delinquent account and then considers the latest document mail date. This parameter is optional. If you do not specify the parameter value, by default, it is set to **N**.

Delinquency Process Type

Once you create the delinquency event types, you need to create the required delinquency process types using which you want the system to create a delinquency process for an account. You can design various delinquency process types which trigger different types of events for the different types of insurance coverages. For example, you can define different delinquency process types for the following:

- Active health plan coverages
- Inactive health plan coverages
- Medical health plan coverages
- Non-medical health plan coverages
- Subsided health plan coverages
- Non-effectuated health plan coverages

You can then configure the delinquency process type for each type of insurance coverage through the delinquency process type preference. The system considers the delinquency process type preference which is specified in the **Delinquency Process Type Field Mapping – Individual** parameter of the **C1-DLNQMNRL** algorithm (which is attached to the delinquency control).

A new maintenance object named **C1-DLQ-PRC-T**, a new business object named **C1-DelinProcType**, and a new screen named **Delinquency Process Type** are introduced in this release. The **Delinquency Process Type** screen enables you to create, edit, delete, and copy a delinquency process type. While creating a delinquency process type, you can specify the following:

- Basic details for the delinquency process type

- Details for the delinquency event trigger date calculation

- Types of delinquency events that you want to trigger in a delinquency process which is created using the delinquency process type

- Algorithms that you want to attach to the delinquency process type

- Details for the grace period calculation

- Details for the termination request (outbound message) trigger process

- Details for the termination effective date calculation

- Details for the trigger date recalculation when a delinquency process which is on hold is released and when a delinquency process which is canceled is resumed

- Actions for which you want to configure the approval process

Delinquency Process Type Details

While creating a delinquency process type, you need to specify the following details:

- The entity type (for example, **Account** or **Person**) for which the delinquency process type is applicable. Note that only the account level delinquency process is applicable for the fully insured individual business.

- The business object using which the delinquency process should be created in the system.

> **Note:** A new maintenance object named **C1-DLN-PROC** and a new business object named **C1-DelinquencyProcess** are introduced in this release.

- The type of object (for example, bill) for which you want to manage collections through the delinquency process. Note that the system enables you select only those objects as the collection objects for which the **Eligible for Delinquency (Y/N)** option type is selected in the respective maintenance object. At present, the **Eligible for Delinquency (Y/N)** option type is only selected for the **BILL** maintenance object.

- Whether the termination process should be automatically initiated for the individual memberships billed to the account for which the delinquency process is created. Note that the automatic reinstatement of individual memberships is not applicable for the fully insured individual business.

- The way in which you want to trigger the delinquency events in the delinquency process. The system supports the following ways:
  - **Automatic** - Used when you want the system to automatically trigger delinquency events on the trigger date. The trigger dates are calculated for the delinquency events when you execute the **C1-DPEVL** batch. The trigger dates of the delinquency events are calculated using the given trigger date calculation details, such as trigger date calculation mode, wait period, etc.

- o **Manual** - Used when you want to manually trigger delinquency events in the delinquency process at the desired interval. In this case, the trigger dates are not calculated for the delinquency events. The **Trigger** icon appears corresponding to a delinquency event in the **Delinquency Process** screen. You can trigger a delinquency event by clicking the **Trigger** icon corresponding to the delinquency event in any order irrespective of its sequence in the delinquency process.

- o **Manual or Automatic** – Used when you want the system to automatically trigger delinquency events on the trigger date. The trigger dates are calculated for the delinquency events when you execute the **C1-DPEVL** batch. The trigger dates of the delinquency events are calculated using the given trigger date calculation details, such as trigger date calculation mode, wait period, etc. Here, you also have an option to manually trigger a delinquency event in the delinquency process whenever required before the trigger date in order to the expedite the collections process. The **Trigger** icon appears corresponding to a delinquency event in the **Delinquency Process** screen. You can trigger a delinquency event by clicking the **Trigger** icon corresponding to the delinquency event. Ideally, you must trigger a delinquency event in the ascending order of its sequence. However, if you manually trigger a delinquency event in any order, all the delinquency events in the **Pending** status before the manually triggered delinquency event will be skipped in the delinquency process. The status of such delinquency events is changed to **Skipped**.

Delinquency Event Trigger Date Calculation Details

While creating a delinquency process type, you need to specify the following details:

- Whether you want to delay a delinquency event for a specific period that can be specified either in days or months

- Whether the trigger date of a delinquency event should fall on a business day

- The method in which you want to calculate the trigger date for the delinquency events. The system supports the following methods:

- o **Latest Bill Due Date with Grace Period** – Used when you want to derive trigger date using the latest bill due date and grace period. The system considers the latest bill due date as the grace start date. It then derives the grace end date using the grace start date and grace period (in days). Once the grace start and end dates are derived, the system calculates the trigger date based on the trigger period and wait period configuration of the delinquency event type as shown below:

Trigger Period	Wait Period	Wait Days	Wait Months	Day of Month	Trigger Month is set to…	Trigger Date is set to…
After Grace End Date	Monthly Cyclic	-	1	5	Grace End Date + 1 Month	5th of Trigger Month
After Grace End Date	Wait Days	10	-	-	-	Grace End Date + 10 days

Trigger Period	Wait Period	Wait Days	Wait Months	Day of Month	Trigger Month is set to…	Trigger Date is set to…
Before Grace End Date	Monthly Cyclic	-	1	7	Grace End Date - 1 Month	7th of Trigger Month
Before Grace End Date	Wait Days	5	-	-	-	Grace End Date - 5 days

If the wait period is set to **Monthly Cyclic**, you need to specify the wait months and the day of the month for each delinquency event type in the delinquency process type. However, if the wait period is set to **Wait Days**, you need to specify the wait days for each delinquency event type in the delinquency process type.

Points to Note:

The system derives the wait days or wait month and the day of the month from the delinquency process type or using an algorithm attached to the **Wait Days Derivation** system event depending on the wait period source which is specified in the delinquency process type.

The system derives the grace period from the delinquency process type or using an algorithm attached to the **Grace Period Derivation** system event depending on the grace period source which is specified in the delinquency process type.

If the calculated trigger date of an event is past dated, the system enables you to move the trigger date of the event to the next month. If the new trigger date is also past dated, the system keeps moving the trigger date by one month until it is a future date. You can avail this functionality for the delinquency events by selecting the **Move Past Dated Event To Future** option corresponding to the respective delinquency event type in the delinquency process type. Note that if the trigger date of an event is moved to a future date, the trigger date of the subsequent events in the delinquency process is also moved accordingly. This is because the system is designed to trigger only one event at a time in a month.

Note: You can enable the **Move Past Dated Event To Future** option only when the wait period is set to **Monthly Cyclic**.

o **Paid Through Date** – Used when you want to derive trigger date using the paid through date of the account and the wait period. The system calculates the trigger date based on the wait period configuration of the delinquency event type as shown below:

Wait Period	Wait Days	Wait Months	Day of Month	Trigger Month is set to…	Trigger Date is set to…
Monthly Cyclic	-	1	5	Paid Through Date + 1 Month	5th of Trigger Month
Wait Days	5	-	-	-	Paid Through Date + 5 days

If the wait period is set to **Monthly Cyclic**, you need to specify the wait months and the day of the month for each delinquency event type in the delinquency process type. However, if the wait period is set to **Wait Days**, you need to specify the wait days for each delinquency event type in the delinquency process type.

Note: The system derives the wait days or wait month and the day of the month from the delinquency process type or using an algorithm attached to the **Wait Days Derivation** system event depending on the wait period source which is specified in the delinquency process type.

If the calculated trigger date of an event is past dated, the system enables you to move the trigger date of the event to the next month. If the new trigger date is also past dated, the system keeps moving the trigger date by one month until it is a future date. You can avail this functionality for the delinquency events by selecting the **Move Past Dated Event To Future** option corresponding to the respective delinquency event type in the delinquency process type. Note that if the trigger date of an event is moved to a future date, the trigger date of the subsequent events in the delinquency process is also moved accordingly. This is because the system is designed to trigger only one event at a time in a month.

Note: You can enable the **Move Past Dated Event To Future** option only when the wait period is set to **Monthly Cyclic**.

- o **Usage Basis** – Used when you want to derive trigger date using either of the following dates:

  - **Delinquency Process Creation Date** – Used when you want to derive trigger date using the delinquency process creation date and the wait period. The system calculates the trigger date based on the wait period configuration of the delinquency event type as shown below:

Wait Period	Wait Days	Wait Months	Day of Month	Trigger Month is set to…	Trigger Date is set to…
Monthly Cyclic	-	1	5	Delinquency Process Creation Date + 1 Month	5th of Trigger Month
Wait Days	15	-	-	-	Delinquency Process Creation Date + 15 days

If the wait period is set to **Monthly Cyclic**, you need to specify the wait months and the day of the month for each delinquency event type in the delinquency process type. However, if the wait period is set to **Wait Days**, you need to specify the wait days for each delinquency event type in the delinquency process type.

> **Note:** The system derives the wait days or wait month and the day of the month from the delinquency process type or using an algorithm attached to the **Wait Days Derivation** system event depending on the wait period source which is specified in the delinquency process type.

If the calculated trigger date of an event is past dated, the system enables you to move the trigger date of the event to the next month. If the new trigger date is also past dated, the system keeps moving the trigger date by one month until it is a future date. You can avail this functionality for the delinquency events by selecting the **Move Past Dated Event To Future** option corresponding to the respective delinquency event type in the delinquency process type. Note that if the trigger date of an event is moved to a future date, the trigger date of the subsequent events in the delinquency process is also moved accordingly. This is because the system is designed to trigger only one event at a time in a month.

> **Note:** You can enable the **Move Past Dated Event To Future** option only when the wait period is set to **Monthly Cyclic**.

- **Grace Start Date** - Used when you want to derive trigger date using the grace start date and the wait period. You can set the grace start date to either of the following - **Delinquency Process Creation Date**, **Latest Bill Due Date**, or **X Days after Latest Bill Due Date**. The system calculates the trigger date based on the wait period configuration of the delinquency event type as shown below:

Grace Start Date Basis	Wait Period	Wait Days	Wait Months	Day of Month	Add Days to Due Date	Trigger Month is set to…	Trigger Date is set to…
Delinquency Process Creation Date	Monthly Cyclic	-	1	5	-	Grace Start Date (i.e. Delinquency Process Creation Date) + 1 Month	5th of Trigger Month
Delinquency Process Creation Date	Wait Days	15	-	-	-	-	Grace Start Date (i.e. Delinquency Process Creation Date) + 15 days
Latest Bill Due Date	Monthly Cyclic	-	2	7	-	Grace Start Date (i.e. Latest Bill Due Date) + 2 Month	7th of Trigger Month

Grace Start Date Basis	Wait Period	Wait Days	Wait Months	Day of Month	Add Days to Due Date	Trigger Month is set to…	Trigger Date is set to…
Latest Bill Due Date	Wait Days	20	-	-	-	-	Grace Start Date (i.e. Latest Bill Due Date) + 20 days
X Days after Latest Bill Due Date	Monthly Cyclic	-	1	10	10	Grace Start Date (i.e. Latest Bill Due Date + 10 days) + 1 Month	10th of Trigger Month
X Days after Latest Bill Due Date	Wait Days	20	-	-	10	-	Grace Start Date (i.e. Latest Bill Due Date + 10 days) + 20 days

If the wait period is set to **Monthly Cyclic**, you need to specify the wait months and the day of the month for each delinquency event type in the delinquency process type. However, if the wait period is set to **Wait Days**, you need to specify the wait days for each delinquency event type in the delinquency process type.

Note: The system derives the wait days or wait month and the day of the month from the delinquency process type or using an algorithm attached to the **Wait Days Derivation** system event depending on the wait period source which is specified in the delinquency process type.

If the calculated trigger date of an event is past dated, the system enables you to move the trigger date of the event to the next month. If the new trigger date is also past dated, the system keeps moving the trigger date by one month until it is a future date. You can avail this functionality for the delinquency events by selecting the **Move Past Dated Event To Future** option corresponding to the respective delinquency event type in the delinquency process type. Note that if the trigger date of an event is moved to a future date, the trigger date of the subsequent events in the delinquency process is also moved accordingly. This is because the system is designed to trigger only one event at a time in a month.

Note: You can enable the **Move Past Dated Event To Future** option only when the wait period is set to **Monthly Cyclic**.

- **Latest Bill Date** - Used when you want to derive trigger date using the latest bill date and the wait period. The system calculates the trigger date based on the wait period configuration of the delinquency event type as shown below:

Wait Period	Wait Days	Wait Months	Day of Month	Trigger Month is set to…	Trigger Date is set to…
Monthly Cyclic	-	1	5	Latest Bill Date + 1 Month	5th of Trigger Month
Wait Days	15	-	-	-	Latest Bill Date + 15 days

If the wait period is set to **Monthly Cyclic**, you need to specify the wait months and the day of the month for each delinquency event type in the delinquency process type. However, if the wait period is set to **Wait Days**, you need to specify the wait days for each delinquency event type in the delinquency process type.

Note: The system derives the wait days or wait month and the day of the month from the delinquency process type or using an algorithm attached to the **Wait Days Derivation** system event depending on the wait period source which is specified in the delinquency process type.

If the calculated trigger date of an event is past dated, the system enables you to move the trigger date of the event to the next month. If the new trigger date is also past dated, the system keeps moving the trigger date by one month until it is a future date. You can avail this functionality for the delinquency events by selecting the **Move Past Dated Event To Future** option corresponding to the respective delinquency event type in the delinquency process type. Note that if the trigger date of an event is moved to a future date, the trigger date of the subsequent events in the delinquency process is also moved accordingly. This is because the system is designed to trigger only one event at a time in a month.

Note: You can enable the Move Past Dated Event To Future option only when the wait period is set to **Monthly Cyclic**.

- **Latest Bill Due Date** – Used when you want to derive trigger date using the latest bill due date and the wait period. The system calculates the trigger date based on the wait period configuration of the delinquency event type as shown below:

Wait Period	Wait Days	Wait Months	Day of Month	Trigger Month is set to…	Trigger Date is set to…
Monthly Cyclic	-	2	15	Latest Bill Due Date + 2 Month	15th of Trigger Month
Wait Days	25	-	-	-	Latest Bill Due Date + 25 days

If the wait period is set to **Monthly Cyclic**, you need to specify the wait months and the day of the month for each delinquency event type in the delinquency process type. However, if the wait period is set to **Wait Days**, you need to specify the wait days for each delinquency event type in the delinquency process type.

> **Note:** The system derives the wait days or wait month and the day of the month from the delinquency process type or using an algorithm attached to the **Wait Days Derivation** system event depending on the wait period source which is specified in the delinquency process type.

If the calculated trigger date of an event is past dated, the system enables you to move the trigger date of the event to the next month. If the new trigger date is also past dated, the system keeps moving the trigger date by one month until it is a future date. You can avail this functionality for the delinquency events by selecting the **Move Past Dated Event To Future** option corresponding to the respective delinquency event type in the delinquency process type. Note that if the trigger date of an event is moved to a future date, the trigger date of the subsequent events in the delinquency process is also moved accordingly. This is because the system is designed to trigger only one event at a time in a month.

> **Note:** You can enable the **Move Past Dated Event To Future** option only when the wait period is set to **Monthly Cyclic**.

