

Oracle® Revenue Management and Billing

Version 2.6.0.0.0

Batch Guide

Revision 9.3

E86429-01

August, 2017

Oracle Revenue Management and Billing Batch Guide

E86429-01

Copyright Notice

Copyright © 2017, Oracle and/or its affiliates. All rights reserved.

Trademark Notice

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

Intel and Intel Xeon 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, Opteron, the AMD logo, and the AMD Opteron logo are trademarks or registered trademarks of Advanced Micro Devices. UNIX is a registered trademark of The Open Group.

License Restrictions Warranty/Consequential Damages Disclaimer

This software and related documentation are provided under a license agreement containing restrictions on use and disclosure, and are protected by intellectual property laws. Except as expressly permitted in your license agreement or allowed by law, you may not use, copy, reproduce, translate, broadcast, modify, license, transmit, distribute, exhibit, perform, publish, or display any part, in any form, or by any means. Reverse engineering, disassembly, or de-compilation of this software, unless required by law for interoperability, is prohibited.

Warranty Disclaimer

The information contained herein is subject to change without notice and is not warranted to be error-free. If you find any errors, please report them to us in writing.

Restricted Rights Notice

If this software or related documentation is delivered to the U.S. Government or anyone licensing it on behalf of the U.S. Government, the following notice is applicable:

U.S. GOVERNMENT RIGHTS

Oracle programs, including any operating system, integrated software, any programs installed on the hardware and/or documentation delivered to U.S. Government end users are “commercial computer software” pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, use, duplication, disclosure, modification, and adaptation of the programs, including any operating system, integrated software, any programs installed on the hardware and/or documentation shall be subject to license terms and restrictions applicable to the programs. No other rights are granted to the U.S. Government.

Hazardous Applications Notice

This software 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 in dangerous applications, then you shall be responsible to take all appropriate fail-safe, backup, redundancy, and other measures to ensure the safe use of this software. Oracle Corporation and its affiliates disclaim any liability for any damages caused by use of this software in dangerous applications.

Third Party Content, Products, and Services Disclaimer

This software and documentation may provide access to or information on 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. 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.

Preface

About This Document

This document provides detail information about various batches in Oracle Revenue Management and Billing.

Intended Audience

This document is intended for the following audience:

- End-users
- Implementation Team
- Consulting Team
- Development Team

Organization of the Document

The information in this document is organized into the following sections:

Section No.	Section Name	Description
Section 1	Billing	Lists and describes billing related batches.
Section 2	Payment	Lists and describes payment related batches.
Section 3	Financial Transaction	Lists and describes financial transaction related batches.
Section 4	Pricing Management	Lists and describes pricing related batches.
Section 5	Funding Request	Lists and describes funding request related batches.
Section 6	Offset Request	Lists and describes offset request related batches.
Section 7	Hold Request	Lists and describes hold request related batches.
Section 8	Upload Request	Lists and describes upload request related batches.
Section 9	Inbound Message	Lists and describes inbound message related batches.
Section 10	Payment Agreement Request	Lists and describes payment agreement request related batches.
Section 11	Accruals	Lists and describes accruals related batches.
Section 12	Earnings Credit Rate	Lists and describes earnings credit rate related batches.
Section 13	Information Lifecycle Management (ILM)	Lists and describes ILM related batches.
Section 14	Request	List and describes the request related batch.

Related Documents

You can refer to the following documents for more information:

Document	Description
<i>Oracle Revenue Management and Billing Banking User Guide</i>	Lists and describes various banking features in Oracle Revenue Management and Billing. It also describes all screens related to these features and explains how to perform various tasks in the application.

Change Log

Revision	Last Update	Updated Section	Comments
9.1	16-Sep-2017	Section 2.9: Upload Lockbox Payment and Remittance Advices (C1-PUPSG)	Added Information
		Section 13: Information Lifecycle Management (ILM)	Modified the names of sub-sections: 13.2 ILM Crawler - Bills and Bill Segments (C1-BLCRL) and 13.5 ILM Crawler - Transaction Rating Calculation Records (C1-TCCRL)
9.2	29-Sep-2017	Section 14: Request	Added Information
9.3	24-Sep-2019	Section 2.2: Automatic Payment Creation (APAYCRET)	Updated Information
		Section 2.3: Activate Automatic Payments (ACTVTAPY)	Updated Information
		Section 2.4: Extract Automatic Payments (APAYACH)	Updated Information
		Section 2.5: Freeze or Cancel Automatic Payments (APAYRA)	Updated Information
		Section 2.6: To Do Creation for Automatic Payment Exception Records (APAYUPTD)	Updated Information
		Section 2.7: Distribute and Freeze Automatic Payments (APAYDSFR)	Added Section
		Section 2.8: Create Tender Controls for Automatic Payments (BALAPY)	Updated Information

Contents

1. Billing	1
1.1 Billing (BILLING)	2
1.2 Pending Bill Generation (C1-PNDBL)	4
1.3 Bill Segment Generation (C1-BLGEN)	6
1.4 Bill Completion (C1-BLPPR).....	10
1.5 Construct Based - Pending Bill Generation (BILLOPEN)	14
1.6 Construct Based - Bill Segment Generation (BSGENREG)	16
1.7 Construct Based - Bill Completion (POSTPROC)	18
1.8 Refresh Pricing (C1-TXNRP)	21
1.9 Upload and Validate Usage Data File (C1-ODFU)	22
1.10 Billable Charge Creation (C1-ODBCH).....	24
1.11 Adhoc Billing (C1-FABL)	25
1.12 Freeze and Complete Adhoc Bills (C1-FCADH)	28
1.13 Pending Bill Segments Deletion (C1-BSEGD)	29
1.14 Pending Bill Deletion (C1-PNBD)	30
1.15 Assign Sequential Bill Numbers (ASSGNSBN)	32
2. Payment.....	33
2.1 Payment Upload (PUPL)	33
2.2 Automatic Payment Creation (APAYCRET)	34
2.3 Activate Automatic Payments (ACTVTAPY)	36
2.4 Extract Automatic Payments (APAYACH)	37
2.5 Freeze or Cancel Automatic Payments (APAYRA)	39
2.6 To Do Creation for Automatic Payment Exception Records (APAYUPTD)	40
2.7 Distribute and Freeze Automatic Payments (APAYDSFR)	41
2.8 Create Tender Controls for Automatic Payments (BALAPY)	43
2.9 Payment Request Periodic Monitor (C1-PAYRQ)	44
2.10 Upload Lockbox Payment and Remittance Advices (C1-PUPSG).....	45
3. Financial Transaction	52
3.1 Foreign Exchange Loss Gain (C1-FXLG).....	52
3.2 Assign GL Account to Financial Transaction (C1-GLASN)	53
3.3 Insert Records in CI_FTTEMP (GLASSGN1)	55
3.4 Assign GL Account to Financial Transaction (GLASSGN2)	57
3.5 GL Download Staging (GLS)	59
3.6 GL Download Extract (GLDL).....	60
4. Pricing Management.....	61
4.1 Identify Accounts for Repricing (C1-REPC1)	61