Delinquency Event Types

While adding the delinquency event types in a delinquency process type, you need to specify the following:

- The delinquency event types using which you want to create the delinquency events in a delinquency process which is created using the delinquency process type
- The order in which you want to create the delinquency event using the delinquency event type
- The time period (i.e. either before or after grace end date) during which you want to trigger the delinquency events when the trigger date calculation mode is set to **Latest Bill Due Date with Grace Period**
- Whether you want the system to derive the wait days, wait months, and day of month during the trigger date calculation from the delinquency process type or through an algorithm
- The number of days for which you want to delay the delinquency event in a delinquency process
- The number of months for which you want to delay the delinquency event in a delinquency process
- The day of the month when you want to trigger the delinquency event of a delinquency process
- Whether you want the system to recalculate the trigger date of a delinquency event if it is a past dated until it is a future date
- Whether you want to preview a delinquency event which helps the relationship manager to inform the customer in advance that he will receive the To Do notification or customer contact within the specified number of days

- The number of days before the trigger date when you want to schedule a preview of a delinquency event
- Whether you want the system to skip the delinquency event in a delinquency process which is past dated
- Whether you want the system to check whether a delinquency event type is eligible for an account and then accordingly create the delinquency event using the delinquency event type

---

Points To Note:

The list includes only those rule types where the rule type usage is set to **Delinquency Event Type Eligibility**.

The membership level notification is not applicable for the fully insured individual business.

Delinquency Process Type Algorithms

While creating a delinquency process type, you need to attach the following algorithms to the respective system event:

System Event	Algorithm	Algorithm Description
Calculate Unpaid and Original Amounts	C1-CUAOAB	This algorithm derives the unpaid and original amounts of each overdue bill for which the delinquency process is created. It invokes the algorithm attached to the **Determine Open Item Bill Amounts** system event in the installation options to calculate the unpaid amount for a bill. **Note:** This algorithm is required when the collection object in the delinquency process type is set to **Bill**.
Cancel Criteria	C1-CANDELPRC	This algorithm cancels the delinquency process when either of the following condition is met: • Payments or adjustments are made against the overdue bills, for which the delinquency process is created, and as a result the unpaid balance of the overdue bills is within the tolerance limit • Collection class of the account, for which the delinquency process is created, is changed • Voluntary termination is received for an individual membership (which is billed to the delinquent account) with a termination date that is earlier than or equal to the calculated termination date • Voluntary cancellation is received for an individual membership (which is billed to the delinquent account) It contains the following parameters:

System Event	Algorithm	Algorithm Description

		• **Tolerance Percentage** - Used to specify the unpaid percentage value (i.e. unpaid amount with respect to original amount). The system cancels the delinquency process when the unpaid percentage of the overdue bills is within the tolerance limit. This parameter is required.
		
		• **Tolerance Amount** - Used to specify the flat unpaid amount. The system cancels the delinquency process when the unpaid amount of the overdue bills is within the tolerance limit. This parameter is required.
		
		• **Tolerance Amount and Percentage Required (Y / N)** - Used to indicate whether you want to consider both the unpaid amount and unpaid percentage to check whether the delinquency process should be canceled or not. The valid values are:
		
		    ◦ **Y** - If you set the value to **Y**, the system considers both the unpaid amount and the unpaid percentage to check whether the delinquency process should be canceled or not.
		
		    ◦ **N** - If you set the value to **N**, the system considers either the unpaid amount or the unpaid percentage to check whether the delinquency process should be canceled or not.
		
		This parameter is mandatory.
		
		• **Include On Account Payments In Threshold Evaluation - (Y/N)** - Used to indicate whether you want to consider the payments made against the on account contracts of the account while calculating the unpaid amount. The valid values are **Y** and **N**.
		
		• **On Account Payment Contract Type (Comma separated valid values)** - Used to indicate the on account contract types that you want to consider while determining the total unpaid amount. You can specify maximum twenty comma-separated values for this parameter.

System Event	Algorithm	Algorithm Description

		This parameter is only required when the **Include On Account Payments In Threshold Evaluation - (Y/N)** parameter is set to **Y**.
		
		• **Use Current Revenue Period Billed For Latest Overdue Due Date In Threshold Evaluation - (Y/N)** – Used to indicate whether the system should calculate the unpaid percentage or amount based on the total amount billed for the coverage period or based on the total amount billed for the overdue bills. The valid values are:
		
		    ○ **Y** – Used when you want to **c**alculate the unpaid percentage or amount based on the total amount billed for the coverage period.
		
		    ○ **N -** Used when you want to calculate the unpaid percentage or amount based on the total amount billed for the overdue bills.
		
		This parameter is optional. If you do not specify the parameter value, by default, it is set to **N**.
		
		• **Which Date Determines Age ('D' - Due Date or 'B' - Bill Date)** - Used to indicate whether you want the system to determine the age of the invoice (i.e. bill) using the bill date or the due date. The valid values are **D** and **B**. This parameter is mandatory.
		
		• **Only Process Bills with an Age >= This Value** - Used to specify the number of days. The system considers an overdue bill of the delinquency process only when the invoice age is greater than or equal to the specified number of days. This parameter is optional.
		
		• **Delinquency Process Cancel Reason for Unpaid Amount Reduction** – Used to indicate the cancel reason that you want to use while canceling a delinquency process when the unpaid amount and/or percentage is within the tolerance limit. Here, you must specify a reason which is defined for the **Canceled** status of the **C1-DelinquencyProcess** business object in the **Status Reason** screen.

System Event	Algorithm	Algorithm Description
		• **Delinquency Process Cancel Reason for Collection Class Change** - Used to indicate the cancel reason that you want to use while canceling a delinquency process when the collection class of the account is changed. Here, you must specify a reason which is defined for the **Canceled** status of the **C1-DelinquencyProcess** business object in the **Status Reason** screen. • **Delinquency Process Cancel Reason for Voluntary Termination** – Used to indicate the cancel reason that you want to use while canceling a delinquency process when an individual membership billed to the account is voluntarily terminated by the customer on or before the calculated termination date. Here, you must specify a reason which is defined for the **Canceled** status of the **C1-DelinquencyProcess** business object in the **Status Reason** screen. • **Delinquency Process Cancel Reason for Membership Cancelation** - Used to indicate the cancel reason that you want to use while canceling a delinquency process when an individual membership billed to the account is voluntarily canceled by the customer. Here, you must specify a reason which is defined for the **Canceled** status of the **C1-DelinquencyProcess** business object in the **Status Reason** screen. • **Evaluate Customer Status when all Events are Completed** – Used when you want the system to evaluate whether an individual membership billed to the delinquent account is voluntarily terminated or canceled by the customer even when all the events in the delinquency process are in the **Completed** status. The valid values are **Y** and **N**. This parameter is mandatory. If you set this parameter value to **N**, the system will evaluate whether an individual membership billed to the delinquent account is voluntarily terminated or canceled by the customer until all the events in the delinquency process are in the **Completed** status and not after all the events in the delinquency process are in the **Completed** status.

System Event	Algorithm	Algorithm Description
		Note: The cancellation criteria only consider the bills that were due on or before the delinquency process was created. This is because the grace period is determined based on the latest due date. From a business perspective, the system should ensure that a customer is given sufficient grace period for the unpaid invoices.
Grace Period Derivation	C1-DERGRBR	This algorithm is required when the **Grace Period Source** flag in the delinquency process type is set to **Algorithm**. It retrieves the business rules of the **Delinquency Grace Period Business Rule** category which are effective on the batch business date while executing the **C1-DPEVL** batch. Note that the system considers only those effective business rules which are in the **Active** status.
		Once the effective business rules are identified, the system executes these business rules in the order of their priority. Once an individual membership meets the criteria defined in the business rule, the business rule is stamped against the individual membership. The system then derives the **Grace Period (Days)** flag from the business rule.
		The system then checks the grace period of each active individual membership and sets the grace period of the delinquency event to the value derived for the membership with the maximum grace days.
		Note: The system derives the individual memberships using an algorithm which is attached to the **Memberships Derivation** system event of the delinquency process type.
Hold Delinquency Process Criteria	C1-HLDDELPRC	This algorithm holds a delinquency process when any of the following conditions are met: • Postpone credit review date stamped on the delinquent account is later than the system date • One or more promise to pay are active for the delinquent account • A payment agreement request is active for the delinquent account It is invoked by the **C1-DPMON** batch.

System Event	Algorithm	Algorithm Description

		If the postpone credit review date of the account is later than the system date, and/or if an active payment agreement request exists for the delinquent account, and/or if one or more promise to pay are active for the delinquent account, the system determines the hold end date. While determining the hold end date, it considers the date which is the latest of the following:
		
		 • Postpone credit review date
		
		 • Latest schedule date specified in the payment agreement request
		
		 • Latest schedule date specified among all the active promise to pay
		
		 • Hold end date manually specified in the delinquency process
		
		 If the derived hold end date is later than the system date, then the system does the following:
		
		 • Sets the status of the delinquency process to the value specified in the **Delinquency Process Hold Status** attribute of the preference where the preference category is set to **Delinquency Process**.
		
		 • Sets the hold start date to a date which is earliest of the following:
		
		    o If the postpone credit review date of the account is later than the system date, the system uses the processing date as the hold start date.
		
		    o If an active payment agreement request exists for the delinquent account, the system uses the date when the payment agreement request was set to the **Active** status as the hold start date.
		
		    o If one or more promise to pay are active for the delinquent account, the system uses the earliest schedule date among all the active promise to pay as the hold start date.
		
		 • Sets the hold reason depending on whether the system considers either of the following as the hold end date:
		
		    o Postpone credit review date

System Event	Algorithm	Algorithm Description

		o Latest schedule date specified in the payment agreement request
		
		   o Latest schedule date specified among all the active promise to pay
		
		   o Hold end date manually specified in the delinquency process
		
		 • Sets the mode of hold for the delinquency process to **Systematic**
		
		This algorithm contains the following parameters:
		
		 • **Hold Request Reason** – Used to indicate the reason that you want to use while holding a delinquency process. The system considers this reason when the hold end date is to set to the postpone credit review date of the account. Here, you must specify the reason which is defined for the **On Hold** status of the **C1-DelinquencyProcess** business object.
		
		 • **Promise To Pay Reason** - Used to indicate the reason that you want to use while holding a delinquency process. The system considers this reason when the hold end date is set to the latest schedule date specified in the payment agreement request. Here, you must specify the reason which is defined for the **On Hold** status of the **C1-DelinquencyProcess** business object.
		
		 • **Payment Agreement Reason** - Used to indicate the reason that you want to use while holding a delinquency process. The system considers this reason when the hold end date is set to the latest schedule date specified among all the active promise to pay. Here, you must specify the reason which is defined for the **On Hold** status of the **C1-DelinquencyProcess** business object.
		
		All the parameters are mandatory.

System Event	Algorithm	Algorithm Description
Memberships Derivation	C1-MEMBDLQ	This algorithm fetches those individual memberships where details of the account, for which the delinquency process is created, are stored as the characteristics on the individual memberships. While fetching the individual memberships which are billed to delinquent account, the system considers the characteristic types which are specified in the **Account Identifier Type Char Type** and **Account Identifier Type Char Value** option types of the **C1-ASOBLLNG** feature configuration.
		Once the individual memberships are derived, the system considers only those memberships where the status is set to the value specified in the **Membership Active Status** option type of the **C1-ASOBLLNG** feature configuration.
		This algorithm then derives a preference where the preference category is set to **Individual Membership**. It checks whether the characteristic type specified in the **Active Selection Characteristics** attribute of the preference is defined on any of the derived active individual memberships.
		Note: If a characteristic of the specified characteristic type exists on an individual membership, it means that the individual membership is created for the next year's coverage. However, if a characteristic of the specified characteristic type does not exist on an individual membership, it means that the individual membership is created for the current year's coverage.
		If the account selection characteristic exists on the active individual membership, the system checks whether the characteristic's effective date is later than or equal to the membership start date. If the characteristic's effective date is later than or equal to the membership start date, the system checks whether the paid through date is stamped on the account. If the paid through date is stamped on the account, the system checks whether the membership start date is later than the paid through date. If the membership start date is later than the paid through date, the system derives the following:

System Event	Algorithm	Algorithm Description

		• Sum of the following payments
		
		    o Payment made against the on account contracts
		
		    o Payments made against the coverage period that falls after the paid through date
		
		• Premium for the first coverage period of the individual membership (which is created for the next year's coverage)
		
		The system then compares the premium of the first coverage period against the sum of payments. If the payment is sufficient to pay the first coverage period premium, the system sets the **Evaluate Guaranteed Availability** flag of the individual membership to **true**.
		
		However, if the account selection characteristic is not defined on the individual membership or if the account selection characteristic's effective date is earlier than the membership start date or if the paid through date is not stamped on the account or if the individual membership start date is earlier than or equal to the paid through date, or if the payment is not sufficient to pay the first coverage premium for the next year's membership, then the system sets the **Evaluate Guaranteed Availability** flag of the individual membership to **false**.
		
		This algorithm contains the following parameter:
		
		• **On Account Payment Contract Types** – Used to indicate the contract types whose contracts you want to consider while deriving the on account payments. This parameter is required.

System Event	Algorithm	Algorithm Description
Release Delinquency Process Criteria	C1-RELDELPRC	This algorithm monitors a delinquency process which is on hold and determines whether the hold end date of the delinquency process should be updated or whether the hold on the delinquency process should be released. It is invoked by the **C1-DPMON** batch. If the postpone credit review date of the account is earlier than or equal to the system date, and if an active payment agreement request does not exist for the delinquent account, and if none of the promise to pay are active for the delinquent account, the system determines whether the hold end date is manually updated by the user. If the hold end date is manually updated by the user, the system sets the mode of hold for the delinquency process to **Manual**. However, if the hold end date is not updated manually by the user or if the manually specified hold end date is earlier than or equal to the system date, the system sets the status of the delinquency process to the value specified in the **Delinquency Process Released Status** attribute of the preference where the preference category is set to **Delinquency Process**. If the postpone credit review date of the account is later than the system date, and/or if an active payment agreement request exists for the delinquent account, and/or if one or more promise to pay are active for the delinquent account, the system determines the hold end date. While determining the hold end date, it considers the date which is the latest of the following: • Postpone credit review date • Latest schedule date specified in the payment agreement request • Latest schedule date specified among all the active promise to pay • Hold end date manually specified in the delinquency process If the derived hold end date is later than the system date and if the derived hold end date is later than the existing hold end date, then the existing hold end date is updated to the derived hold end date. The hold reason is also updated depending on whether the system considers either of the following as the hold end date: • Postpone credit review date • Latest schedule date specified in the payment agreement request

System Event	Algorithm	Algorithm Description
		• Latest schedule date specified among all the active promise to pay
		In addition, the system sets the mode of hold for the delinquency process to **Systematic**.
		This algorithm contains the following parameters:
		• **Hold Request Reason** – Used to indicate the reason that you want to use while holding a delinquency process. The system considers this reason when the hold end date is to set to the postpone credit review date of the account. Here, you must specify the reason which is defined for the **On Hold** status of the **C1-DelinquencyProcess** business object.
		• **Promise To Pay Reason** - Used to indicate the reason that you want to use while holding a delinquency process. The system considers this reason when the hold end date is set to the latest schedule date specified in the payment agreement request. Here, you must specify the reason which is defined for the **On Hold** status of the **C1-DelinquencyProcess** business object.
		• **Payment Agreement Reason** - Used to indicate the reason that you want to use while holding a delinquency process. The system considers this reason when the hold end date is set to the latest schedule date specified among all the active promise to pay. Here, you must specify the reason which is defined for the **On Hold** status of the **C1-DelinquencyProcess** business object.
		All the parameters are mandatory.
Termination Date Rule Derivation	C1-DERTRBR	This algorithm retrieves the business rules of the **Delinquency Termination Date Rule Business Rule** category which are effective on the batch business date while executing the **C1-DPMON** batch. Note that the system considers only those effective business rules which are in the **Active** status.
		Once the effective business rules are identified, the system executes these business rules in the order of their priority.
		Once an individual membership meets the criteria defined in the business rule, the business rule is stamped against the individual membership. The system then derives the value for the **Termination Date Rule** flag from the business rule.