4.2	Process Repricing Request and Persist Pricing for an Account (C1-REPC2)	62
4.3	Process Fees Request and Persist Fees for an Account (C1-ACFEE)	63
5.	Funding Request	65
5.1	Funding Request Periodic Monitor (C1-FNDRQ)	65
6.	Offset Request	66
6.1	Offset Request Periodic Monitor (C1-OFSRQ)	66
7.	Hold Request	67
7.1	Hold Request Periodic Monitor (C1-HLDRQ)	67
7.2	Hold Request Monitor (C1-HLMON)	68
7.3	Delete Pending Bill Segments and Bills (C1-DELBI)	70
8.	Upload Request	72
8.1	Upload Request Periodic Monitor (C1-UPLRQ)	72
9.	Inbound Message	75
9.1	Customer Inbound Message Periodic Monitor (C1-CUSIN)	75
9.2	Membership Inbound Message Periodic Monitor (C1-MEMIN)	76
10.	Payment Agreement Request	79
10.1	Payment Agreement Request Periodic Monitor (C1-PAREQ)	79
10.2	Generate Automatic Payment for Payment Agreement Request (C1-APPAB)	80
11.	Accruals	82
11.1	Accrual Calculation (C1_ACCAL)	82
11.2	Accrual Creation (C1_ACCRE)	83
11.3	Accrual Reversal (C1_ACREV)	84
11.4	Create To Do for Account without Accrual Cycle (C1_ACELG)	85
12.	Earnings Credit Rate	86
12.1	Financial Transaction Distribution Monitor (C1-FTDRD)	86
12.2	Financial Transaction Distribution (C1-FTDTS)	88
13.	Information Lifecycle Management (ILM)	90
13.1	ILM Crawler - Adjustments (C1-ADCRL)	90
13.2	ILM Crawler - Bills and Bill Segments (C1-BLCRL)	91
13.3	ILM Crawler - Billable Charges (C1-BCCRL)	93
13.4	ILM Crawler - Transaction Feed Management (C1-FMCRL)	94
13.5	ILM Crawler - Transaction Rating Calculation Records (C1-TCCRL)	96
14.	Request	98
14.1	Generic Request Periodic Monitor (C1-GENRQ)	98

1. Billing

Oracle Revenue Management and Billing enables you to carry out billing based on the bill cycle (defined for the account). In other words, you are able to generate a bill for a bill cycle using the billable charges whose start or end date is earlier than or equal to the cut-off date. You can generate bills online or through a batch process. If you generate bills through a batch process, the batch business date is treated as the cut-off date.

Oracle Revenue Management and Billing provides you with a batch named **BILLING**, which generates the bills based on the bill cycle defined for the account. In the **BILLING** batch, multi-threading is based on account ID and chunks for multi-threading are created based on numerical distribution of account ID. This chunking logic might not be efficient while handling voluminous data. To handle high volume with better performance, you must generate bills by executing the following three batches in the specified order:

1. Pending Bill Generation (C1-PNDBL)
2. Bill Segment Generation (C1-BLGEN)
3. Bill Completion (C1-BLPPR)

Note: These three batches are primarily designed and developed for the Banking and Payments industries.

Before you execute these billing batches (i.e. **BILLING**, **C1-PNDBL**, **C1-BLGEN**, and **C1-BLPPR**), you must do the following:

- Set the **Bill Segment Freeze Option** field in the **Billing** tab of the **Installation Options** screen to either **Freeze At Will** or **Freeze At Bill Completion**.
- Set the **Billable Charges Date** option type in the **C1_FINTRANOP** feature configuration to either **S** or **E** depending on whether you want to consider the billable charges whose start or end date is earlier than or equal to the cut-off date.

Note:

If the **Billable Charges Date** option type of the **C1_FINTRANOP** feature configuration is set to **E**, the system considers the following:

>> Recurring billable charges whose start date is earlier than the cut-off date and end date is later than the cut-off date

>> Non-recurring billable charges whose end date is earlier than or equal to the cut-off date

This section provides detail information about the following batches:

- Billing (BILLING)
- Pending Bill Generation (C1-PNDBL)
- Bill Segment Generation (C1-BLGEN)
- Bill Completion (C1-BLPPR)
- Construct Based - Pending Bill Generation (BILLOPEN)

- Construct Based - Bill Segment Generation (BSGENREG)
- Construct Based - Bill Completion (POSTPROC)
- Refresh Pricing (C1-TXNRP)
- Upload and Validate Usage Data File (C1-ODFU)
- Billable Charge Creation (C1-ODBCH)
- Adhoc Billing (C1-FABL)
- Freeze and Complete Adhoc Bills (C1-FCADH)
- Pending Bill Segments Deletion (C1-BSEGD)
- Pending Bill Deletion (C1-PNBD)
- Assign Sequential Bill Numbers (ASSGNSBN)

1.1 Billing (BILLING)

The **Billing (BILLING)** batch is used to generate bills based on the bill cycle defined for the account. Based on the batch business date, the system identifies the bill cycles for which the bills must be generated. Once the bill cycles are identified, the system identifies a set of accounts on which these bill cycles are defined. Then, based on a set of parameters specified while executing this batch, the bills are generated for accounts that meet the criteria and the status of the bills is set to **Pending**.

Then, based on the batch business date, the system identifies the recurring and non-recurring billable charges for which bill segments must be generated. Once the billable charges are identified, the bill segments are generated in the pending bills. In addition, the post-processing bill segments, if any, are generated in the pending bills.

Note:

If the **Billable Charges Date** option type of the **C1_FINTRANOP** feature configuration is set to **E**, the system considers the following:

>> Recurring billable charges whose start date is earlier than the cut-off date and end date is later than the cut-off date

>> Non-recurring billable charges whose end date is earlier than or equal to the cut-off date

If the **Freeze and Complete** check box is selected for the bill cycle for which the bill is generated, the status of the bill segments is set to **Frozen** and the status of the bill is set to **Complete**. But, if the **Freeze and Complete** check box is not selected for the bill cycle for which the bill is generated, the status of the bill segments is set to **Freezable** and the status of the bill remains as **Pending**.

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. You can specify the following parameters while executing this batch:

Parameter Name	Mandatory (Yes or No)	Description
Batch Business Date	No	<p>Used to identify the following:</p> <ul style="list-style-type: none"> • Bill cycles within which the specified date falls • Billable charges for which bill segments must be generated <p>Note: All billable charges whose start or end date is earlier than or equal to the specified date are considered during bill segment generation. If you do not specify any date, the batch business date is set to the current date.</p>
Thread Count	No	<p>Used to specify the number of threads you want to spawn in parallel.</p> <p>Note: By default, the thread count is set to 0.</p>
Division	No	Used when you want to generate bills for accounts belonging to a particular division.
Bill Cycle	No	<p>Used when you want to generate bills for accounts having a particular bill cycle.</p> <p>Note: You can specify maximum twenty comma separated values in this parameter.</p>
Account ID	No	Used when you want to generate bills for a particular account.
Override Maximum Number of Errors	No	Used to override the maximum number of errors after which the batch must be terminated.
Thread Pool Name	No	Used to specify the thread pool on which you want to execute the batch.

Note: If the **Billing (BILLING)** batch fails or aborts due to some reason, you can restart the batch over and over again with the same set of parameters.

Post Execution Check/Clean Up:

On successful completion of this batch, the bills are generated for accounts that meet the criteria.

1.2 Pending Bill Generation (C1-PNDBL)

The **Pending Bill Generation (C1-PNDBL)** batch is used to generate pending trial or actual bills for accounts that meet the criteria.

Based on the batch business date, the system identifies the bill cycles for which the pending trial or actual bills must be generated. Once the bill cycles are identified, the system identifies a set of accounts on which these bill cycles are defined. Then, based on a set of parameters specified while executing this batch, the pending trial or actual bills are generated for accounts that meet the criteria.

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. You can specify the following parameters while executing this batch:

Parameter Name	Mandatory (Yes or No)	Description
Batch Business Date	No	Used to identify the bill cycles within which the specified date falls. Note: If you do not specify any date, the batch business date is set to the current date.
Thread Count	No	Used to specify the number of threads you want to spawn in parallel. Note: By default, the thread count is set to 0.
Division	Yes	Used when you want to generate pending trial or actual bills for accounts belonging to a particular division.
Bill Cycle	No	Used when you want to generate pending trial or actual bills for accounts having a particular bill cycle. Note: You can specify maximum twenty comma separated values in this parameter.

Parameter Name	Mandatory (Yes or No)	Description
Delete Existing Bill	No	<p>Used to indicate whether you want to delete existing actual bills which are in the Pending status. The valid value is:</p> <ul style="list-style-type: none"> Y <div style="border: 1px solid black; padding: 5px;"> <p>Note:</p> <p>If you do not want to delete the existing pending bill, you must leave this parameter blank. In such case, new pending bill is not generated, and the existing pending bill (available for the account and bill cycle combination) is used for further processing.</p> <p>The system does not delete the existing bills which are in the Complete status.</p> <p>If the existing pending bills have any bill segments in the Freezable or Error status, the bills are deleted. But, if the existing pending bills have any bill segments in the Frozen or Cancelled status, the bills are not deleted.</p> <p>This parameter is not supported when you are generating trial bills.</p> </div>
Bill Generation Type	No	<p>Used to indicate whether you want to generate pending trial or actual bills. If you want to generate pending trial bills, the valid value is Trial. And, if you want to generate pending actual bills, you must leave this parameter blank.</p>
Description for Trial Billing Batch Run	No	<p>Used to specify the description for the batch run. This description is stamped on each trial bill which is generated in the batch run. This helps to identify the trial bills which are generated in a particular batch run.</p> <div style="border: 1px solid black; padding: 5px;"> <p>Note:</p> <p>This parameter is used only when you are generating pending trial bills.</p> <p>If you do not specify the description, by default, the description is set to "Trial Invoicing Run On YYYY-MM-DD-HH.MM.SS".</p> </div>

Parameter Name	Mandatory (Yes or No)	Description
Process All or Selected Accounts	No	Used to indicate whether you want to generate pending trial bills for all accounts (that meet the criteria) or only for those accounts (that meet the criteria and) where the Trial Bill check box is selected. If you want to generate pending trial bills for only those accounts where the Trial Bill check box is selected, set the value of this parameter to Y . If you want to generate pending trial bills for all accounts irrespective of whether the Trial Bill check box is selected or not, you must leave this parameter blank. Note: This parameter is used either when you are creating trial bills or when you are creating actual bills using trial bills.
Account ID	No	Used when you want to generate pending trial or actual bills for a particular account.
Override Maximum Number of Errors	No	Used to override the maximum number of errors after which the batch must be terminated.
Thread Pool Name	No	Used to specify the thread pool on which you want to execute the batch.

Note: If the **Pending Bill Generation (C1-PNDBL)** batch fails or aborts due to some reason, you can restart the batch over and over again with the same set of parameters.

Post Execution Check/Clean Up:

On successful completion of this batch, the pending trial or actual bills are generated for the accounts that meet the criteria.

1.3 Bill Segment Generation (C1-BLGEN)

The **Bill Segment Generation (C1-BLGEN)** batch is used to generate bill segments in the pending trial or actual bills. It is also used to copy trial bills' bill segments to the actual bills while creating actual bills using trial bills.

Based on the batch business date, the system identifies the recurring and non-recurring billable charges for which bill segments must be generated. Once the billable charges are identified, the bill segments are generated in the pending trial or actual bills.

Note:

If the **Billable Charges Date** option type of the **C1_FINTRANOP** feature configuration is set to **E**, the system considers the following:

>> Recurring billable charges whose start date is earlier than the cut-off date and end date is later than the cut-off date

>> Non-recurring billable charges whose end date is earlier than or equal to the cut-off date

If the **Freeze and Complete** check box is selected for the bill cycle for which the pending actual bill is generated, the status of the bill segments is set to **Frozen**. But, if the **Freeze and Complete** check box is not selected for the bill cycle for which the pending actual bill is generated, the status of the bill segments is set to **Freezable**. However, in case of pending trial bills, irrespective of whether the **Freeze and Complete** check box is selected or not for the bill cycle, the status of the bill segments is set to **Freezable**.

While creating actual bills using trial bills, the system copies trial bills' bill segments which are in the **Freezable** status to the actual bills. If you want to freeze the bill segments in such actual bills, you need to execute the **Bill Segment Generation (C1-BLGEN)** batch once again without specifying the trial billing batch run number.

Note:

There might be situations when you have received certain billing information after the bill and bill segments are generated. In such scenarios, you must first execute the **Pending Bill Generation (C1-PNDBL)** batch to fetch the newly created billable charges and then execute the **Bill Segment Generation (C1-BLGEN)** batch. While executing the **Pending Bill Generation (C1-PNDBL)** batch, you must leave the **Delete Existing Bill** parameter blank indicating that you want to use the existing pending bill for further processing.

In addition, there might be situations when you want to regenerate the bill segments before freezing them. In such case, you need to set the **Bill Segment Regeneration on Freeze** option type of the **C1_FINTRANOP** feature configuration to **Y**, and then execute the **Pending Bill Generation (C1-PNDBL)** and **Bill Segment Generation (C1-BLGEN)** batches once again. While executing the **Pending Bill Generation (C1-PNDBL)** batch, you must leave the **Delete Existing Bill** parameter blank indicating that you want to use the existing pending bill for further processing. If the **Freeze and Complete** check box is selected for the bill cycle and the **Bill Segment Regeneration on Freeze** option type is set to **Y**, then the bill segments are regenerated and the status of the bill segments is set to **Frozen**. Note that if the **Freeze and Complete** check box is not selected for the bill cycle, then the bill segments are always regenerated irrespective of whether the **Bill Segment Regeneration on Freeze** option type is set to **Y** or not.

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

Parameter Name	Mandatory (Yes or No)	Description
Batch Business Date	No	Used to identify the billable charges for which bill segments must be generated. Note: All billable charges whose start or end date is earlier than or equal to the specified date are considered during bill segment generation. If you do not specify any date, the batch business date is set to the current date.
Thread Count	No	Used to specify the number of threads you want to spawn in parallel. Note: By default, the thread count is set to 0.
Division	Yes	Used when you want to generate bill segments for accounts belonging to a particular division.
Bill Cycle	No	Used when you want to generate bill segments for accounts having a particular bill cycle. Note: You can specify maximum twenty comma separated values in this parameter.
Billing Processing Sequence	No	Used to specify the billing processing sequence of the contract's contract type whose bill segments you want to generate. This parameter is useful when you want to generate bill segments in a particular order of importance. Note: You can specify comma separated values in this parameter. For example, 10, 20, 30. In such case, the bill segments of contracts whose contract types' billing processing sequence is 10, 20, or 30 will be generated simultaneously.

Parameter Name	Mandatory (Yes or No)	Description
Bill Generation Type	No	Used to indicate whether you want to generate bill segments for pending trial or actual bills. If you want to generate bill segments for pending trial bills, the valid value is Trial . And, if you want to generate bill segments for pending actual bills or if you want to copy trial bills' bill segments while creating actual bills using trial bills, you must leave this parameter blank.
Trial Billing Batch Run Number	No	Used to indicate the trial billing batch run number whose trial bills' bill segments you want to copy to the actual bills. Note: This parameter is used only when you are creating actual bills using trial bills.
Process All or Selected Accounts	No	Used to indicate whether you want to generate bill segments for all accounts (that meet the criteria) or only for those accounts (that meet the criteria and) where the Trial Bill check box is selected. If you want to generate bill segments for only those accounts where the Trial Bill check box is selected, set the value of this parameter to Y . If you want to generate bill segments for all accounts irrespective of whether the Trial Bill check box is selected or not, you must leave this parameter blank. Note: This parameter is used either when you are creating trial bills or when you are creating actual bills using trial bills. If you set the value of this parameter to Y while creating actual bills using trial bills, the system will copy trial bills' bill segments of only those accounts (that meet the criteria and) where the Trial Bill check box is selected. And, if you leave this parameter blank, the system will copy trial bills' bill segments of all accounts (that meet the criteria) irrespective of whether the Trial Bill check box is selected or not. If the bill segments do not exist for an account while creating actual bills using trial bills, the system generates the bill segments for the account.
Account ID	No	Used when you want to generate bill segments for a particular account.