System Event	Algorithm	Algorithm Description
		For each active individual membership which is billed to the delinquent account, the system checks the value of the **Termination Date Rule** flag. It then considers the termination date rule with the highest priority (defined in the respective delinquency process type) and sets it against the delinquency event.
		Note: The system derives the individual memberships using an algorithm which is attached to the **Memberships Derivation** system event of the delinquency process type.
Termination and Reinstatement Configuration Derivation	C1-DERTROBR	This algorithm is invoked by the **C1-DETTRTGDT** algorithm when the **Termination Reinstatement Configuration Source** flag in the delinquency process type is set to **Algorithm**. It retrieves the business rules of the **Delinquency Miscellaneous Options Business Rule** category which are effective on the batch business date. Note that the system considers only those effective business rules which are in the **Active** status. Once the effective business rules are identified, the system executes these business rules in the order of their priority. Once an individual membership meets the criteria defined in the business rule, the business rule is stamped against the individual membership. The system then derives the value for the **Trigger Termination Process** flag and the termination wait days from the business rule.
		The system checks whether the **Trigger Termination Process** flag is set to **Immediate** for all the active individual memberships which are billed to the delinquent account. If so, the system sets the termination request date of the delinquency event to the termination event trigger date and the **Trigger Termination Process** flag of the delinquency event to **Immediate**. However, if the **Trigger Termination Process** flag is set to **Month End** or **After X Days** for any or all the active memberships, the system calculates the termination request date for each active membership and then considers the latest termination request date of the individual membership. The system sets the termination request date of the delinquency event to the latest termination request date of an active membership and sets the **Trigger Termination Process** flag of the delinquency event to the value derived for the membership with the latest termination request date.

System Event	Algorithm	Algorithm Description
		Note: The system derives the individual memberships using an algorithm which is attached to the **Memberships Derivation** system event of the delinquency process type.
Trigger Date Recalculation Option Derivation	C1-DERRTDBR	This algorithm is required when the **Trigger Date Recalculation on Hold Release Source** flag or **Trigger Date Recalculation on Resume Source** flag in the delinquency process type is set to **Algorithm**. It retrieves the business rules of the **Delinquency Miscellaneous Options Business Rule** category which are effective on the batch business date while executing the **C1-DPMON** batch. Note that the system considers only those effective business rules which are in the **Active** status.
		Once the effective business rules are identified, the system executes these business rules in the order of their priority. Once an individual membership meets the criteria defined in the business rule, the business rule is stamped against the individual membership. The system then derives the value for the **Trigger Date Recalculation on Hold Release** or **Trigger Date Recalculation on Resume** flag, respectively, from the business rule.
		For each active individual membership which is billed to the delinquent account, the system checks the value of the **Trigger Date Recalculation on Hold Release** flag while releasing the hold on the delinquency process. If the **Trigger Date Recalculation on Hold Release** flag is set to **Yes** for at least one active membership, the system sets the **Trigger Date Recalculation on Hold Release** flag of the delinquency event to **Yes**. However, if the **Trigger Date Recalculation on Hold Release** flag is set to **No** for all active memberships, the system sets the **Trigger Date Recalculation on Hold Release** flag of the delinquency event to **No**.
		Similarly, for each active individual membership which is billed to the delinquent account, the system checks the value of the **Trigger Date Recalculation on Resume** flag while resuming the delinquency process. If the **Trigger Date Recalculation on Resume** flag is set to **Yes** for at least one active membership, the system sets the **Trigger Date Recalculation on Resume** flag of the delinquency event to **Yes**. However, if the **Trigger Date Recalculation on Resume** flag is set to **No** for all active memberships, the system sets the **Trigger Date Recalculation on Resume** flag of the delinquency event to **No**.

System Event	Algorithm	Algorithm Description
		Note: The system derives the individual memberships using an algorithm which is attached to the **Memberships Derivation** system event of the delinquency process type.
Wait Days Derivation	C1-DERWDBR	This algorithm is required when the **Wait Period Source** flag in the delinquency process type is set to **Algorithm**. It retrieves the business rules of the **Delinquency Event Attributes Business Rule** category which are effective on the batch business date while executing the **C1-DPMON** batch.
		Note that the system considers only those effective business rules which are in the **Active** status.
		Once the effective business rules are identified, the system executes these business rules in the order of their priority. Once an individual membership meets the criteria defined in the business rule, the business rule is stamped against the individual membership. The system then derives the wait days or wait month and day of month from the business rule depending on the wait period which is configured in the delinquency process type.
		The system then checks the wait days or wait months of each active individual membership and sets the wait days or wait months of the event to the value derived for the membership with the maximum wait days or wait months.
		Note: The system derives the individual memberships using an algorithm which is attached to the **Memberships Derivation** system event of the delinquency process type.

Note: The **Member Notification Days Derivation**, **Policies Derivation**, and **Send Member Level Notification Option Derivation** system events are not applicable for the fully insured individual business.

Grace Period Configuration

While creating a delinquency process type, you need to specify the following grace details using which the system calculates the trigger date for a delinquency event in the delinquency process.

- Whether you want the system to derive the grace period (in days) to calculate grace end date from the delinquency process type or through an algorithm
- The number of days of grace you want to offer from the grace start date

- Whether you want the system to set either of the following as the grace start date when **the Trigger Date Calculation Mode** flag is set to **Usage Basis** and the **Usage Basis** flag is set to **Grace Start Date**:
    - **Delinquency Process Creation Date** – Used when you want to set the grace start date to the date when the delinquency process is created in the system.
    - **Latest Bill Due Date** – Used when you want to set the grace start date to the latest due date of a bill when there are one or more overdue bills for which the delinquency process is initiated.
    - **X Days after Latest Bill Due Date** – Used when you want to set the grace start date after X number of days from the latest bill due date. You can specify the X number of days in the **Add Days To Due Date** field.

Termination and Reinstatement Configuration

While creating a delinquency process type, you need to specify the following details using which you want to calculate the termination request date:

- Whether you want the system to derive the value for the **Trigger Termination Process** flag and the termination wait days from the delinquency process type or through an algorithm
- Whether you want to send an outbound message to the enrollment system for terminating the individual memberships billed to the delinquent account immediately, at the end of the month when the termination event is triggered, or after X number of days (i.e. termination wait days)

Points to Note:

The **Allow Automatic Reinstatement**, **Maximum Automatic Reinstatement Count**, **Trigger Reinstatement Process**, **Reinstatement Wait Days**, **Reinstatement Threshold**, **Days Since Termination**, **Membership Notification Days Source**, and **Member Notification Days** fields are not applicable for the fully insured individual business.

Also, the **Termination Reasons For Automatic Reinstatement** section is not applicable for the fully insured individual business.

Termination Date Rule Priority

While creating a delinquency process type, you need to define the priority for each termination date rule based on the requirements. The system will then consider the termination date rule in the order of its priority. For example, if you have set the following priority in the DT1 delinquency process type:

Termination Date Rule	Priority
Latest Billed Coverage End Date	10
Latest Due Coverage End Date	20
X days After Paid Through Date	30
X Months After Paid Through Date	40
Month End of Grace Period Start	50

Termination Date Rule	Priority
Month End of Termination Letter Creation	60
Month End of Termination Request Event	70
X days After Grace End Date	80
X days After Termination Request Event	90

Let us assume that the A1 account is used to bill the M1 and M2 individual memberships. Now, when the A1 account is delinquent, the system derives the **X days After Paid Through Date** terminate date rule for the M1 membership and the **X days After Grace End Date** termination date rule for the M2 membership using the business rules. The system then considers the termination date rule with the highest priority defined in the respective delinquency process type. In this case, the **X days After Paid Through Date** termination date rule has higher priority than the **X days After Grace End Date** termination date rule, and therefore the system sets the termination date rule of the delinquency event to **X days After Paid Through Date**.

Trigger Date Recalculation Configuration

The system enables you to recalculate trigger dates of the events when a delinquency process which is on hold is released. To enable this feature, you need to specify the following details while creating the respective delinquency process type:

- Whether you want the system to derive the value for the **Trigger Date Recalculation on Hold Release** flag from the delinquency process type or through an algorithm
- Whether you want to recalculate trigger dates of the events when a delinquency process which is on hold is released

Similarly, the system enables you to recalculate trigger dates of the events when a delinquency process which is canceled is resumed. To enable this feature, you need to specify the following details while creating the respective delinquency process type:

- Whether you want the system to derive the value for the **Trigger Date Recalculation on Resume** flag from the delinquency process type or through an algorithm
- Whether you want to recalculate trigger dates of the events when a delinquency process which is canceled is resumed

Approval Configuration

The system enables you to configure the approval process for various actions that you can perform for a delinquency process. For example, the system can initiate the approval process when a user manually holds a delinquency process. Note that you can configure the approval process only for any manual intervention in the delinquency process. To configure the approval process, you need to specify the following details while creating a delinquency process type:

- Whether the approval process must be initiated or not when a user performs certain actions
- The To Do type using which you want to create a To Do for the approver
- The To Do role which indicates the users for whom the To Do should be created

---

- The user actions for which you want to configure the approval process

You can configure the approval process for the following actions which can be manually performed by a user:

- Cancel
- Edit Hold
- Edit Trigger Date
- Hold
- Release Hold
- Request Reinstatement
- Skip Event
- Trigger Event

Delinquency Process Creation

Whenever a bill is completed, the system calculates the next credit review date for the account using the bill due date and credit review grace days (which is defined on the customer class of the account). This next credit review date is stored in a new table named **CI_ADM_RVW_SCH**. The system calculates the next credit review date for the account on the following system events:

- Bill Completion
- Payment Cancellation (due to Non-Sufficient Funds)
- Active Overdue Process Review
- Match Event is Added, Changed, or Deleted
- Payment Freeze
- Payment Agreement Request or Promise to Pay is Broken or Canceled
- Written Off Bill is Reversed

On executing the **C1-ACDLQ** batch, the system considers the accounts which meet the following conditions:

- Collection method of the collection class to which the account belongs is set to **Self-Control Delinquency**
- Next credit review date is either not defined for the account or is earlier than or equal to the batch business date
- Postpone credit review until date is either not defined for the account or is earlier than or equal to the batch business date
- The date derived using the last credit review date + minimum credit review frequency (defined in the delinquency control) is either blank or is earlier than or equal to the batch business date

Once the list of accounts are derived, the system executes the delinquency monitor rule algorithm which is attached to the delinquency control of the respective account's collection class. This algorithm determines whether the account's debt is overdue based on the credit rating, invoice age, and unpaid threshold amount or percentage. If so, it creates a new delinquency process using the delinquency process type which is defined in the specified delinquency process type preference. For more information about the delinquency process type preference, refer to the [Field Mapping](#) section. Note that if the delinquency process already exists for the account in the given status, then the system will add new overdue bills in the existing delinquency process and will not create a new delinquency process for the account. While adding a bill to an existing delinquency process, the system enables you to create a To Do for the delinquency process. For more information on how the **C1-DLNQMNRL** algorithm derives the overdue bills, refer to the [Delinquency Control](#) section.

If the system is not able to derive the delinquency monitor rule algorithm for an account, an appropriate error message is shown in the batch run tree. If a new delinquency process is successfully created for an account, the status of the delinquency process is set to **Initiated**. A delinquency event is created for each delinquency event type which is added in the delinquency process type. If the delinquency event trigger mode is set to **Manual** in the delinquency process type, the status of the delinquency event is set to **Pending**. However, if the delinquency event trigger mode is set to **Automatic** or **Manual or Automatic** in the delinquency process type, the status of the delinquency event is set to **Pending Evaluation**. Once the delinquency events are created for the delinquency process, the system does the following:

- Derives the grace period from the delinquency process type or using an algorithm attached to the **Grace Period Derivation** system event depending on the grace period source which is specified in the delinquency process type
- Derives the grace start date based on the trigger date calculation mode, usage basis, and grace start date basis defined in the delinquency process type
- Calculates the grace end date using the grace start date and the grace period
- Sets the delinquency due date of the delinquency process to the latest bill due date
- Sets the last credit review date of the account to the batch business date

This batch is a multi-threaded batch. The multi-threading is based on account ID and chunks for multi-threading are created based on numerical distribution of account ID. For more information about the batch, refer to *Oracle Revenue Management and Billing Batch Guide*.

Delinquency Event Eligibility Evaluation and Trigger Date Calculation

After executing the **C1-ACDLQ** batch, you need to execute the **C1-DPEVL** batch. The **C1-DPEVL** batch is used to monitor or check whether there is any delinquency process in the **Initiated** status. If there is a delinquency process in the **Initiated** status, the system checks whether its delinquency event trigger mode is set to **Automatic** or **Manual or Automatic** in the respective delinquency process type. If its delinquency event trigger mode is set to **Manual** in the respective delinquency process type, the batch does not consider the delinquency process for the delinquency event eligibility evaluation and trigger date calculation. However, if its delinquency event trigger mode is set to **Automatic** or **Manual or Automatic** in the respective delinquency process type, the system checks whether the trigger dates exist for the delinquency events. If the trigger dates exist for the delinquency events of a delinquency process, the batch does not consider the delinquency process for the delinquency event eligibility evaluation and trigger date calculation. However, if the trigger dates does not exist for the delinquency events of a delinquency process, the batch continues with the delinquency event eligibility evaluation for each delinquency event in the delinquency process.

While evaluating whether the delinquency event is eligible to be included in the delinquency process, the system checks whether the rule type is specified corresponding to the delinquency event type in the respective delinquency process type. If the rule type is specified corresponding to the delinquency event type, the system checks whether the delinquency entity type of the delinquency process is set to **Account** and the delinquency category is set to **Individual**. If so, the system derives the individual memberships which are billed to the delinquent account using an algorithm which is attached to the **Memberships Derivation** system event of the respective delinquency process type.

Once the individual memberships are derived for the account, the system derives the health plan for which the individual membership is availed. Once the health plan is derived for each individual membership, the system derives the rules created using the rule type (which is specified to corresponding to the delinquency event type) from the rules engine which are effective on the batch business date. The effective rules are executed in the order of their priority. If at least one individual membership satisfies the rule, the delinquency event is retained in the delinquency process. However, if none of the individual memberships satisfy the rule, the delinquency event is removed from the delinquency process.

Once the delinquency event eligibility evaluation is done for each delinquency event, the system does the following:

- Derives the wait days or wait month and the day of the month from the delinquency process type or using an algorithm attached to the **Wait Days Derivation** system event depending on the wait period source which is specified in the delinquency process type
- Calculates the trigger date for the delinquency event based on the trigger date calculation mode, usage basis, and wait period defined in the delinquency process type
- Sets the delinquency due date of the delinquency process to the latest bill due date

If the **Consider Work Days** option is selected in the respective delinquency process type, the system derives the work calendar of the division to which the account belongs. The system then checks whether the calculated trigger date falls on the working (i.e. business) day. If the calculated trigger date does not fall on the working (i.e. business) day, the system will accordingly add days to the calculated trigger date to ensure that it falls on the next working day.