Parameter Name	Mandatory (Yes or No)	Description
Override Maximum Number of Errors	No	Used to override the maximum number of errors after which the batch must be terminated.
Thread Pool Name	No	Used to specify the thread pool on which you want to execute the batch.

Note:

You must specify same parameters in the **C1-PNDBL**, **C1-BLGEN** and **C1-BLPPR** batches. Otherwise, erroneous results will occur.

If the **Bill Segment Generation (C1-BLGEN)** batch fails or aborts due to some reason, you can restart the batch over and over again with the same set of parameters.

Post Execution Check/Clean Up:

On successful completion of this batch, the bill segments are generated in the pending trial or actual bills.

1.4 Bill Completion (C1-BLPPR)

The **Bill Completion (C1-BLPPR)** batch is used to generate post-processing bill segments in the pending trial or actual bills during the bill completion process. The adjustments, if any, created on the accounts are swept onto the pending trial or actual bills. In addition, the bill routing information and bill messages are stamped on the pending trial or actual bills. If the adjustment currency is different from the account's invoice currency, the transfer adjustment is created on the pending actual bill. However, in case of pending trial bill, the currency conversion for adjustments is done, but the transfer adjustments are not created. Finally, the status of the actual bill is set to **Complete**, whereas the status of the trial bill remains as **Pending**.

Based on the batch business date, the system identifies the bill cycles whose pending bills must be completed. Once the bill cycles are identified, the system identifies a set of accounts on which these bill cycles are defined. Then, based on a set of parameters specified while executing this batch, the pending bills are completed for accounts that meet the criteria.

If the **Freeze and Complete** check box is selected for the bill cycle for which the pending actual bill is generated, the status of the post-processing bill segments is set to **Frozen** and the status of the actual bill is changed to **Complete**. But, if the **Freeze and Complete** check box is not selected for the bill cycle for which the pending actual bill is generated, the status of the post-processing bill segments is set to **Freezable** and the status of the actual bill remains as **Pending**. However, in case of pending trial bills, irrespective of whether the **Freeze and Complete** check box is selected or not for the bill cycle, the status of the post-processing bill segments is set to **Freezable** and the status of the trial bill remains as **Pending**.

While creating actual bills using trial bills, the system copies trial bills' post-processing bill segments which are in the **Freezable** status to the actual bills. If you want to freeze the post-processing bill segments and complete such actual bills, you need to execute the **Bill Completion (C1-BLPPR)** batch once again without specifying the trial billing batch run number.

Note:

There might be situations when you have selected the **Freeze and Complete** check box for the bill cycle after all three batches are executed. In such scenarios, the system allows you to re-execute these batches. However, to enhance performance, you must first execute the **Bill Segment Generation (C1-BLGEN)** batch and then execute the **Bill Completion (C1-BLPPR)** batch. The status of the bill segments and bills is changed accordingly.

In addition, there might be situations when you want to regenerate the post-processing bill segments before freezing them. In such case, you need to set the **Bill Segment Regeneration on Freeze** option type of the **C1_FINTRANOP** feature configuration to **Y**, and then execute the **Bill Completion (C1-BLPPR)** batch once again. If the **Freeze and Complete** check box is selected for the bill cycle and the **Bill Segment Regeneration on Freeze** option type is set to **Y**, then the post-processing bill segments are regenerated, and the status of the bill segments is set to **Frozen**. Note that if the **Freeze and Complete** check box is not selected for the bill cycle, then the post-processing bill segments are always regenerated irrespective of whether the **Bill Segment Regeneration on Freeze** option type is set to **Y** or not.

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. You can specify the following parameters while executing this batch:

Parameter Name	Mandatory (Yes or No)	Description
Batch Business Date	No	Used to identify the bill cycles within which the specified date falls. Note: If you do not specify any date, the batch business date is set to the current date.
Thread Count	No	Used to specify the number of threads you want to spawn in parallel. Note: By default, the thread count is set to 0.
Division	Yes	Used when you want to initiate the bill completion process for accounts belonging to a particular division.
Bill Cycle	No	Used when you want to initiate the bill completion process for accounts having a particular bill cycle. Note: You can specify maximum twenty comma separated values in this parameter.

Parameter Name	Mandatory (Yes or No)	Description
Bill Generation Type	No	Used to indicate whether you want to initiate the bill completion process for pending trial or actual bills. If you want to initiate the bill completion process for pending trial bills, the valid value is Trial . And, if you want to initiate the bill completion process for pending actual bills or if you want to copy trial bills' post-processing bill segments while creating actual bills using trial bills, you must leave this parameter blank.
Trial Billing Batch Run Number	No	Used to indicate the trial billing batch run number whose trial bills' post-processing bill segments you want to copy to the actual bills. <div>Note: This parameter is used only when you are creating actual bills using trial bills.</div>

Parameter Name	Mandatory (Yes or No)	Description
Process All or Selected Accounts	No	<p>Used to indicate whether you want to initiate the bill completion process for all accounts (that meet the criteria) or only for those accounts (that meet the criteria and) where the Trial Bill check box is selected. If you want to initiate the bill completion process for those accounts where the Trial Bill check box is selected, set the value of this parameter to Y. If you want to initiate the bill completion process for all accounts irrespective of whether the Trial Bill check box is selected or not, you must leave this parameter blank.</p> <div> <p>Note:</p> <p>This parameter is used either when you are creating trial bills or when you are creating actual bills using trial bills.</p> <p>If you set the value of this parameter to Y while creating actual bills using trial bills, the system will copy trial bills' post-processing bill segments of only those accounts (that meet the criteria and) where the Trial Bill check box is selected. And, if you leave this parameter blank, the system will copy trial bills' post-processing bill segments of all accounts (that meet the criteria) irrespective of whether the Trial Bill check box is selected or not. If the post-processing bill segments do not exist for an account while creating actual bills using trial bills, the system generates the post-processing bill segments (if any) for the account.</p> </div>
Account ID	No	Used when you want to initiate the bill completion process for a particular account.
Override Maximum Number of Errors	No	Used to override the maximum number of errors after which the batch must be terminated.
Thread Pool Name	No	Used to specify the thread pool on which you want to execute the batch.

Note:

You must specify same parameters in the **C1-PNDBL**, **C1-BLGEN** and **C1-BLPPR** batches. Otherwise, erroneous results will occur.

If the **Bill Completion (C1-BLPPR)** batch fails or aborts due to some reason, you can restart the batch over and over again with the same set of parameters.

To improve the performance of the **Bill Completion (C1-BLPPR)** batch, you can store the post-processing price item pricing information. For more information, refer to the [Refresh Pricing \(C1-TXNRP\)](#) section.

Post Execution Check/Clean Up:

On successful completion of this batch, the post-processing bill segments (if any) are generated in the pending trial or actual bills. The status of the actual bill is changed to **Complete** and the status of the trial bill remains as **Pending**.

1.5 Construct Based - Pending Bill Generation (BILLOPEN)

The **Construct Based - Pending Bill Generation (BILLOPEN)** batch is used to generate pending trial or actual bills for accounts that meet the criteria.