If the calculated trigger date of a delinquency event is later than or equal to the batch business date, the status of the delinquency event is changed from **Pending Evaluation** to **Pending**. But, if the calculated trigger date of a delinquency event is earlier than the batch business date and if the **Skip Past Event** option is selected corresponding to the respective delinquency event type, the system sets the status of the delinquency event to **Skipped**. However, if the calculated trigger date of a delinquency event is earlier than the batch business date and if the **Skip Past Event** option is not selected corresponding to the respective delinquency event type, the system checks whether the **Move Past Dated Event to Future** option is selected corresponding to the respective delinquency event type.

> **Note:** You can select the **Move Past Dated Event to Future** option only when the wait period is set to **Monthly Cyclic** in the respective delinquency process type.

If the **Move Past Dated Event to Future** option is selected corresponding to the respective delinquency event type, the system moves the trigger date by adding one month until the trigger date is later than or equal to the batch business date. But, if the **Move Past Dated Event to Future** option is not selected corresponding to the respective delinquency event type, the system makes no changes to the trigger date even if it is past dated. Finally, the system sets the status of the delinquency event to **Pending**.

> **Points to Note:**
>
> If the trigger date of an event is moved to a future date, the trigger date of the subsequent events in the delinquency process is also moved accordingly. This is because the system is designed to trigger only one event at a time in a month.
>
> Usually, this batch changes the status of the delinquency event from **Pending Evaluation** to **Pending**. But, if the wait period in the respective delinquency process type is set to **Monthly Cyclic** and if the **Move Past Dated Event to Future** option is selected corresponding to the respective delinquency event type, the **C1-ADJTGDT** algorithm attached to the **Post-Processing** system event of the **C1-DPEVL** batch moves the past dated trigger date to a future date and changes the status of the delinquency event to **Pending**.

If the **Preview Required** option is not selected corresponding to the delinquency event type in the respective delinquency process type, the system sets the preview status of the delinquency event to **Not Required**. However, if the **Preview Required** option is selected corresponding to the delinquency event type in the respective delinquency process type, the system does the following:

- Derives the preview days specified corresponding to the delinquency event type in the delinquency process type
- Calculates the preview date (i.e. Trigger Date – Preview Days) for the delinquency event
- Sets the preview status of the delinquency event to **Pending**

The **C1-DPEVL** batch is a multi-threaded batch. The multi-threading is based on delinquency event ID and chunks for multi-threading are created based on numerical distribution of delinquency event ID. For more information about the batch, refer to *Oracle Revenue Management and Billing Batch Guide*.

Delinquency Process Maintenance

Once the delinquency process is created in the **Initiated** status, the system enables you to view the details of the delinquency process. Two new screens named **Delinquency Process Search** and **Delinquency Process** are introduced in this release. The **Delinquency Process Search** screen enables you to search for a delinquency process either using the details of the delinquency process or using the details of the account, person, or membership for which the delinquency process is created.

Note: The Person ID field while searching for a delinquency process is not applicable for the fully insured individual business.

The **Delinquency Process** screen enables you to view the details of a delinquency process. It contains the following zones:

- **Delinquency Process Details** – Display the basic details of the delinquency process.
- **Additional Attributes** – Display the characteristics of the delinquency process.
- **Bills** – Lists the overdue bills of the account for which the delinquency process is created.
- **Events** - List all events that are included in the delinquency process.
- **Event Notifications** – Lists the notifications (i.e. To Do or customer contact), if any, created for the delinquency event.
- **Approval Details** – Lists the approval records that are created while performing manual actions at the delinquency process or delinquency event level.
- **Hold Details** – Lists the hold records that are created whenever you hold the delinquency process.
- **Log** – Lists the complete trail of actions performed on the delinquency process.

Manual Actions at the Delinquency Process Level

The **Delinquency Process** screen enables you to perform the following actions at the delinquency process level:

- **Edit** – Used when you want to add additional information for the delinquency process in the form of comments or characteristics.
- **Hold** – Used when you want to manually hold the delinquency process. The **Hold** button appears only when the delinquency process is in the **Initiated**, **Delinquency In Progress**, or **Pending Termination** status. While manually holding a delinquency process, you need to specify the following:
  - A hold end date which is later than the system date
  - The reason why you want to hold the delinquency process. Note that the list includes only those reasons which are defined for the **Initiate Hold** status of the **C1-DelinquencyProcess** business object.

The hold start date is set to the system date when you manually hold the delinquency process. On saving the changes, the status of the delinquency process is changed to **Initiate Hold**. The system then checks whether the approval is required for the **Hold** action in the respective delinquency process type.

If the approval is not required while manually holding the delinquency process, the status of the delinquency process is changed to **On Hold** immediately. However, if the approval is required while manually holding the delinquency process, an approval record is created in the **Approval In Progress** status. The status of the delinquency process is changed to its previous status (i.e. **Initiated**, **Delinquency In Progress**, or **Pending Termination**). Note that, if the delinquency process is in the **Initiated** or **Delinquency In Progress** status, the system continues to trigger the delinquency events on the trigger date even when the approval is in progress until the approver approves the manual hold.

If the approver approves the manual hold, the status of the delinquency process is changed to **On Hold** and the hold start date is changed to the date when the approver approves the manual hold. In addition, the status of the approval record is changed to **Approved**. However, if the approver rejects the manual hold, the status of the approval record is changed to **Rejected** and the status of the delinquency process remains as **Initiated**, **Delinquency In Progress**, or **Pending Termination**.

- **Cancel** – Used when you want to manually cancel the delinquency process. The **Cancel** button appears only when the delinquency process is in the **Initiated**, **Delinquency In Progress**, or **On Hold** status. While manually canceling a delinquency process, you need to specify the cancel reason and comments, if any. You must specify a reason which is defined for the **Initiate Cancel** status of the **C1-DelinquencyProcess** business object in the **Status Reason** screen.

  On manually canceling a delinquency process, the status of the delinquency process is set to **Initiate Cancel**. The system then checks whether the approval is required for the **Cancel** action in the respective delinquency process type. If the approval is not required while manually canceling the delinquency process, the status of the delinquency process is changed to **Canceled**.

  However, if the approval is required while manually canceling the delinquency process, an approval record is created in the **Approval In Progress** status. The status of the delinquency process is changed to the previous status (i.e. **Initiated**, **Delinquency In Progress**, or **On Hold)**. Note that, if the delinquency process is in the **Initiated** or **Delinquency In Progress** status, the system continues to trigger the delinquency events on the trigger date even when the approval is in progress until the approver approves the manual cancellation of the delinquency process.

  If the approver approves the manual cancellation of the delinquency process, the status of the delinquency process is set to **Canceled**. In addition, the status of the approval record is changed to **Approved**. However, if the approver rejects the manual cancellation of the delinquency process, the status of the approval record is changed to **Rejected** and the status of the delinquency process remains as **Initiated**, **Delinquency In Progress**, or **On Hold**.

- **Release** – Used when you want to manually release the hold on the delinquency process. The **Release** button appears only when the delinquency process is in the **On Hold** status. While manually releasing the hold on the delinquency process, you are prompted to confirm the hold release. On confirming the hold release, the system checks whether the approval is required for the **Release Hold** action in the respective delinquency process type. If the approval is not required while manually releasing the hold on the delinquency process, the status of the delinquency process is changed to the previous status (i.e. **Initiated**, **Delinquency In Progress**, or **Pending Termination**) immediately. In addition, the hold end date is set to the date when you manually release the hold on the delinquency process.

  However, if the approval is required while manually releasing the hold on the delinquency process, an approval record is created in the **Approval In Progress** status. The status of the delinquency process remains as **On Hold**. If the approver approves the hold release, the status of the delinquency process is changed to the previous status (i.e. **Initiated**, **Delinquency In Progress**, or **Pending Termination**). The hold end date is set to the date when the approver approves the hold release. In addition, the status of the approval record is changed to **Approved**. However, if the approver rejects the hold release, the status of the approval record is changed to **Rejected** and the status of the delinquency process remains as **On Hold**.

- **Request Reinstatement** – This button is not applicable for the delinquency processes which are created for the fully insured individual business.

- **Edit Hold** – On clicking the **Edit** icon corresponding to the hold record of the delinquency process, you can edit the hold end date and hold reason. Note that the **Edit** icon appears only when the delinquency process is in the **On Hold** status.

  While editing the hold details, you can specify the following:

  - A hold end date which is later than the system date and which is later than the date which is the latest of the following:

    o   Postpone credit review date

    o   Latest schedule date specified in the payment agreement request

    o   Latest schedule date specified among all the active promise to pay

  - The reason why you want to hold the delinquency process. Note that the list includes only those reasons which are defined for the **Initiate Hold** status of the **C1-DelinquencyProcess** business object.

  On saving the changes, the system checks whether the approval is required for the **Edit Hold** action in the respective delinquency process type. If the approval is not required while manually editing the hold details, the hold end date and hold reason of the delinquency process are updated immediately.

  However, if the approval is required while manually editing the hold details, an approval record is created in the **Approval In Progress** status. The status of the delinquency process remains as **On Hold**. If the approver approves the changes, the hold end date and hold reason of the delinquency process are updated. In addition, the status of the approval record is changed to **Approved**. However, if the approver rejects the hold changes, the status of the approval record is changed to **Rejected** and the hold changes are not reflected in the delinquency process.

Manual Actions at the Delinquency Event Level

You can also perform the following actions for each delinquency event of the delinquency process from the **Delinquency Process** screen:

- **Edit Trigger Date** – On clicking the **Edit Trigger Date** icon corresponding to a delinquency event, you can specify a new trigger date for the delinquency event. Note that the **Edit Trigger Date** icon appears only when the following conditions are met:
  - The delinquency process is in the **Initiated** or **Delinquency In Progress** status
  - The delinquency event trigger mode is set to **Manual or Automatic** in the respective delinquency process type
  - The delinquency event is in the **Pending** status

  Once you specify the new trigger date and save the changes, the system checks whether the approval is required for the **Edit Trigger Date** action in the respective delinquency process type. If the approval is not required while manually editing the trigger date, the trigger date of the delinquency event is updated immediately.

  However, if the approval is required while manually editing the trigger date, an approval record is created for the delinquency event in the **Approval In Progress** status. Note that the icons in the **Trigger Events** and **Skip Event** columns (if any) corresponding to the delinquency event are removed immediately, so that users should not be able to perform any other actions on the delinquency event. The status of the delinquency process remains as **Initiated** or **Delinquency In Progress**. Note that the system continues to trigger the delinquency events on the trigger date even when the approval is in progress. If a delinquency event is triggered before the approval is received for the delinquency event, the status of the approval record is changed to **Expired**.

  If the approver approves the changes, the trigger date of the delinquency event is updated. The icons in the **Trigger Events** and **Edit Trigger Date** columns (if any) corresponding to the delinquency event are displayed again. In addition, the status of the approval record is changed to **Approved**. However, if the approver rejects the changes, the status of the approval record is changed to **Rejected** and the trigger date of the delinquency event is not updated. However, the icons in the **Trigger Events** and **Edit Trigger Date** columns (if any) corresponding to the delinquency event are displayed again.

- **Skip Event** – On clicking the **Skip Event** icon corresponding to a delinquency event, the system checks whether the approval is required for the **Skip Event** action in the respective delinquency process type. Note that the **Skip Event** icon appears only when the following conditions are met:
  - The delinquency process is in the **Initiated** or **Delinquency In Progress** status
  - The delinquency event trigger mode is set to **Manual** or **Manual or Automatic** in the respective delinquency process type
  - The delinquency event is in the **Pending** status

If the approval is not required while manually skipping the delinquency events, the status of the delinquency event is changed to **Skipped** immediately. In addition, the icons in the **Trigger Events** and **Edit Trigger Date** columns (if any) corresponding to the delinquency event are also removed.

However, if the approval is required while manually skipping the delinquency events, an approval record is created for the delinquency event in the **Approval In Progress** status. Note that the icons in the **Trigger Events** and **Edit Trigger Date** columns (if any) corresponding to the delinquency event are removed immediately, so that users should not be able to perform any other actions on the delinquency event. The status of the delinquency process remains as **Initiated** or **Delinquency In Progress**. Note that the system continues to trigger the delinquency events on the trigger date even when the approval is in progress. If a delinquency event is triggered before the approval is received for the delinquency event, the status of the approval record is changed to **Expired**.

If the approver approves the skip action, the status of the delinquency event is changed to **Skipped** immediately. In addition, the status of the approval record is changed to **Approved**. Note that the icons in the **Trigger Events** and **Edit Trigger Date** columns corresponding to the delinquency event are not displayed again. However, if the approver rejects the skip action, the status of the approval record is changed to **Rejected**. The status of the delinquency event remains as **Pending**. In addition, the icons in the **Trigger Events** and **Edit Trigger Date** columns (if any) corresponding to the delinquency event are displayed again.

- **Trigger Event** - On clicking the **Trigger Event** icon corresponding to a delinquency event, the system checks whether the approval is required for the **Trigger Event** action in the respective delinquency process type. Note that the **Trigger Event** icon appears only when the following conditions are met:

    o The delinquency process is in the **Initiated** or **Delinquency In Progress** status

    o The delinquency event trigger mode is set to **Manual** or **Manual or Automatic** in the respective delinquency process type

    o The delinquency event is in the **Pending** status

If the approval is not required while manually triggering the delinquency events, the algorithms attached to the **Event Activation** system event of the respective delinquency event type are executed immediately in the specified order. The status of the delinquency event is changed to **Completed** when all algorithms attached to the **Event Activation** system event are successfully executed. In addition, the icons in the **Skip Event** and **Edit Trigger Date** columns (if any) corresponding to the delinquency event are also removed.

However, if the approval is required while manually triggering the delinquency events, an approval record is created for the delinquency event in the **Approval In Progress** status. Note that the icons in the **Skip Event** and **Edit Trigger Date** columns (if any) corresponding to the delinquency event are removed immediately, so that users should not be able to perform any other actions on the delinquency event.

The status of the delinquency process remains as **Initiated** or **Delinquency In Progress**. Note that the system continues to trigger the delinquency events on the trigger date even when the approval is in progress. If a delinquency event is triggered before the approval is received for the delinquency event, the status of the approval record is changed to **Expired**.

If the approver approves the manual trigger action, the algorithms attached to the **Event Activation** system event of the respective delinquency event type are executed immediately in the specified order. The status of the delinquency event is changed to **Completed** when all algorithms attached to the **Event Activation** system event are successfully executed. In addition, the status of the approval record is changed to **Approved**. Note that the icons in the **Skip Event** and **Edit Trigger Date** columns corresponding to the delinquency event are not displayed again. However, if the approver rejects the manual trigger action, the status of the approval record is changed to **Rejected**. The status of the delinquency event remains as **Pending**. In addition, the icons in the **Skip Event** and **Edit Trigger Date** columns (if any) corresponding to the delinquency event are displayed again.

Note that the algorithms attached to the **Monitor Completed Event** system event of the respective delinquency event type are not executed in the online mode when the status of the delinquency event is changed to **Completed**. These algorithms are executed when you invoke the **C1-DPMON** batch.

Delinquency Process Approval

At a time, the system creates an approval record for only one manual action at the delinquency process or delinquency event level. Therefore, if an approval record exists for a delinquency process, the system does not allow you to perform other manual action on the delinquency process until the respective approval record's status changes to **Approved** or **Rejected**. Similarly, if an approval record exists for a delinquency event, the system does not allow you to perform other manual action on the delinquency event until the respective approval record's status changes to **Approved** or **Rejected**. However, you can still perform a manual action for other delinquency events of the delinquency process.

The system enables you to withdraw an approval request for any manual action at the delinquency process or delinquency event level. Only the users of the user group to which the submitter belongs can withdraw the approval request which is in the **Approval In Progress** status. On confirming the withdrawal of an approval request, the status of the approval record is changed to **Withdrawn**.

The system enables you to perform the following actions for an approval record from the **Delinquency Process** screen:

- **Add Comments** – Used when you want to add additional information about the approval record. Both the submitter and approver can add the comments.

- **View Details** – Used to view the details of the manual action which is performed by the submitter. If the approval record is created while manually editing the trigger date of a delinquency event, the system enables you to view the old and new trigger date of the delinquency event. However, if the approval record is created while manually editing the hold details of the delinquency process, the system enables you to view the old and new hold end dates and hold reasons. Both the submitter and approver can view the details of an approval record.

- **Approve** – Used when you want to approve the manual action performed at the delinquency process or delinquency event level. Only users with the approver To Do role can approve the approval request.

- **Reject** - Used when you want to reject the manual action performed at the delinquency process or delinquency event level. Only users with the approver To Do role can reject the approval request.

- **Withdraw** – Used when you want to withdraw the approval request. Only the users of the user group to which the submitter belongs can withdraw the approval request which is in the **Approval In Progress** status.

Points to Note:

If the delinquency process is in the **On Hold** status, you cannot approve or reject any other manual actions except the **Release Hold** and **Edit Hold** actions. All other manual actions can be reviewed and accordingly approved or rejected by the approver only after the hold on the delinquency process is released.

When a delinquency process is canceled, the status of the pending approval records is changed to **Expired**.

Canceling a Delinquency Process

Oracle Revenue Management and Billing enables you to cancel a delinquency process of an account manually or automatically. You can manually cancel a delinquency process whenever required. In addition, the system automatically cancels a delinquency process when either of the following condition is met:

- Payments or adjustments are made against the overdue bills, for which the delinquency process is created, and as a result the unpaid balance of the overdue bills is within the tolerance limit

- Collection class of the account, for which the delinquency process is created, is changed

- Voluntary termination is received for an individual membership (which is billed to the delinquent account) with a termination date that is earlier than or equal to the calculated termination date

- Voluntary cancellation is received for an individual membership (which is billed to the delinquent account)

To enable the automatic cancellation process for the delinquency processes which are created using a particular delinquency process type, you need to attach the **C1-CANDELPRC** algorithm to the **Cancel Criteria** system event of the respective delinquency process type. This algorithm is invoked when you execute the **C1-DPMON** batch. For more information about the automatic cancellation process, refer to the respective algorithm description in the [Delinquency Process Type Algorithms](#) section.

The **Cancel** button in the **Delinquency Process** screen enables you to manually cancel a delinquency process. It appears only when the delinquency process is in the **Initiated**, **Delinquency In Progress**, or **On Hold** status. On manually canceling a delinquency process, the status of the delinquency process is changed to **Initial Cancel**. For more information about the manual cancellation process, refer to the [Manual Actions at the Delinquency Process Level](#) section.

Once a delinquency process is manually or automatically canceled, the status of the delinquency process is changed to **Canceled**. The system then executes the **C1-DPCAN** algorithm attached to the **Canceled** status in the lifecycle of the **C1-DelinquencyProcess** business object. The **C1-DPCAN** algorithm does the following:

- Creates a To Do notification with the cancel reason
- If there is any hold record for the delinquency process in the **Active** status, the system changes the status of the hold record to **Complete**.
- If there is any pending record for approval at the delinquency event or process level, the system changes the status of pending approval records to **Expired**.

This algorithm contains the following parameters:

- **To Do Type** – Used to indicate the To Do type using which you want to create a To Do when a delinquency process is canceled. This parameter is required.
- **To Do Role** – Used to indicate that users with the specified To Do role must receive the To Do notification. This parameter is optional. If you do not specify this parameter, the system considers the default To Do role associated with the To Do type.
- **To Do Entry required for Individual Delinquency (Valid Values - Y/N)** – Used to indicate whether the To Do notification should be created on cancellation for the fully insured individual business. The valid values are **Y** and **N**.

Resuming a Delinquency Process

Oracle Revenue Management and Billing enables you to resume a delinquency process which was canceled due to a payment or adjustment made against the overdue bill for which the delinquency process was created. The system enables you to resume a delinquency process only when the respective payment or adjustment is canceled due to non-sufficient funds. The insurance company may come across the following scenarios:

- Scenarios wherein the delinquency process is canceled in between events when the status of the delinquency process is **Initiated**, **Delinquency In Progress**, or **On Hold**
- Scenarios wherein the delinquency process is canceled after the delinquency process moves to the **Customer Reinstated** status due the individual membership reinstatement

The system enables you to handle both these scenarios when a payment or adjustment made against a bill is canceled.

Resuming a Delinquency Process on Payment Cancellation

To automatically resume a delinquency process when a payment made against the respective overdue bill is canceled, you need to attach the following algorithms to the **Payment Cancellation** system event of the required customer classes:

- **C1-CDPPAYCAN** - This algorithm is invoked when a payment is canceled. It checks whether the **Delinquency Process** (**C1-DELPO)** characteristic is defined on the payment. If the **C1-DELPO** characteristic is not defined on the payment, the system creates a customer contact for the main customer of the account for which the payment was made. In addition, the system stamps the payment ID as a characteristic using the given characteristic type on the customer contact.

  However, if the **C1-DELPO** characteristic is defined on the payment, the system derives the delinquency process ID from characteristic value. It then validates whether the delinquency process exist with the given ID in the system. Note that if the delinquency process does not exist, no further action takes place on the payment cancellation. However, if the delinquency process ID is successfully validated, the system checks the following:

  - The status of the delinquency process is set to a status which is specified in the **Delinquency Process Canceled Status** attribute of the delinquency process preference.
  - The related delinquency process does not exist for the delinquency process.

  If either or both the conditions are not satisfied, no further action takes place on the payment cancellation. However, if both the conditions are satisfied, the system invokes the **C1-CANDELPRC** algorithm attached to the **Cancel Criteria** system event of the respective delinquency process type. If the cancel criteria is met, the delinquency process remains in the existing status and no further action takes place on the payment cancellation. However, if the cancel criteria is not met, the status of the delinquency process is set to the status prior to cancellation (i.e. **Initiated**, **Delinquency In Progress**, or **On Hold**). If the status of the delinquency process before cancellation was **On Hold**, then the status of the hold record is changed to **Active**. The system also creates a customer contact for the main customer of the account for which the payment was made. In addition, the system does the following:

  - Adds a log entry for the customer contact in the delinquency process
  - Stamps the customer contact ID as a characteristic using the given characteristic type on the payment
  - Stamps the payment ID as a characteristic using the given characteristic type on the customer contact

  This algorithm derives the preferred contact method for the customer contact. While deriving the preferred contact method, the algorithm first fetches the main customer's bill route type on the account and then fetches the bill routing method of the bill route type. Once the bill routing method is derived, the system considers the preference which is defined using the **Routing Method – Contact Method Mapping** preference category. It then checks whether any preferred contact method is mapped to the bill routing method in the preference.

If the preferred contact method is mapped to the bill routing method in the preference, the system stamps it on the customer contact. However, if the preferred contact method is not mapped to the bill routing method in the preference, the system considers the default contact method as the preferred contact method and stamps it on the customer contact.

This algorithm contains the following parameters:

- **Customer Contact Type** - Used to indicate the customer contact type using which you want to create the customer contact.

- **Customer Contact Class** - Used to indicate the customer contact class to which the customer contact type belongs.

- **Default Contact Method** - Used to indicate the contact method that you want to use when the preferred contact method cannot be derived for the routing method from the preference.

- **Customer Contact Characteristic Type** - Used to indicate the characteristic type using which you want to store the customer contact ID on the payment. Here, you must specify a characteristic type where the characteristic entity is set to **Payment**.

- **Payment Characteristic For Customer Contact** - Used to indicate the characteristic type using which you want to store the payment ID on the customer contact. Here, you must specify a characteristic type where the characteristic entity is set to **Customer Contact**.

All the above parameters are mandatory.

- **C1-RODPDUEPX** - This algorithm is invoked when a payment is canceled. It derives the reason why the payment was canceled and validates whether it is listed as a value in the **Excluded Payment Cancel Reason List** parameter. If the payment cancel reason is listed in the **Excluded Payment Cancel Reason List** parameter, no further action takes place on the payment cancellation. However, if the payment cancel reason is not listed in the **Excluded Payment Cancel Reason List** parameter, the system checks whether the **Delinquency Process** (**C1-DELPO**) characteristic is defined on the payment. If the **C1-DELPO** characteristic is not defined on the payment, no further action takes place on the payment cancellation.

  However, if the **C1-DELPO** characteristic is defined on the payment, the system derives the delinquency process ID from characteristic value. It then validates whether the delinquency process exist with the given ID in the system. If the delinquency process does not exist, no further action takes place on the payment cancellation. However, if the delinquency process ID is successfully validated, the system checks whether the status of the delinquency process is set to a status which is specified in the **Delinquency Process Customer Reinstated Status** attribute of the delinquency process preference. If so, the system checks the value of the **Delinquency Process Resume Switch (Valid Values – Y,N)** parameter. If the **Delinquency Process Resume Switch (Valid Values – Y,N)** parameter is set to **Y**, the system does the following:

  o Sets the status of the delinquency process to a status which is specified in the **Delinquency Reinstatement Payment/Adjustment Canceled Status** attribute of the delinquency process preference

  However, if the **Delinquency Process Resume Switch (Valid Values – Y,N)** parameter is set to **N**, the system does the following:

o   If a delinquency process created using the given delinquency process type already exists for the account, the system adds the overdue bill for which the payment is canceled in the existing delinquency process. Note that the system considers an existing delinquency process which is in the status that is listed in the **Delinquency Process Status For Existing Delinquency Process** parameter of the **C1-DLNQMNRL** algorithm attached to the delinquency control of the account's collection class.

o   If a delinquency process does not exist for the account, the system creates a new delinquency process using the given delinquency process type. It adds the overdue bill for which the payment is canceled in the new delinquency process. Also, if there are any other overdue bills of the account that do not meet the tolerance limit are also added in the delinquency process. The status of the delinquency process is set to a status which is specified in the **Delinquency Process Initiated Status** attribute of the delinquency process preference.

This algorithm contains the following parameters:

- **Excluded Payment Cancel Reason List** – Used to specify a list of comma-separated payment cancel reasons. The system will then skip the payments which are canceled using the given payment cancel reason and will not take any actions on the payment cancellation. You can specify maximum five comma-separated values for this parameter. Here, you must specify a payment cancel reason which is already defined in the system.

- **Delinquency Process Resume Switch (Valid Values – Y,N)** – Used to indicate whether the original delinquency process which was canceled due to the payment should be resumed when the payment is canceled. The valid values are – **Y** and **N**. This parameter is mandatory.

- **Delinquency Process Type** – Used to indicate the delinquency process type using which a new process delinquency process should be created for the account when the payment is canceled. It is also used to check whether any existing delinquency process created using the given delinquency process type already exists in the system. Here, you must specify a delinquency process type which is already defined in the system. This parameter is required only when the **Delinquency Process Resume Switch (Valid Values – Y,N)** parameter is set to **N**.

Resuming a Delinquency Process on Adjustment Cancellation

To automatically resume a delinquency process when an adjustment made against the respective overdue bill is canceled, you need to attach the following algorithms to the **Adjustment Cancellation** system event of the required adjustment types:

- **C1-DPADJCAN -** This algorithm is invoked when an adjustment is canceled. It checks whether the **Delinquency Process** (**C1-DELPO**) characteristic is defined on the adjustment. If the **C1-DELPO** characteristic is not defined on the adjustment, the system creates a customer contact for the main customer of the account for which the adjustment was created. In addition, the system stamps the adjustment ID as a characteristic using the given characteristic type on the customer contact.

However, if the **C1-DELPO** characteristic is defined on the adjustment, the system derives the delinquency process ID from characteristic value. It then validates whether the delinquency process exist with the given ID in the system. Note that if the delinquency process does not exist, no further action takes place on the adjustment cancellation. However, if the delinquency process ID is successfully validated, the system checks the following:

- o The status of the delinquency process is set to a status which is specified in the **Delinquency Process Canceled Status** attribute of the delinquency process preference.
- o The related delinquency process does not exist for the delinquency process.

If either or both the conditions are not satisfied, no further action takes place on the adjustment cancellation. However, if both the conditions are satisfied, the system invokes the **C1-CANDELPRC** algorithm attached to the **Cancel Criteria** system event of the respective delinquency process type. If the cancel criteria is met, the delinquency process remains in the existing status and no further action takes place on the adjustment cancellation. However, if the cancel criteria is not met, the status of the delinquency process is set to the status prior to cancellation (i.e. **Initiated**, **Delinquency In Progress**, or **On Hold**). If the status of the delinquency process before cancellation was **On Hold**, then the status of the hold record is changed to **Active**. The system also creates a customer contact for the main customer of the account for which the adjustment was created. In addition, the system does the following:

- Adds a log entry for the customer contact in the delinquency process
- Stamps the customer contact ID as a characteristic using the given characteristic type on the adjustment
- Stamps the adjustment ID as a characteristic using the given characteristic type on the customer contact

This algorithm derives the preferred contact method for the customer contact. While deriving the preferred contact method, the algorithm first fetches the main customer's bill route type on the account and then fetches the bill routing method of the bill route type. Once the bill routing method is derived, the system considers the preference which is defined using the **Routing Method – Contact Method Mapping** preference category. It then checks whether any preferred contact method is mapped to the bill routing method in the preference. If the preferred contact method is mapped to the bill routing method in the preference, the system stamps it on the customer contact. However, if the preferred contact method is not mapped to the bill routing method in the preference, the system considers the default contact method as the preferred contact method and stamps it on the customer contact.

This algorithm contains the following parameters:

- **Customer Contact Type** - Used to indicate the customer contact type using which you want to create the customer contact.
- **Customer Contact Class** - Used to indicate the customer contact class to which the customer contact type belongs.
- **Default Contact Method** - Used to indicate the contact method that you want to use when the preferred contact method cannot be derived for the routing method from the preference.

- **Customer Contact Characteristic Type** - Used to indicate the characteristic type using which you want to store the customer contact ID on the adjustment. Here, you must specify a characteristic type where the characteristic entity is set to **Adjustment**.

- **Adjustment Characteristic For Customer Contact** - Used to indicate the characteristic type using which you want to store the adjustment ID on the customer contact. Here, you must specify a characteristic type where the characteristic entity is set to **Customer Contact**.

All the above parameters are mandatory.

- **C1-RODPONADX** - This algorithm is invoked when an adjustment is canceled. It derives the reason why the adjustment was canceled and validates whether it is listed as a value in the **Excluded Adjustment Cancel Reason List** parameter. If the adjustment cancel reason is listed in the **Excluded Adjustment Cancel Reason List** parameter, no further action takes place on the adjustment cancellation. However, if the adjustment cancel reason is not listed in the **Excluded Adjustment Cancel Reason List** parameter, the system checks whether the **Delinquency Process (C1-DELPO)** characteristic is defined on the adjustment. If the **C1-DELPO** characteristic is not defined on the adjustment, no further action takes place on the adjustment cancellation.