Based on the batch business date, the system identifies the bill cycles for which the pending trial or actual bills must be generated. Once the bill cycles are identified, the system identifies a set of accounts on which these bill cycles are defined. These accounts may be usage or invoice accounts. Then, based on a set of parameters specified while executing this batch, the pending trial or actual bills are generated for accounts that meet the criteria.

In addition, this batch fetches all recurring and non-recurring billable charges of the accounts with the **Adhoc** flag set to **No** and whose start or end date is earlier than or equal to the batch business date. If a billable charge satisfies the rule defined on any active construct, the construct and invoice account details are stamped in the temporary area. And, if a billable charge does not satisfy the rule defined on any active construct, the invoice account details are stamped in the temporary area. Note that in such case, the account will be billed individually and not through any invoice account.

Note:

If the **Billable Charges Date** option type of the **C1_FINTRANOP** feature configuration is set to **E**, the system considers the following:

>> Recurring billable charges whose start date is earlier than the cut-off date and end date is later than the cut-off date

>> Non-recurring billable charges whose end date is earlier than or equal to the cut-off 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. You can specify the following parameters while executing this batch:

Parameter Name	Mandatory (Yes or No)	Description
Batch Business Date	No	Used to identify the bill cycles within which the specified date falls. Note: If you do not specify any date, the batch business date is set to the current date.

Parameter Name	Mandatory (Yes or No)	Description
Thread Count	No	Used to specify the number of threads you want to spawn in parallel. Note: By default, the thread count is set to 0.
Division	Yes	Used when you want to generate pending trial or actual bills for accounts belonging to a particular division.
Bill Cycle	No	Used when you want to generate pending trial or actual bills for accounts having a particular bill cycle. Note: You can specify maximum twenty comma separated values in this parameter.
Delete Existing Bill	No	Used to indicate whether you want to delete existing actual bills which are in the Pending status. The valid value is: <ul style="list-style-type: none"> Y Note: If you do not want to delete the existing pending bill, you must leave this parameter blank. In such case, new pending bill is not generated, and the existing pending bill (available for the account and bill cycle combination) is used for further processing. The system does not delete the existing bills which are in the Complete status. If the existing pending bills have any bill segments in the Freezable or Error status, the bills are deleted. But, if the existing pending bills have any bill segments in the Frozen or Cancelled status, the bills are not deleted. This parameter is not supported when you are generating trial bills.
Bill Generation Type	No	Used to indicate whether you want to generate pending trial or actual bills. If you want to generate pending trial bills, the valid value is Trial . And, if you want to generate pending actual bills, you must leave this parameter blank.

Parameter Name	Mandatory (Yes or No)	Description
Description for Trial Billing Batch Run	No	<p>Used to specify the description for the batch run. This description is stamped on each trial bill which is generated in the batch run. This helps to identify the trial bills which are generated in a particular batch run.</p> <div> <p>Note:</p> <p>This parameter is used only when you are generating pending trial bills.</p> <p>If you do not specify the description, by default, the description is set to “Trial Invoicing Run On YYYY-MM-DD-HH.MM.SS”.</p> </div>
Override Maximum Number of Errors	No	Used to override the maximum number of errors after which the batch must be terminated.
Thread Pool Name	No	Used to specify the thread pool on which you want to execute the batch.

Note: If the **Construct Based - Pending Bill Generation (BILLOPEN)** batch fails or aborts due to some reason, you can restart the batch over and over again with the same set of parameters.

Post Execution Check/Clean Up:

On successful completion of this batch, the pending trial or actual bills are generated for the accounts that meet the criteria.

1.6 Construct Based - Bill Segment Generation (BSGENREG)

The **Construct Based - Bill Segment Generation (BSGENREG)** batch is used to generate bill segments in the pending trial or actual bills. It is also used to copy trial bills' bill segments to the actual bills while creating actual bills using trial bills.

If the construct and invoice account details are stamped corresponding to a billable charge in the temporary area, the bill segment is generated on the pending trial or actual bill of the invoice account. However, if only the invoice account details are stamped corresponding to a billable charge in the temporary area, the bill segment is generated on the pending trial or actual bill of the usage account.

If the **Freeze and Complete** check box is selected for the bill cycle for which the pending actual bill is generated, the status of the bill segments is set to **Frozen**. But, if the **Freeze and Complete** check box is not selected for the bill cycle for which the pending actual bill is generated, the status of the bill segments is set to **Freezable**. However, in case of pending trial bills, irrespective of whether the **Freeze and Complete** check box is selected or not for the bill cycle, the status of the bill segments is set to **Freezable**.

While creating actual bills using trial bills, the system copies trial bills' bill segments which are in the **Freezable** status to the actual bills. If you want to freeze the bill segments in such actual bills, you need to execute the **Construct Based - Bill Segment Generation (BSGENREG)** batch once again without specifying the trial billing batch run number.

Note:

There might be situations when you have received certain billing information after the bill and bill segments are generated. In such scenarios, you must first execute the **Construct Based - Pending Bill Generation (BILLOPEN)** batch to fetch the newly created billable charges and then execute the **Construct Based - Bill Segment Generation (BSGENREG)** batch. While executing the **Construct Based - Pending Bill Generation (BILLOPEN)** batch, you must leave the **Delete Existing Bill** parameter blank indicating that you want to use the existing pending bill for further processing.

In addition, there might be situations when you want to regenerate the bill segments before freezing them. In such case, you need to set the **Bill Segment Regeneration on Freeze** option type of the **C1_FINTRANOP** feature configuration to **Y**, and then execute the **Construct Based - Pending Bill Generation (BILLOPEN)** and **Construct Based - Bill Segment Generation (BSGENREG)** batches once again. While executing the **Construct Based - Pending Bill Generation (BILLOPEN)** batch, you must leave the **Delete Existing Bill** parameter blank indicating that you want to use the existing pending bill for further processing. If the **Freeze and Complete** check box is selected for the bill cycle and the **Bill Segment Regeneration on Freeze** option type is set to **Y**, then the bill segments are regenerated and the status of the bill segments is set to **Frozen**. Note that if the **Freeze and Complete** check box is not selected for the bill cycle, then the bill segments are always regenerated irrespective of whether the **Bill Segment Regeneration on Freeze** option type is set to **Y** or not.

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

Parameter Name	Mandatory (Yes or No)	Description
Batch Business Date	No	Used to identify the billable charges for which bill segments must be generated. Note: All billable charges whose start or end date is earlier than or equal to the specified date are considered during bill segment generation. If you do not specify any date, the batch business date is set to the current date.
Thread Count	No	Used to specify the number of threads you want to spawn in parallel. Note: By default, the thread count is set to 0.
Division	No	Used when you want to generate bill segments for accounts belonging to a particular division.

Parameter Name	Mandatory (Yes or No)	Description
Bill Cycle	No	Used when you want to generate bill segments for accounts having a particular bill cycle. Note: You can specify maximum twenty comma separated values in this parameter.
Bill Generation Type	No	Used to indicate whether you want to generate bill segments for pending trial or actual bills. If you want to generate bill segments for pending trial bills, the valid value is Trial . And, if you want to generate bill segments for pending actual bills or if you want to copy trial bills' bill segments while creating actual bills using trial bills, you must leave this parameter blank.
Trial Billing Batch Run Number	No	Used to indicate the trial billing batch run number whose trial bills' bill segments you want to copy to the actual bills. Note: This parameter is used only when you are creating actual bills using trial bills.
Override Maximum Number of Errors	No	Used to override the maximum number of errors after which the batch must be terminated.
Thread Pool Name	No	Used to specify the thread pool on which you want to execute the batch.