  However, if the **C1-DELPO** characteristic is defined on the adjustment, the system derives the delinquency process ID from characteristic value. It then validates whether the delinquency process exist with the given ID in the system. If the delinquency process does not exist, no further action takes place on the adjustment cancellation. However, if the delinquency process ID is successfully validated, the system checks whether the status of the delinquency process is set to a status which is specified in the **Delinquency Process Customer Reinstated Status** attribute of the delinquency process preference. If so, the system checks the value of the **Delinquency Process Resume Switch** parameter. If the **Delinquency Process Resume Switch** parameter is set to **Y**, the system does the following:

  - Sets the status of the delinquency process to a status which is specified in the **Delinquency Reinstatement Payment/Adjustment Canceled Status** attribute of the delinquency process preference

  However, if the **Delinquency Process Resume Switch** parameter is set to **N**, the system does the following:

  - If a delinquency process created using the given delinquency process type already exists for the account, the system adds the overdue bill for which the adjustment is canceled in the existing delinquency process. Note that the system considers an existing delinquency process which is in the status that is listed in the **Delinquency Process Status For Existing Delinquency Process** parameter of the **C1-DLNQMNRL** algorithm attached to the delinquency control of the account's collection class.

o   If a delinquency process does not exist for the account, the system creates a new delinquency process using the given delinquency process type. It adds the overdue bill for which the adjustment is canceled in the new delinquency process. Also, if there are any other overdue bills of the account that do not meet the tolerance limit are also added in the delinquency process. The status of the delinquency process is set to a status which is specified in the **Delinquency Process Initiated Status** attribute of the delinquency process preference.

This algorithm contains the following parameters:

- **Excluded Adjustment Cancel Reason List** – Used to specify a list of comma-separated adjustment cancel reasons. The system will then skip the adjustments which are canceled using the given adjustment cancel reason and will not take any actions on the adjustment cancellation. You can specify maximum five comma-separated values for this parameter. Here, you must specify an adjustment cancel reason which is already defined in the system.

- **Delinquency Process Resume Switch** – Used to indicate whether the original delinquency process which was canceled due to the adjustment should be resumed when the adjustment is canceled. The valid values are – **Y** and **N**. This parameter is mandatory.

- **Delinquency Process Type** – Used to indicate the delinquency process type using which a new process delinquency process should be created for the account when the adjustment is canceled. It is also used to check whether any existing delinquency process created using the given delinquency process type already exists in the system. Here, you must specify a delinquency process type which is already defined in the system. This parameter is required only when the **Delinquency Process Resume Switch** parameter is set to **N**.

Recalculating Trigger Dates for Pending Events

Whenever a delinquency process moves from the **Canceled** status to **Initiated**, **Delinquency In Progress**, or **On Hold** on the payment or adjustment cancellation, the system executes the **C1-RECALCTG** algorithm attached to the **Canceled** status of the **C1-DelinquencyProcess** business object. This algorithm recalculates the trigger date for the delinquency events which are in the **Pending** status. It derives the difference between the delinquency process resume date and delinquency process cancellation date. These extra days are then added to the existing trigger date of the pending delinquency events and thereby the new trigger dates are calculated for the remaining events.

If the **Consider Work Days** option is selected in the respective delinquency process type, the system derives the work calendar of the division to which the account belongs. The system then checks whether the new trigger date falls on the working (i.e. business) day. If the new trigger date does not fall on the working (i.e. business) day, the system will accordingly add days to the new trigger date to ensure that it falls on the next working day.

Membership Cancellation Outbound Message

The **C1-MemberCancRequest** business object is newly introduced in this release. It enables you to create a cancellation outbound message for an individual membership. To create the cancellation outbound messages for individual memberships through the **Delinquency Management** feature, you need to do the following:

- Create an external system to which you want to send the outbound messages. Note that the implementation team needs to design how the outbound message should be sent to the external system based on the customer requirements (i.e. through a batch control or a real-time service). Also, any response from the external system on receiving the outbound message should be captured in ORMB.

- Add the external system in the **C1-SourceSystemLookup** extendable lookup

- Create an outbound message type using which you want to create a cancellation outbound message for an individual membership. Here, you must specify the following:

    o The **C1-MemberCancRequest** business object using which you want to create the cancellation outbound messages for the individual memberships.

    o An external system to which you want to send the cancellation outbound messages that are created using the outbound message type.

- Create a delinquency process type using which you want to create a delinquency process to initiate cancellation request through an outbound message. For more information on how to design a delinquency process type to initiate cancellation request for an individual membership, refer to the Membership Cancellation Delinquency Process section.

Once the status reason of the individual memberships is set to the status reason which is specified in the **Awaiting Membership Cancellation Reason** attribute of the delinquency process preference, you need to execute the **C1-MTOMT** batch to generate the cancellation outbound messages for the individual memberships. For more information about the batch, refer to the Outbound Message Generation for Delinquency Management section.

Membership Termination Outbound Message

The **C1-MemberTermRequest** business object is newly introduced in this release. It enables you to create a termination outbound message for an individual membership. To create the termination outbound messages for individual memberships through the **Delinquency Management** feature, you need to do the following:

- Create an external system to which you want to send the outbound messages. Note that the implementation team needs to design how the outbound message should be sent to the external system based on the customer requirements (i.e. through a batch control or a real-time service). Also, any response from the external system on receiving the outbound message should be captured in ORMB.

- Add the external system in the **C1-SourceSystemLookup** extendable lookup

- Create an outbound message type using which you want to create a termination outbound message for an individual membership. Here, you must specify the following:

  o The **C1-MemberTermRequest** business object using which you want to create the termination outbound messages for the individual memberships.

  o An external system to which you want to send the termination outbound messages that are created using the outbound message type.

- Create a delinquency process type using which you want to create a delinquency process to initiate termination request through an outbound message. For more information on how to design a delinquency process type to initiate termination request for an individual membership, refer to the Membership Termination Delinquency Process section.

Once the status reason of the individual memberships is set to the status reason which is specified in the **Membership Termination Reason for Delinquency** attribute of the delinquency process preference, you need to execute the **C1-MTOMT** batch to generate the termination outbound messages for the individual memberships. For more information about the batch, refer to the Outbound Message Generation for Delinquency Management section.

Outbound Message Generation for Delinquency Management

A new batch named **C1-MTOMT** is introduced in this release. This batch is used to create the termination and cancellation outbound messages whenever the termination or cancellation process is initiated for the individual memberships through the **Delinquency Management** feature. It creates the termination and cancellation outbound messages using the given outbound message types for the individual memberships which have the given status and status reasons. In addition, it adds a log for the outbound message in the respective individual membership and delinquency process.

Note: Once an outbound message is created, a record is added for the individual membership and delinquency event combination in the **C1_DELIN_PROC_EVT_NOTIF** table where the **DP_EVT_NOTIF_TYPE_FLG** column is set to **Outbound Message** and the **DP_EVT_NOTIF_ID** column is set to outbound message ID.

This batch is a multi-threaded batch. The multi-threading is based on membership ID and chunks for multi-threading are created based on numerical distribution of membership ID. You can specify the following parameters while executing this batch:

Parameter Name	Mandatory (Yes or No)	Description
Outbound Message Type For Termination	Yes	Used to indicate the outbound message type using which you want to create the termination outbound messages. **Note:** You must specify an outbound message type where the outbound message business object is set to **C1-MemberTermRequest**.

Parameter Name	Mandatory (Yes or No)	Description
Outbound Message Type For Cancellation	Yes	Used to indicate the outbound message type using which you want to create the cancellation outbound messages. **Note:** You must specify an outbound message type where the outbound message business object is set to **C1-MemberCancRequest**.
Source System	No	Used to indicate the external source system to which you want to send the termination or cancellation outbound messages. **Note:** You must specify a source system which is already defined in the **C1-SourceSystemLookup** extendable lookup.
Membership Active Status	Yes	Used to indicate the status code of the **Active** status of an individual membership. **Note:** You must specify a status code which is already defined in the lifecycle of the **C1-IndMembership** business object.
Membership Awaiting Termination Status Reason	Yes	Used to indicate the status reason which is specified in the **Membership Termination Reason for Delinquency** attribute of the delinquency process preference. **Note:** You must specify a status reason which is already defined for the **Active** status of the **C1-IndMembership** business object.
Membership Awaiting Cancellation Status Reason	Yes	Used to indicate the status reason which is specified in the **Awaiting Membership Cancellation Reason** attribute of the delinquency process preference. **Note:** You must specify a status reason which is already defined for the **Active** status of the **C1-IndMembership** business object.
Thread pool name	No	Used to specify the thread pool on which you want to execute the batch.
Chunk Size	Yes	Used to specify the number of outbound messages you want to create in each work unit.

Delinquency Grace Period Business Rule

Oracle Revenue Management and Billing enables you to define business rules for determining the grace period during the delinquency management. The insurance company may offer different grace period to different individuals based on certain parameters, such as division, customer class, region, and jurisdiction in which the individual customer has availed for the health plan coverage. The system will then accordingly use the grace period to calculate the following:

- Trigger dates for the delinquency events
- Termination request date when you want to send the outbound message to the external system requesting to terminate the individual membership

The system considers the delinquency grace period business rules to determine the grace period only when the grace period source is set to **Algorithm** in the respective delinquency process type. If the grace period source is set to **Algorithm**, the system executes the **C1-DERGRBR** algorithm attached to the **Grace Period Derivation** system event of the delinquency process type when you execute the **C1-DPEVL** batch. For more information on how the grace period is derived, refer to the respective algorithm description in the Delinquency Process Type Algorithms section.

While defining the delinquency grace period business rule criteria, you can use the parameters which are defined on the following entities:

- Account
- Customer Class
- Delinquency Event Type
- Division
- Health Plan
- Health Product
- Membership
- Membership Person
- Policy
- Policy Plan

You can define, edit, and delete a delinquency grace period business rule through the **Business Rule** screen. While defining a delinquency grace period business rule, you need to set its category to **Delinquency Grace Period Business Rule** and specify the following details:

- **Criteria Description** – Used to indicate the business criteria that an individual membership must satisfy in order to derive the grace period from the business rule.
- **Effective Date** – Used to specify the date from when the business rule is effective.
- **Status** – Used to indicate the status of the business rule. The valid values are **Active** and **Inactive**.
- **Grace Period (Days)** – Used to specify the grace period in days.

You can view the delinquency grace period business rules through the **Business Rule** screen.

Delinquency Event Attribute Business Rule

Oracle Revenue Management and Billing enables you to define business rules for determining the waiting time (in days or months) before triggering a delinquency event in the delinquency process. The insurance company may offer different waiting time for different delinquency event types (such as sending letters, sending To Do notification, or initiating the termination request) based on certain parameters, such as division, customer class, region, and jurisdiction in which the individual customer has availed for the health plan coverage. The system will then accordingly use the wait days or the wait months and the day of month (depending on the wait period that is configured) to calculate the trigger dates for the delinquency events.

The system considers the delinquency event attributes business rules to determine the wait period only when the wait period source is set to **Algorithm** in the respective delinquency process type. If the wait period source is set to **Algorithm**, the system executes the **C1-DERWDBR** algorithm attached to the **Wait Days Derivation** system event of the delinquency process type when you execute the **C1-DPMON** batch. For more information on how the wait period is derived, refer to the respective algorithm description in the [Delinquency Process Type Algorithms](#) section.

While defining the delinquency event attributes business rule criteria, you can use the parameters which are defined on the following entities:

- Account
- Customer Class
- Delinquency Event Type
- Division
- Health Plan
- Health Product
- Membership
- Membership Person
- Policy
- Policy Plan

You can define, edit, and delete a delinquency event attributes business rule through the **Business Rule** screen. While defining a delinquency event attributes business rule, you need to set its category to **Delinquency Event Attributes Business Rule** and specify the following details:

- **Criteria Description** – Used to indicate the business criteria that an individual membership must satisfy in order to derive the wait period from the business rule.
- **Effective Date** – Used to specify the date from when the business rule is effective.
- **Status** – Used to indicate the status of the business rule. The valid values are **Active** and **Inactive**.
- **Wait Days –** Used to indicate the number of days for which you want to delay a delinquency event in a delinquency process.

- **Wait Month** – Used to indicate the number of months for which you want to delay a delinquency event in a delinquency process.
- **Wait Day of Month** – Used to indicate the day of the month when you want to trigger a delinquency event of a delinquency process.
- **Send Membership Level Notification** – This parameter is not applicable for the fully insured individual business.

You can view the delinquency event attributes business rules through the **Business Rule** screen.

Delinquency Termination Date Rule Business Rule

Oracle Revenue Management and Billing enables you to define business rules for determining the termination date rule during the delinquency management. The termination date rule is used to derive the termination effective date for an individual membership. The insurance company may derive different termination date rule for different individual memberships based on certain parameters, such as division, customer class, region, and jurisdiction in which the individual customer has availed for the health plan coverage. The system will then accordingly use the termination date rule to calculate the termination effective date for the delinquency event type that requests to terminate the individual memberships billed to the delinquent account.

The system always derives the termination date rule through a delinquency termination date rule business rule. The system executes the **C1-DERTRBR** algorithm attached to the **Termination Date Rule Derivation** system event of the respective delinquency process type when you execute **C1-DPEVL** batch. For more information on how the termination date rule is derived, refer to the respective algorithm description in the [Delinquency Process Type Algorithms](#) section.

While defining the delinquency termination date rule business rule criteria, you can use the parameters which are defined on the following entities:

- Account
- Customer Class
- Delinquency Event Type
- Division
- Health Plan
- Health Product
- Membership
- Membership Person
- Policy
- Policy Plan

You can define, edit, and delete a delinquency termination date rule business rule through the **Business Rule** screen. While defining a delinquency termination date rule business rule, you need to set its category to **Delinquency Termination Date Rule Business Rule** and specify the following details:

- **Criteria Description** – Used to indicate the business criteria that an individual membership must satisfy in order to derive the termination date rule from the business rule.

- **Effective Date** – Used to specify the date from when the business rule is effective.
- **Status** – Used to indicate the status of the business rule. The valid values are **Active** and **Inactive**.
- **Termination Date Rule** – Used to indicate the termination date rule using which you want to calculate the termination date. The valid values are:
  - Latest Billed Coverage End Date
  - Latest Due Coverage End Date
  - X days After Paid Through Date
  - X Months After Paid Through Date
  - Month End of Grace Period Start
  - Month End of Termination Letter Creation
  - Month End of Termination Request Event
  - X days After Grace End Date
  - X days After Termination Request Event

For more information on how the above termination date rules are used to calculate the termination date, refer to the [Delinquency Event Activation](#) section.

You can view the delinquency termination date rule business rules through the **Business Rule** screen.

Delinquency Miscellaneous Options Business Rule

Oracle Revenue Management and Billing enables you to define business rules for determining the following during the delinquency management:

- Whether the trigger dates of delinquency events should be recalculated when the hold on the delinquency process is released
- Whether the trigger dates of delinquency events should be recalculated when the delinquency process is resumed after cancellation
- Whether you want to send an outbound message to the enrollment system for terminating the individual memberships billed to the delinquent account immediately, at the end of the month when the termination event is triggered, or after X number of days (i.e. termination wait days) from the termination event trigger date
- The number of days that you want to add to the base date while calculating the termination effective date

The insurance company may offer different delinquency arrangements to different individuals based on certain parameters, such as division, customer class, region, and jurisdiction in which the individual customer has availed for the health plan coverage. The system will then accordingly use the business rule to calculate the following:

- Trigger dates for the delinquency events
- Termination request date when you want to send the outbound message to the external system requesting to terminate the individual membership

- Termination effective date for the delinquency event type that requests to terminate the individual memberships billed to the delinquent account

The system considers the delinquency miscellaneous options business rules to determine the following:

- The value for the **Trigger Date Recalculation On Hold Release** flag when the trigger date recalculation on hold release source is set to **Algorithm** in the respective delinquency process type. If the trigger date recalculation on hold release source is set to **Algorithm**, the system executes the **C1-DERRTDBR** algorithm attached to the **Trigger Date Recalculation Option Derivation** system event of the delinquency process type when you execute the **C1-DPMON** batch. For more information on how the value is derived for the **Trigger Date Recalculation On Hold Release** flag, refer to the respective algorithm description in the [Delinquency Process Type Algorithms](#) section.

- The value for the **Trigger Date Recalculation On Resume** flag when the trigger date recalculation on resume source is set to **Algorithm** in the respective delinquency process type. If the trigger date recalculation on resume source is set to **Algorithm**, the system executes the **C1-DERRTDBR** algorithm attached to the **Trigger Date Recalculation Option Derivation** system event of the delinquency process type when you execute the **C1-DPMON** batch. For more information on how the value is derived for the **Trigger Date Recalculation On Resume** flag, refer to the respective algorithm description in the [Delinquency Process Type Algorithms](#) section.

- The value for the **Trigger Termination Process** flag and the termination wait days when the termination reinstatement configuration source is set to **Algorithm** in the respective delinquency process type. If the termination reinstatement configuration source is set to **Algorithm**, the system executes the **C1-DERTROBR** algorithm attached to the **Termination and Reinstatement Configuration Derivation** system event of the delinquency process type when you execute the **C1-DPMON** batch. For more information on how the value is derived for the **Trigger Termination Process** flag, refer to the respective algorithm description in the [Delinquency Process Type Algorithms](#) section.

- The number of days that you want to add to the base date while deriving the termination effective date. The system executes the **C1-DETTERMDT** algorithm attached to the **Event Activation** system event of the respective delinquency event type whenever the delinquency event is triggered manually or automatically on the trigger date. For more information on how the following parameters of the business rule are used to derive the termination effective date, refer to the respective algorithm description in the [Delinquency Event Activation](#) section:

  - Add Days to Coverage End Date

  - Add Days to Paid Through Date

  - Add Months to Paid Through Date

  - Add Days to Grace End Date

  - Add Days to Date of Processing of Termination Request

While defining the delinquency miscellaneous options business rule criteria, you can use the parameters which are defined on the following entities:

- Account
- Customer Class
- Delinquency Event Type

- Division
- Health Plan
- Health Product
- Membership
- Membership Person
- Policy
- Policy Plan

You can define, edit, and delete a delinquency miscellaneous options business rule through the **Business Rule** screen. While defining a delinquency miscellaneous options business rule, you need to set its category to **Delinquency Miscellaneous Options Business Rule** and specify the following details:

- **Criteria Description** – Used to indicate the business criteria that an individual membership must satisfy in order to derive the required parameters from the business rule.
- **Effective Date** – Used to specify the date from when the business rule is effective.
- **Status** – Used to indicate the status of the business rule. The valid values are **Active** and **Inactive**.
- **Trigger Date Recalculation On Hold Release** – Used to indicate whether the trigger dates of delinquency events should be recalculated when the hold on the delinquency process is released. The valid values are – **Yes** and **No**.
- **Trigger Date Recalculation On Resume** – Used to indicate whether the trigger dates of delinquency events should be recalculated when the delinquency process is resumed after cancellation. The valid values are – **Yes** and **No**.
- **Trigger Termination Process** – Used to indicate when you want the system to send an outbound message to the enrollment system for terminating the individual memberships billed to the delinquent account. The valid values are:
  - o **Immediate** – Used when you want the system to send an outbound message for termination on the same day when the termination event is triggered.
  - o **Month End** - Used when you want the system to send an outbound message for termination at the end of the month when the termination event is triggered.
  - o **After X Days** - Used when you want the system to send an outbound message for termination after X number of days from the date when the termination event is triggered.
- **Termination Wait Days** – Used to specify the number of days after which an outbound message for termination should be sent to the enrollment system. This parameter is required only when the **Trigger Termination Process** parameter in the business rule is set to **After X Days**.
- **Add Days to Coverage End Date** – Used to specify the number of days that you want to add to coverage end date in order to derive termination effective date.
- **Add Days to Paid Through Date** - Used to specify the number of days that you want to add to the paid through date (of the account) in order to derive termination effective date.
- **Add Months to Paid Through Date** - Used to specify the number of months that you want to add to the paid through month (derived from the paid through date of the account) in order to derive termination effective date.

- **Add Days to Grace End Date** - Used to specify the number of days that you want to add to grace end date in order to derive termination effective date.

- **Add Days to Date of Processing of Termination Request** - Used to specify the number of days that you want to add to termination request date in order to derive termination effective date.

Note: The **Allow Automatic Reinstatement**, **Maximum Automatic Reinstatement Count**, **Trigger Reinstatement Process**, **Reinstatement Wait Days**, **Reinstatement Threshold (%)**, **Days Since Termination**, and **Member Notification Days** parameters in the delinquency miscellaneous options business rule are not applicable for the fully insured individual business.

You can view the delinquency miscellaneous options business rules through the **Business Rule** screen.

Enhancements (Specific to Insurance)

This section lists the enhancements made to the following features which can be used in the health insurance domain:

- Inbound Message
- Offset Request
- Refund/Write Off Request
- Payment Request
- Customer 360° View
- Binder Payment Preference (Field Mapping)
- Health Product 360° View
- Reconciliation
- Membership
- Invoice Request
- Individual Health Insurance Billing
- Individual Health Insurance Pricing
- Field Mapping
- Business Rules
- Parameter
- Collection Class
- Customer Class
- Adjustments

Inbound Message

The following changes are made to the Inbound Message feature:

- The **C1-HCInboundMessage** business object is enhanced to maintain skip auto maintenance information for an account.
- The following tags are newly introduced in the health care inbound message schema:

Parent Tag	New Tags
<accountData>	<skipAutoMaintenanceSw></skipAutoMaintenanceSw> <dragDays></dragDays>
<personData>	<collectionClass></collectionClass> <lastCreditReviewDate></lastCreditReviewDate> <dragDays></dragDays> <advanceDepositAmount></advanceDepositAmount> <advanceDepositGraceDays></advanceDepositGraceDays> <advanceDepositLevel></advanceDepositLevel>
<address>	<addressStatus></addressStatus>
<memberData>	<sequence></sequence> <repriceAfterDate></repriceAfterDate>
<personAddressOverride>	<addressStatus></addressStatus>
<policyData>	<advanceDepositApplSW></advanceDepositApplSW> <advanceDepositAmount></advanceDepositAmount> <advanceDepositGraceDays></advanceDepositGraceDays> <advanceDepositAccountIdentifierType></advanceDepositAccountIdentifierType> <advanceDepositAccountIdentifierValue></advanceDepositAccountIdentifierValue>

- The following tag is renamed in the health care inbound message schema:

Parent Tag	Old Tag	New Tag
<personAddressOverride>	<postal>	<zip>

- The **C1-HCProdHlthPlnInbound** business object is enhanced to maintain the Health Insurance Oversight System (HIOS) ID for a health plan. The system allows you to specify the same HIOS ID for multiple health plans provided their coverage period is different and not overlapping.

- The following tag is newly introduced in the health product and plan inbound message schema:

Parent Tag	New Tags
<healthPlanData>	<hiosId></hiosId>

Offset Request

The following changes are made to the Offset Request feature:

- A new field named **Offset Category** is introduced in the **C1-OffsetRequestType** business object. The offset requests are now classified under the following two categories:

    - **Regular** – The regular or traditional offset requests are manual offset requests. You can create a manual offset request using an offset request type where the offset category is set to **Regular**.

    - **Auto Maintenance** – The auto maintenance offset requests are automatic offset requests. You can create an automatic offset request using an offset request type where the offset category is set to **Auto Maintenance**.

- The **Offset Request** screen is enhanced to support an auto maintenance offset request. A new zone named **Auto Maintenance Financial Transactions** appears in the **Main** tab of the **Offset Request** screen. It lists the financial transactions of the account for which the corresponding offset request adjustments are created during the auto maintenance process.

- While viewing the details of an auto maintenance offset request, the system displays the **Auto Maintenance Financial Transactions** zone instead of the **Selected Bills** and **Offset Request Adjustments** zones in the **Main** tab of the **Offset Request** screen. The system also displays the account information in the **Main** section of the **Offset Request** zone.

- If you set the **Skip Auto Maintenance for Accounts with Open Offset Request (Y/N)** parameter to **N** while executing the **C1-AUTOM** batch, the system will create an auto maintenance offset request for an account even if the regular offset request exists for the account in the non-final status. The offset request feature is enhanced to handle such scenarios.

    The system will throw an appropriate error message when you process the manual offset request containing bills or bill line items which are closed during the auto maintenance of the account.

Refund/Write Off Request

The following changes are made to the Refund/Write Off Request feature:

- If you set the **Skip Auto Maintenance for Accounts with Open Refund or Write Off Request (Y/N)** parameter to **N** while executing the **C1-AUTOM** batch, the system will create an auto maintenance offset request for an account even if the refund or write off request exists for the account in the non-final status. The refund/write off request feature is enhanced to handle such scenarios.

    The system will throw an appropriate error message when you process the refund/write off request containing bills or bill line items which are closed during the auto maintenance of the account.

Payment Request

The following changes are made to the Payment Request feature:

- If you set the **Skip Auto Maintenance for Accounts with Open Payment, Payment Transfer Request (Y/N)** parameter to **N** while executing the **C1-AUTOM** batch, the system will create an auto maintenance offset request for an account even if the payment creation or payment transfer request exists for the account in the non-final status. The payment request feature is enhanced to handle such scenarios.

  The system will apply the payment on an excess credit contract of the account when you process the payment creation or payment transfer request containing bills or bill line items which are closed during the auto maintenance of the account.

Customer 360° View

The following changes are made to the Customer 360° View feature:

- Now, you can view the skip auto maintenance information of the account in the **Account Information** zone of the **Account** tab. Note that the **Skip Auto Maintenance** field appears for the account only when it is set to **Y**.

- You can now view the person's collection details, such as collection class, postpone credit review date, and the last credit review date. A new section named **Person Collection** is introduced in the **Person Information** zone of the **Person** tab.

- You can now view the information of a delinquency process when the delinquency process is created for an overdue bill of the account. A new column named **Delinquency Process Information** is introduced in the **Open Bills** zone of the **Account** tab.

Binder Payment Preference (Field Mapping)

The following attributes are newly introduced for the **Binder Payment** preference category:

- **Membership Status Reason when Binder Payment Cancelled** – Used to specify the membership status reason code that you want to use while creating a To Do notification during the binder payment cancellation.

- **To Do Type for Binder Payment Cancelled** – Used to specify the To Do type using which you want to create a To Do during the binder payment cancellation. Note that the **C1-BPCNC** To Do type is shipped with the product.

Health Product 360° View

The following changes are made to the Health Product 360° View feature:

- The **C1-HealthPlan** business object is enhanced to maintain the HIOS ID for a health plan. The system allows you to specify the same HIOS ID for multiple health plans provided their coverage period is different and not overlapping.

- Now, you can search for a health plan using its HIOS ID. A new field named **HIOS ID** is introduced in the **360° Search** zone when you search using the **Health Plan Details** option from the **Search By** list.

- You can view the HIOS ID of the health plan in the **Plan Information** zone of the **Health Plan 360° Information** screen. Note that this field appears only when the HIOS ID is available for the health plan.

Reconciliation

The following changes are made to the Reconciliation feature:

- Now, the Reconciliation feature is enhanced to support the fully insured individual business.

- The **C1-PayInstruction** business object is enhanced to store the following information in a pay instruction for the fully insured individual business:

  - Health Plan Code

  - HIOS ID

  - Payor Identifier Type

  - Payor Identifier Value

  - Payment Type

  - Price Item

- The **Health Plan** and **Payment Type** fields are added in the **Pay Instructions** zone of the **Reconciliation** screen. These fields appear only when you are viewing the details of a reconciliation object which is created for the fully insured individual business. However, the **Policy Number** field appears in the **Pay Instructions** zone when you are viewing the details of a reconciliation object which is created for the fully insured group business.

- The **Payor Identifier Type**, **Payor Identifier**, **Health Plan**, **HIOS ID**, and **Payment Type** columns are added in the **Pay Instructions** zone of the **Reconciliation** screen. These columns appear only when you are viewing the details of a reconciliation object which is created for the fully insured individual business. However, the **Policy Number** and **Plan Number** columns appear in the **Pay Instructions** zone when you are viewing the details of a reconciliation object which is created for the fully insured group business.

- The **Health Plan**, **HIOS ID**, **Payment Type**, **Payor Identifier Type**, and **Payor Identifier** fields are added in the **Main** section of the **Pay Instruction** zone on the **Pay Instruction** screen. These fields appear only when you are viewing the details of a pay instruction which is created for the fully insured individual business. However, the **Policy Number** and **Plan Number** fields appear in the **Pay Instruction** zone when you are viewing the details of a pay instruction which is created for the fully insured group business.

- The **Health Plan**, **HIOS ID**, **Payment Type**, **Payor Identifier Type**, and **Payor Identifier** columns are added in the **Discrepancy Line Items** zone of the **Discrepancy Report** screen. These columns appear when you are viewing the discrepancy report for the fully insured individual business. However, the **Policy Number** and **Plan Number** columns appear when you are viewing the discrepancy report for the fully insured group business.

- Earlier, when a pay instruction was canceled, the system did not check whether there is any related pay instruction for it due to any take back or partial payment scenarios. For example, in the month of Jan 2023, a pay instruction (i.e. P1) of $300 was received for the A1 account which was reconciled against its B1 ($100) and B2 ($100) line items. And, in the month of Feb 2023, a negative pay instruction (i.e. P2) of -$100 received for the A1 account was reconciled against its Jan month's open pay instruction (i.e. P1). If, due to any reason, the P2 pay instruction was canceled, the system did not automatically reopen the P1 pay instruction and vice-versa. From this release onwards, this cancellation issue is handled for the take back and partial payment scenarios. Now, while canceling a pay instruction, the system checks whether there is any related pay instruction for it. If so, the system first accordingly reopens the related pay instruction and then cancels the primary pay instruction.

Membership

The following changes are made to the Membership screen:

- You can now search for a health plan using its HIOS ID. A new field named **HIOS ID** is added in the **Health Plan Search** window.

- Appropriate log entries are added in the **Log** tab when an invoice request is created for the individual membership's account.

Invoice Request

The following changes are made to the Invoice Request feature:

- This feature is enhanced to support the fully insured individual business along with other lines of business.

- A new value named **C1IM** is added in the **EXT_REF_TYPE_FLG** lookup field. Therefore, now, while searching for an invoice request, the following two values appear in the **External Reference Type** list:

    o Policy

    o Membership

- While creating an invoice request for an individual membership's account, the system considers the individual membership as the external reference and sets the external reference type of the invoice request to **Membership**.

- While viewing the details of an invoice request which is created for the fully insured individual business, the system does the following:

    o Displays the individual membership details in the **External Reference** section of the **Invoice Request** zone

    o Hides the **Account Information** column in the **Invoice Request Details** zone

    However, while viewing the details of an invoice request which is created for the fully insured group business, the system does the following:

    o Displays the policy details in the **External Reference** section of the **Invoice Request** zone

    o Shows the **Account Information** column in the **Invoice Request Details** zone

Individual Health Insurance Billing

The following changes are made to the Individual Health Insurance Billing feature:

- Until now, the system enabled you to create multiple accounts with a unique invoice type (i.e. account type) for a member who is financially responsible for an individual membership. Now onwards, you can create multiple accounts with the same invoice type (i.e. account type) for a member who is financially responsible for an individual membership.