Note:

You must specify same parameters in the **BILLOPEN**, **BSGENREG** and **POSTPROC** batches. Otherwise, erroneous results will occur.

If the **Construct Based - Bill Segment Generation (BSGENREG)** batch fails or aborts due to some reason, you can restart the batch over and over again with the same set of parameters.

Post Execution Check/Clean Up:

On successful completion of this batch, the bill segments are generated in the pending trial or actual bills.

1.7 Construct Based - Bill Completion (POSTPROC)

The **Construct Based - Bill Completion (POSTPROC)** batch is used to generate post-processing bill segments in the pending trial or actual bills during the bill completion process. The post-processing bill segments, if any, are generated on the invoice or usage account depending on the type of price assignment. If an adjustment created on the account satisfies the rule defined on any active construct, the adjustment is swept onto the pending trial or actual bill of the invoice account. However, if an adjustment created on the account does not satisfy the rule defined on any active construct, the adjustment is swept onto the pending trial or actual bill of the usage account.

In addition, the bill routing information and bill messages are stamped on the pending trial or actual bills. If the adjustment currency is different from the account's invoice currency, the transfer adjustment is created on the pending actual bill. However, in case of pending trial bill, the currency conversion for adjustments is done, but the transfer adjustments are not created. Finally, the status of the actual bill is set to **Complete**, whereas the status of the trial bill remains as **Pending**.

Based on the batch business date, the system identifies the bill cycles whose pending bills must be completed. Once the bill cycles are identified, the system identifies a set of accounts on which these bill cycles are defined. Then, based on a set of parameters specified while executing this batch, the pending bills are completed for accounts that meet the criteria.

If the **Freeze and Complete** check box is selected for the bill cycle for which the pending actual bill is generated, the status of the post-processing bill segments is set to **Frozen** and the status of the actual bill is changed to **Complete**. But, if the **Freeze and Complete** check box is not selected for the bill cycle for which the pending actual bill is generated, the status of the post-processing bill segments is set to **Freezable** and the status of the actual bill remains as **Pending**. However, in case of pending trial bills, irrespective of whether the **Freeze and Complete** check box is selected or not for the bill cycle, the status of the post-processing bill segments is set to **Freezable** and the status of the trial bill remains as **Pending**.

While creating actual bills using trial bills, the system copies trial bills' post-processing bill segments which are in the **Freezable** status to the actual bills. If you want to freeze the post-processing bill segments and complete such actual bills, you need to execute the **Construct Based - Bill Completion (POSTPROC)** batch once again without specifying the trial billing batch run number.

Note:

There might be situations when you have selected the **Freeze and Complete** check box for the bill cycle after all three batches are executed. In such scenarios, the system allows you to re-execute these batches. However, to enhance performance, you must first execute the **Construct Based - Bill Segment Generation (BSGENREG)** batch and then execute the **Construct Based - Bill Completion (POSTPROC)** batch. The status of the bill segments and bills is changed accordingly.

In addition, there might be situations when you want to regenerate the post-processing bill segments before freezing them. In such case, you need to set the **Bill Segment Regeneration on Freeze** option type of the **C1_FINTRANOP** feature configuration to **Y**, and then execute the **Construct Based - Bill Completion (POSTPROC)** batch once again. If the **Freeze and Complete** check box is selected for the bill cycle and the **Bill Segment Regeneration on Freeze** option type is set to **Y**, then the post-processing bill segments are regenerated, and the status of the bill segments is set to **Frozen**. Note that if the **Freeze and Complete** check box is not selected for the bill cycle, then the post-processing bill segments are always regenerated irrespective of whether the **Bill Segment Regeneration on Freeze** option type is set to **Y** or not.

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. You can specify the following parameters while executing this batch:

Parameter Name	Mandatory (Yes or No)	Description
Batch Business Date	No	Used to identify the bill cycles within which the specified date falls. Note: If you do not specify any date, the batch business date is set to the current date.
Thread Count	No	Used to specify the number of threads you want to spawn in parallel. Note: By default, the thread count is set to 0.
Division	Yes	Used when you want to initiate the bill completion process for accounts belonging to a particular division.
Bill Cycle	No	Used when you want to initiate the bill completion process for accounts having a particular bill cycle. Note: You can specify maximum twenty comma separated values in this parameter.
Bill Generation Type	No	Used to indicate whether you want to initiate the bill completion process for pending trial or actual bills. If you want to initiate the bill completion process for pending trial bills, the valid value is Trial . And, if you want to initiate the bill completion process for pending actual bills or if you want to copy trial bills' post-processing bill segments while creating actual bills using trial bills, you must leave this parameter blank.
Trial Billing Batch Run Number	No	Used to indicate the trial billing batch run number whose trial bills' post-processing bill segments you want to copy to the actual bills. Note: This parameter is used only when you are creating actual bills using trial bills.
Override Maximum Number of Errors	No	Used to override the maximum number of errors after which the batch must be terminated.
Thread Pool Name	No	Used to specify the thread pool on which you want to execute the batch.

Note:

You must specify same parameters in the BILLOPEN, BSGENREG and POSTPROC batches. Otherwise, erroneous results will occur.

If the Construct Based - Bill Completion (POSTPROC) batch fails or aborts due to some reason, you can restart the batch over and over again with the same set of parameters.

Post Execution Check/Clean Up:

On successful completion of this batch, the post-processing bill segments (if any) are generated in the pending trial or actual bills. The status of the actual bill is changed to **Complete** and the status of the trial bill remains as **Pending**.

1.8 Refresh Pricing (C1-TXNRP)

You can store the regular and post-processing price item pricing information in the following tables and thereby improve the performance of the **Price Item Pricing Verification (C1-TXNVP)** and **Bill Completion (C1-BLPPR)** batches:

- CI_PRC_AGRD
- CI_PRC_PL
- CI_PRC_INH_PL

If you set the **Materialized View Used** option type of the **C1_FM** feature configuration to **true**, the system will store the regular and post-processing price item pricing information in the above mentioned tables. But, if you set the **Materialized View Used** option type of the **C1_FM** feature configuration to **false**, the system will not store the regular and post-processing price item pricing information in the above mentioned tables.

This mechanism to store pricing information in the above mentioned tables helps to quickly search for regular or post-processing price item pricing information while executing the **Price Item Pricing Verification (C1-TXNVP)** or **Bill Completion (C1-BLPPR)** batch, respectively. The system stores all price items pricing information irrespective of whether it is effective or not. If the price assignment ID is not stamped on any billable charge, the system refers to the price assignment search algorithm which searches for the pricing in the original tables and not in the above mentioned tables.

If there are any pricing changes, you will have to update these tables before you execute the **Price Item Pricing Verification (C1-TXNVP)** or **Bill Completion (C1-BLPPR)** batch. The **Refresh Pricing (C1-TXNRP)** batch is used to update the regular and post-processing price item pricing information in the above mentioned tables.

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

Parameter Name	Mandatory (Yes or No)	Description
Thread Count	No	Used to specify the number of threads you want to spawn in parallel. <div>Note: By default, the thread count is set to 0.</div>

Parameter Name	Mandatory (Yes or No)	Description
Chunk Size	Yes	Used to specify the number of persons whose regular and post-processing price item pricing information you want to update in each work unit.
Division	No	Used when you want to update the price item pricing information of accounts belonging to a particular division.
Thread Pool Name	No	Used to specify the thread pool on which you want to execute the batch.

Note: If the **Refresh Pricing (C1-TXNRP)** batch fails or aborts due to some reason, you can restart the batch over and over again with the same set of parameters.

Post Execution Check/Clean Up:

On successful completion of this batch, the regular and post-processing price item pricing information is updated in the CI_PRC_AGRD, CI_PRC_PL, and CI_PRC_INH_PL tables. During this process, the existing data is first removed from these tables, and then the latest information is added into these tables.

1.9 Upload and Validate Usage Data File (C1-ODFU)

The **Upload and Validate Usage Data File (C1-ODFU)** batch is used to upload and validate a usage data file. It is also used for revalidating a usage data file which is already uploaded in the system.

While uploading or revalidating a usage data file, a batch job is created using this batch automatically when the usage data file size exceeds the file size limit defined in the **On Demand Billing (C1-ODBFC)** feature configuration.

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

Parameter Name	Mandatory (Yes or No)	Description
File Group	Yes	Used to specify the group in which you want to upload the usage data file.
File Name	Yes	Used to specify the name of the usage data file that you want to upload or revalidate.
Thread Pool Name	No	Used to specify the thread pool on which you want to execute the batch. Note: If a batch job is created automatically, the value defined in the Thread Pool Name option type of the On Demand Billing (C1-ODBFC) feature configuration is used.
File Type	Yes	Used to specify the type of usage data file.

Parameter Name	Mandatory (Yes or No)	Description
Billing Method	Yes	<p>Used to specify the type of billing you want to use for the usage data file. The valid values are:</p> <ul style="list-style-type: none"> • ADB • RLB <p>Note: ADB stands for adhoc billing and RLB stands for regular billing.</p>
Access Group	No	Used to specify the access group of the user who is uploading or revalidating the usage data file.
Bill After Date	Yes (Conditional)	<p>Used to specify the date after which the bills must be generated for the usage data file.</p> <p>Note: This parameter is required only when the billing method is ADB.</p>
Default Usage Start Date	Yes	Used to specify the date from when the price item was used. The default usage start date is used when the start date is not specified for a record in the usage data file.
Default Usage End Date	Yes	Used to specify the date till when the price item was used. The default usage end date is used when the end date is not specified for a record in the usage data file.
File Path	Yes (Conditional)	<p>Used to specify the path of the usage data file that you want to upload.</p> <p>Note: This parameter is not required when you are revalidating a usage data file.</p>
Revalidate	Yes	<p>Used to indicate whether you want to revalidate the usage data file. The valid values are:</p> <ul style="list-style-type: none"> • Y – Used when you want to revalidate a usage data file. • N – Used when you want to upload and validate a usage data file.
Chunk Size	Yes	<p>Used to specify the number of usage records you want to validate or revalidate in each work unit.</p> <p>Note: If a batch job is created automatically, the value defined in the Validation Chunk Size option type of the On Demand Billing (C1-ODBFC) feature configuration is used.</p>

Note:

If a batch job is created automatically, the values for these parameters are automatically specified in the batch job depending on whether you are uploading or revalidating a usage data file.

If the **Upload and Validate Usage Data File (C1-ODFU)** batch fails or aborts due to some reason, you cannot restart the batch. Instead, you need to upload the usage data file once again in the system.

Post Execution Check/Clean Up:

On successful completion of this batch, records are either uploaded and validated or revalidated in the system. If a usage record is successfully validated, the status of the usage record is changed to **Valid**. And, if a usage record could not pass through the validation process, the status of the usage record is changed to **Invalid**.

In addition, a To Do entry is created using the **C1-ODFU** To Do type. You can view this To Do entry only when you are assigned a To Do role which is mapped to the **C1-ODFU** To Do type. This To Do indicates whether the batch was successfully executed or not. If the batch was successfully executed, it indicates the following:

- Number of records that were uploaded in the system (appears only on uploading a usage data file)
- Number of records which were successfully validated
- Number of records which could not pass through the validation process
- Number of records which failed to validate due to some technical reasons

1.10 Billable Charge Creation (C1-ODBCH)

The **Billable Charge Creation (C1-ODBCH)** batch is used to create a billable charge for a valid usage record. While submitting or approving usage records, a batch job is created using this batch automatically when the total number of valid records in a usage data file exceeds the transaction limit defined in the **On Demand Billing (C1-ODBFC)** feature configuration.

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

Parameter Name	Mandatory (Yes or No)	Description
File Name	Yes	Used to specify the name of the usage data file for which you want to create the billable charges.
Thread Pool Name	No	Used to specify the thread pool on which you want to execute the batch. Note: If a batch job is created automatically, the value defined in the Thread Pool Name option type of the On Demand Billing (C1-ODBFC) feature configuration is used.

Parameter Name	Mandatory (Yes or No)	Description
Chunk Size	Yes	Used to specify the number of valid records for which you want to create billable charges in each work unit. Note: If a batch job is created automatically, the value defined in the Billable Charge Chunk Size option type of the On Demand Billing (C1-ODBFC) feature configuration is used.
File Group Header ID	No	Used to specify the header ID of the file group for which you want to create the billable charges.

Note:

If a batch job is created automatically, the values for these parameters are automatically specified in the batch job.

If the **Billable Charge Creation (C1-ODBCH)** batch fails or aborts due to some reason, you cannot restart the batch. Instead, you need to submit or approve the usage records once again in the system.

Post Execution Check/Clean Up:

On successful completion of this batch, a billable charge is created for each valid record. The status of the usage record is changed to **Processed**.

In addition, a To Do entry is created using the **C1-ODBCH** To Do type. You can view this To Do entry only when you are assigned a To Do role which is mapped to the **C1-ODBCH** To Do type. This To Do indicates whether the batch was successfully executed or not. If the batch was successfully executed, it indicates the following:

- Number of records for which billable charges were created
- Number of records for which billable charges could not be created successfully

1.11 Adhoc Billing (C1-FABL)

The **Adhoc Billing (C1-FABL)** batch is used to generate an adhoc bill for a file group and account combination or for an account when the bill after date is earlier than or equal to the batch business date. Both, automatically generated and manually created billable charges (with the **Adhoc Bill** flag set to **Y**) are considered during adhoc billing.

While generating adhoc bills through the **On Demand Billing** screen, a batch job is created using this batch automatically when the total number of billable charges for a file group exceeds the online bill creation limit defined in the **On Demand Billing (C1-ODBFC)** feature configuration.

Once the billable charges are identified, bill segments are generated in the adhoc bill. The status of bill segments is set to **Freezable** and the status of adhoc bill is set to **Pending**. In the following scenarios, the system behaves in the following manner:

- If an adhoc bill with the **Pending** status already exists for an account or for a file group and account combination, then the existing bill is deleted and regenerated.

- If an adhoc bill with the **Completed** status already exists for an account or for a file group and account combination, then a new bill is generated (but the existing bill is not deleted).
- If a completed bill of an account or of a file group and account combination is cancelled (in other words, bill segments in an adhoc bill are cancelled), a new bill is generated (but the existing bill is not deleted).

This batch is a multi-threaded batch. The chunks for multi-threading are created randomly and there is no specific logic defined for chunking in the system. You can specify the following parameters while executing this batch:

Parameter Name	Mandatory (Yes or No)	Description
Batch Business Date	No	<p>Used to identify the billable charges for which bill segments must be generated. If the file group is specified, the billable charges that meet the following criteria are considered during adhoc billing:</p> <ul style="list-style-type: none"> • Billable charges are created for accounts whose usage data is uploaded in the specified file group. • The Adhoc Bill flag of the billable charge is set to Y. • Billable charges whose bill after date is earlier than or equal to the specified date. <div> Note: If you do not specify any date, the batch business date is set to the current date. This parameter is used to identify billable charges only when the cut-off date is not specified. </div>
File Group	Yes	<p>Used to indicate the file group for which you want to generate the adhoc bills.</p> <div> Note: If a batch job is created automatically, the value for this parameter is automatically specified in the batch job. </div>
Thread Pool Name	No	<p>Used to specify the thread pool on which you want to execute the batch.</p> <div> Note: If a batch job is created automatically, the value defined in the Thread Pool Name option type of the On Demand Billing (C1-ODBFC) feature configuration is used. </div>
Division	No	Used when you want to generate adhoc bills for accounts belonging to a particular division.