- The charges of an individual membership are billed using the invoice type priority specifications given for the respective price item in the respective pricing rule type. Until now, as you were able to create only one account of a particular invoice type (i.e. account type), the system was able to derive the account of the member who is financially responsible for an individual membership using the given invoice type priority. But now, as the system enables you to create multiple accounts with the same invoice type, you need to specify the required account to which the individual membership should be billed as characteristic on the individual membership.

  While fetching the characteristics on the individual membership, the system considers the characteristic types which are specified in the **Account Identifier Type Char Type** and **Account Identifier Value Char Type** option types of the **C1-ASOBLLNG** feature configuration. Therefore, you need to define the account details for the individual membership using the characteristic types which are specified in the **Account Identifier Type Char Type** and **Account Identifier Value Char Type** option types through a health care inbound message.

Individual Health Insurance Pricing

The following changes are made to the Individual Health Insurance Pricing feature:

- Now, the system enables you to define separate maximum number of dependents for a minor and non-minor main subscriber. A new field named **Minor Maximum Number of Dependents** is available while defining a fully insured pricing business rule for the fully insured individual business. If the main subscriber is a non-minor, the system will use the maximum number of dependents specified in the **Maximum Dependents** column of the business rule. However, if the main subscriber is a minor, the system will use the maximum number of dependents specified in the **Minor Maximum Number of Dependents** column of the business rule.

  Now, the following algorithms will recalculate the age of main subscriber and will use the maximum number of dependents from the **Minor Maximum Number of Dependents** or **Maximum Dependents** field, respectively, depending on whether the main subscriber is a minor or non-minor:

  - C1_AGEBDELG
  - C1-DERSUBTR
  - C1-MEMRLAUD

- The **No Of Days** parameter is removed from the **C1-PPRAMAGE** algorithm type. Instead, a new field named **Age Recalculation Number of Days** is available in the **Age Based Specific Additional Data** section when you define an age based pricing rule type using the **C1-PrcRuleTypeAgeBased** business object. You can define the number of days in the **Age Recalculation Number of Days** field.

Field Mapping

The following changes are made to the Field Mapping feature:

- The following new preference categories are introduced in this release:
    - Delinquency Process
    - Delinquency Process Type
    - Membership Repricing Reasons
    - Routing Method – Contact Method Mapping
- While creating a preference using the **Delinquency Process Type** preference category, you can specify the following attributes:
    - **Active Individual Delinquency** – Used to indicate the delinquency process type using which you want to create the delinquency process when the main customer of the account, for which you want to create the delinquency process, is an active customer. An individual is deemed to be an active customer when at least one membership of the individual customer is in the **Active** status.
    - **Active Group Customers – Account Level** – This attribute is not applicable for the fully insured individual business.
    - **Active Group Customers – Person Level** - This attribute is not applicable for the fully insured individual business.
    - **Benefit Sub Type** – Used to indicate the benefit sub type. The system considers the benefit (i.e. subsidy) charges of the specified benefit sub type to derive the appropriate delinquency process type for the account. The system considers this attribute when the **Active Individual Delinquency** attribute is not defined and when paid through date is present on the account.
    - **Effectuated Individual Delinquency for Ancillary** - Used to indicate the delinquency process type using which you want to create the delinquency process when the following conditions are met:
        - The **Active Individual Delinquency** attribute is not defined
        - Paid through date is present on the account
        - Earliest unpaid coverage period does not include subsidy charges
        - Individual is enrolled to a health plan which belongs to a non-medical health product
    - **Inactive Group Customers – Account Level** - This attribute is not applicable for the fully insured individual business.

- o **Inactive Group Customers – Person Level** - This attribute is not applicable for the fully insured individual business.

- o **Effectuated Individual without Benefit** - Used to indicate the delinquency process type using which you want to create the delinquency process when the following conditions are met:

  - ▪ The **Active Individual Delinquency** attribute is not defined

  - ▪ Paid through date is present on the account

  - ▪ Earliest unpaid coverage period does not include subsidy charges

  - ▪ Individual is enrolled to a health plan which belongs to a medical health product

- o **Effectuated Individual with Benefit** – Used to indicate the delinquency process type using which you want to create the delinquency process when the following conditions are met:

  - ▪ The **Active Individual Delinquency** attribute is not defined

  - ▪ Paid through date is present on the account

  - ▪ Earliest unpaid coverage period includes subsidy charges

- o **Non-Effectuated Individual Delinquency** – Used to indicate the delinquency process type using which you want to create the delinquency process when the **Active Individual Delinquency** attribute is not defined and the paid through date is not present on the account.

- o **Termed Individual Delinquency** - Used to indicate the delinquency process type using which you want to create the delinquency process when the main customer of the account, for which you want to create the delinquency process, is an inactive customer. An individual is deemed to be an inactive customer when all memberships of the individual customer is in the **Terminated** status.

- While creating a preference using the **Bill Route Method - Contact Method Mapping** (**C1_BillRouteCntMethMap**) business object, you can specify the following attributes:

  - o **Bill Routing Method** – Used to indicate the bill routing method for which you want to map the preferred contact method. Here, the list includes all the values which are defined in the **BILL_RTG_METH_FLG** lookup field.

  - o **Preferred Contact Method** – Used to indicate the preferred contact method for the bill routing method. Here, the list includes all the values which are defined in the **CONTACT_METH_FLG** lookup field.

If a preference is defined using the **C1_BillRouteCntMethMap** business object, the system will derive the preferred contact method for the bill routing method from the preference while creating a customer contact for the delinquent account.

Business Rules

The following changes are made to the Business Rules feature:

- The following new business rule categories are introduced in this release:
    - Delinquency Event Attributes Business Rule
    - Delinquency Grace Period Business Rule
    - Delinquency Miscellaneous Options Business Rule
    - Delinquency Termination Date Rule Business Rule

- Earlier, the **Business Rule Type** zone of the **Business Rule Type** screen was not operational. Now, while clicking the **Broadcast** icon corresponding to a business rule type in the **Search Business Rule Type** zone, the details of the business rule type appear in the **Business Rule Type** zone.

- The **Business Rule Type** screen enables you to maintain business rule criteria for the newly introduced business rules categories (listed above).

Parameter

The following changes are made to the Parameter feature:

- The following new source entities are introduced in this release:
    - Customer Class
    - Delinquency Event Type
    - Division

- The system enables you to define parameters which can be used while defining or editing a business rule criteria. A new option named **Business Rule Eligibility Criteria** is added to the **Parameter Usage** section. It appears only when the source entity is set to either of the following:
    - Account
    - Customer Class
    - Delinquency Event Type
    - Division
    - Health Plan
    - Health Product
    - Membership
    - Membership Person
    - Policy
    - Policy Plan

Collection Class

The following changes are made to the Collection Class feature:

- The following collection methods are introduced in this release:

    o **Self-Control Delinquency** – Used when you want to create a separate delinquency process for the accounts which belong to the collection class. These accounts will not be considered during the delinquency management of the parent person.

    o **Parental Delinquency** – This option is not applicable for the fully insured individual business.

Customer Class

The following changes are made to the Customer Class feature:

- Two new algorithms named **C1-CDPPAYCAN** and **C1-RODPDUEPX** are introduced in this release. They should be attached to the **Payment Cancellation** system event of a customer class to resume the delinquency process for the respective accounts on the payment cancellation. For more information about the algorithms, refer to the Resuming a Delinquency Process on Payment Cancellation section.

Adjustments

The following changes are made to the Adjustments feature:

- Two new algorithms named **C1-DPADJCAN** and **C1-RODPONADX** are introduced in this release. They should be attached to the **Adjustment Cancellation** system event of an adjustment type to resume the delinquency process for the account whose corresponding adjustment is canceled. For more information about the algorithms, refer to the Resuming a Delinquency Process on Adjustment Cancellation section.

User Interface (UI) Level Changes

The following table lists changes made to the existing screens in Oracle Revenue Management and Billing:

Screen Name (in 5.1.0.0.0)	Changes
Offset Request Type (Used for Adding, Editing, and Copying)	The following change is made to this screen: • A new field named **Offset Category** is added in the **Main** section.
Offset Request Type	The following change is made to this screen: • The **Offset Category** field is added in the **Main** section of the **Offset Request Type** zone.
Offset Request (Used for Viewing)	The following changes are made to this screen: • The **Selected Bills** and **Offset Request Adjustments** zones do not appear while viewing the details of an auto maintenance offset request. Instead, a new zone named **Auto Maintenance Financial Transactions** appears in the **Main** tab. • The **Account Information** field appears in the **Main** section of the **Offset Request** zone while viewing the details of an auto maintenance offset request.
Customer 360° Information	The following change is made to this screen: • A new field named **Skip Auto Maintenance** appears in the **Account Information** zone of the **Account** tab when it is set to **Y** for the respective account.
Health Product 360° View	The following change is made to this screen: • The **HIOS ID** field is newly introduced in the **360° Search** zone when you search using the **Health Plan Details** option from the **Search By** list.
Health Plan 360° Information	The following change is made to this screen: • The **HIOS ID** field is added in the **Plan Information** zone. It appears only when the HIOS ID is available for the health plan.
Reconciliation Type (Used for Adding and Editing)	The following change is made to this screen: • A new field named **Reconciliation Category** is added in the **Main** section.

Screen Name (in 5.1.0.0.0)	Changes
Reconciliation Type	The following changes are made to this screen: • The **Reconciliation Category** column is added in the **Reconciliation Type List** zone. • The **Reconciliation Category** field is added in the **Main** section of the **Reconciliation Type** zone.
Reconciliation (Used for Viewing)	The following changes are made to this screen: • The **Health Plan** and **Payment Type** fields are added in the **Pay Instructions** zone. These fields appear only when you are viewing the details of a reconciliation object which is created for the fully insured individual business. • The **Member Identifier Type** and **Member Identifier** fields are renamed to **Subscriber Identifier Type** and **Subscriber Identifier**, respectively, in the **Pay Instructions** zone. • The **Member Name**, **Member Identifier Type**, and **Member Identifier** columns are renamed to **Subscriber Name**, **Subscriber Identifier Type**, and **Subscriber Identifier**, respectively, in the **Pay Instructions** zone. • The **Payor Identifier Type**, **Payor Identifier**, **Health Plan**, **HIOS ID**, and **Payment Type** columns are added in the **Pay Instructions** zone. These columns appear only when you are viewing the details of a reconciliation object which is created for the fully insured individual business.
Pay Instruction (Used for Viewing)	The following changes are made to this screen: • The **Health Plan**, **HIOS ID**, **Payment Type**, **Payor Identifier Type**, and **Payor Identifier** fields are added in the **Main** section of the **Pay Instruction** zone when you are viewing the details of a pay instruction which is created for the fully insured individual business. • The **Member Name**, **Member Identifier Type**, and **Member Identifier** fields are renamed to **Subscriber Name**, **Subscriber Identifier Type**, and **Subscriber Identifier**, respectively, in the **Main** section of the **Pay Instruction** zone.

Screen Name (in 5.1.0.0.0)	Changes
Discrepancy Report (Used for Viewing)	The following changes are made to this screen: • The **Health Plan**, **HIOS ID**, **Payment Type**, **Payor Identifier Type**, and **Payor Identifier** columns are added in the **Discrepancy Line Items** zone. These columns appear when you are viewing the discrepancy line item for the fully insured individual business. • The **Member Name**, **Member Identifier Type**, and **Member Identifier** columns are renamed to **Subscriber Name**, **Subscriber Identifier Type**, and **Subscriber Identifier**, respectively, in the **Discrepancy Line Items** zone.
Health Plan Search (Window)	The following change is made to this screen: • The **HIOS ID** field is added in the **Search Criteria** section.
Invoice Request (Used for Searching)	The following change is made to this screen: • Now, the following two values appear in the **External Reference Type** list – Policy and Membership.
Invoice Request (Used for Viewing)	The following change is made to this screen: • The **Account Information** column does not appear in the **Invoice Request Details** zone when you are viewing the details of an invoice request which is created for the fully insured individual business. It appears only when the invoice request is created for the fully insured group business.
Business Rule (Used for Searching)	The following change is made to this screen: • The **Minor Maximum Number of Dependents** column is added to the **Search Results** section when you search for a fully insured pricing business rule in the **Search Business Rule** zone.
Fully-Insured Pricing Business Rule (Used for Adding and Editing)	The following change is made to this screen: • The **Minor Maximum Number of Dependents** column is added to the grid.
Pricing Rule Type (Used for Adding and Editing)	The following change is made to this screen: • The **Age Recalculation Number of Days** field is available in the **Age Based Specific Additional Data** section when you define or edit an age based pricing rule type.

Screen Name (in 5.1.0.0.0)	Changes
Pricing Rule Type (Used for Viewing)	The following change is made to this screen: • The **Age Recalculation Number of Days** field is added in the **Age Based Specific Additional Data** section of the **Pricing Rule Type** zone.
Business Rule Type (Used for Viewing)	The following changes are made to this screen: • The **Main**, **Parameters**, **Characteristics**, **Record Actions**, and **Record Information** sections are added to the **Business Rule Type** zone.
Parameter (Used for Searching)	The following changes are made to this screen: • The following options are newly added in the **Source Entity** list - Customer Class, Delinquency Event Type, and Division.

Parameter (Used for Viewing)	The following changes are made to this screen:
	
	• A new option named **Business Rule Eligibility Criteria** is added to the **Parameter Usage** section. It appears only when the source entity is set to either of the following:
	
	    o Account
	    o Customer Class
	    o Delinquency Event Type
	    o Division
	    o Health Plan
	    o Health Product
	    o Membership
	    o Membership Person
	    o Policy
	    o Policy Plan

Screen Name (in 5.1.0.0.0)	Changes

Parameter (Used for Adding and Editing)	The following changes are made to this screen:
	
	• A new option named **Business Rule Eligibility Criteria** is added to the **Parameter Usage** section. It appears only when the source entity is set to either of the following:
	
	    o Account
	    o Customer Class
	    o Delinquency Event Type
	    o Division
	    o Health Plan
	    o Health Product
	    o Membership
	    o Membership Person
	    o Policy
	    o Policy Plan

Business Rule (Used for Searching)	The following changes are made to this screen: • The **Delinquency Grace Period Business Rule**, **Delinquency Event Attributes Business Rule**, **Delinquency Termination Date Rule Business Rule**, and **Delinquency Miscellaneous Options Business Rule** options are added in the **Search By** list.

Screen Name (in 5.1.0.0.0)	Changes

Business Rule (Used for Adding)	The following changes are made to this screen:
	
	• The **Delinquency Grace Period Business Rule**, **Delinquency Event Attributes Business Rule**, **Delinquency Termination Date Rule Business Rule**, and **Delinquency Miscellaneous Options Business Rule** options are added in the **Business Rule Category** list.
	
	• The following new screens are introduced in this release:
	
	    ○ **Delinquency Event Attribute Business Rule**
	
	    ○ **Delinquency Grace Period Business Rule**
	
	    ○ **Delinquency Miscellaneous Options Business Rule**
	
	    ○ **Delinquency Miscellaneous Options Business Rule**
	
	    ○ **Delinquency Termination Date Rule Business Rule**
	
	Each screen appears when the respective category is selected from the **Business Rule Category** list.

Customer 360° Information	The following changes are made to this screen: • The **Person Collection** section is added in the **Person Information** zone of the **Person** tab. • The **Delinquency Process Information** column is added in the **Open Bills** zone of the **Account** tab.
Collection Class	The following changes are made to this screen: • The **Parental Delinquency** and **Self-Control Delinquency** options are added in the **Collection Method** list.