Parameter Name	Mandatory (Yes or No)	Description
Account ID	No	Used when you want to generate adhoc bills for a particular account.
Lead Days	No	Used to specify the number of lead days that you want to consider while identifying the billable charges. If you specify the lead days, the system considers all adhoc billable charges whose bill after date is earlier than or equal to batch business date + lead days.
Cutoff Date	No	Used to identify the billable charges for which bill segments must be generated. If the file group is specified, the billable charges that meet the following criteria are considered during adhoc billing: <ul style="list-style-type: none"> • Billable charges are created for accounts whose usage data is uploaded in the specified file group. • The Adhoc Bill flag of the billable charge is set to Y. • Billable charges whose bill after date is earlier than or equal to the specified date.
Accounting Date	No	Used to specify the accounting date. <div> Note: The accounting date cannot be earlier than the cutoff date. </div>

Note: If the **Adhoc Billing (C1-FABL)** batch fails or aborts due to some reason, you cannot restart the batch. Instead, you need to generate adhoc bills for the closed file group once again in the system.

Post Execution Check/Clean Up:

On successful completion of this batch, an adhoc bill is generated for an account or for a file group and account combination when the bill after date is earlier than or equal to the batch business date or batch business date + lead days.

In addition, a To Do entry is created using the **C1-FABL** To Do type. You can view this To Do entry only when you are assigned a To Do role which is mapped to the **C1-FABL** To Do type. This To Do indicates whether the batch was successfully executed or not. If the batch was successfully executed, it indicates the following:

- Number of billable charges which were processed
- Number of billable charges for which the bill after date is a future date
- Number of adhoc bills generated for the usage data file
- Number of billable charges which could not be processed successfully

1.12 Freeze and Complete Adhoc Bills (C1-FCADH)

The **Freeze and Complete Adhoc Bills (C1-FCADH)** batch is used to freeze and complete all adhoc bills of a file group which are in the **Pending** status. While executing this batch, the system does not freeze and complete pending adhoc bills which have bill segments in the **Error** status.

While freezing and completing adhoc bills through the **On Demand Billing** screen, a batch job is created using this batch automatically.

This batch is a multi-threaded batch. The chunks for multi-threading are created randomly and there is no specific logic defined for chunking in the system. You can specify the following parameters while executing this batch:

Parameter Name	Mandatory (Yes or No)	Description
File Group	Yes	Used to indicate the file group whose adhoc bills you want to freeze and complete. Note: If a batch job is created automatically, the value for this parameter is automatically specified in the batch job.
Accounting Date	No	Used to specify the accounting date. This parameter is useful when you want to override the accounting date specified while creating adhoc bills.
Bill Date	No	Used to specify the bill date that you want to display on the adhoc bills. Note: If you do not specify any date, the bill date is set to the current date.
Thread Pool Name	No	Used to specify the thread pool on which you want to execute the batch. Note: If a batch job is created automatically, the value defined in the Thread Pool Name option type of the On Demand Billing (C1-ODBFC) feature configuration is used.

Note: If the **Freeze and Complete Adhoc Bills (C1-FCADH)** batch fails or aborts due to some reason, you cannot restart the batch. Instead, you need to freeze and complete adhoc bills of the closed file group once again in the system.

Post Execution Check/Clean Up:

On successful completion of this batch, adhoc bills of the file group are completed. The status of the adhoc bills is changed to **Complete**.

In addition, a To Do entry is created using the **C1-FCADH** To Do type. You can view this To Do entry only when you are assigned a To Do role which is mapped to the **C1-FCADH** To Do type. This To Do indicates whether the batch was successfully executed or not. If the batch was successfully executed, it indicates the following:

- Number of bills which were picked up for completion
- Number of bills which were completed
- Number of bills which could not be completed successfully

1.13 Pending Bill Segments Deletion (C1-BSEGD)

The **Pending Bill Segments Deletion (C1-BSEGD)** batch is used to delete bill segments which are in the **Freezable** or **Error** status. It does not delete bill segments which are in the **Pending Cancel**, **Frozen**, or **Cancel** status.

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

Parameter Name	Mandatory (Yes or No)	Description
Thread Count	No	Used to specify the number of threads you want to spawn in parallel. Note: By default, the thread count is set to 0.
Account ID	No	Used to indicate the account whose bill segments you want to delete.
Division	No	Used when you want to delete bill segments of accounts belonging to a particular division.
Bill Cycle	No	Used when you want to delete bill segments of accounts having a particular bill cycle.
Request Type	No	Used to indicate whether you want to delete bill segments during the disaggregation process. The valid values are: <ul style="list-style-type: none"> • BILLING – Used when you want to delete bill segments of pending bills of accounts that meet the criteria. • DISAGG – Used when you want to delete bill segments of pending bills of accounts that meet the criteria and for which the disaggregation request is created. Note: If you do not specify any value, by default, the parameter value is set to BILLING .

Parameter Name	Mandatory (Yes or No)	Description
Chunk Size	Yes	Used to specify the number of transactions you want to execute in each work unit.
Maximum Batch Count	Yes	Used to specify the maximum number of transactions after which the data must be committed in the database.
Thread Pool Name	No	Used to specify the thread pool on which you want to execute the batch.

Note:

At present, the batch business date is not used (or considered) while executing the **Pending Bill Segments Deletion (C1-BSEGD)** batch.

If the **Pending Bill Segments Deletion (C1-BSEGD)** batch fails or aborts due to some reason, you can restart the batch over and over again with the same set of parameters.

Post Execution Check/Clean Up:

On successful completion of this batch, the bill segments in the **Freezable** or **Error** status and their corresponding financial transactions are deleted. The corresponding bill ID is added in the CI_DELETE_BILL_DETAIL table for further processing.

1.14 Pending Bill Deletion (C1-PNBD)

The **Pending Bill Deletion (C1-PNBD)** batch checks whether the pending bills listed in the CI_DELETE_BILL_DETAIL table have any bill segments in the **Pending Cancel**, **Frozen**, or **Cancel** status. If there are bill segments in the **Pending Cancel**, **Frozen**, or **Cancel** status, the pending bill is not deleted. However, if there are no bill segments in the **Pending Cancel**, **Frozen**, or **Cancel** status, the pending bill is deleted. This batch deletes regular pending bills and not adhoc pending bills.

Note: If the pending bills have bill segments, you must first execute the **Pending Bill Segments Deletion (C1-BSEGD)** batch and then execute the **Pending Bill Deletion (C1-PNBD)** batch. While executing these batches in the specified order, ensure that you specify the same parameters in both these batches.

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

Parameter Name	Mandatory (Yes or No)	Description
Thread Count	No	Used to specify the number of threads you want to spawn in parallel. <div>Note: By default, the thread count is set to 0.</div>
Account ID	No	Used to indicate the account whose pending bills you want to delete.

Parameter Name	Mandatory (Yes or No)	Description
Division	No	Used when you want to delete pending bills of accounts belonging to a particular division.
Bill Cycle Code	No	Used when you want to delete pending bills of accounts having a particular bill cycle.
Request Type	No	<p>Used to indicate whether you want to delete pending bills during the disaggregation process. The valid values are:</p> <ul style="list-style-type: none"> • BILLING – Used when you want to delete pending bills of accounts that meet the criteria. • DISAGG – Used when you want to delete pending bills of accounts that meet the criteria and for which the disaggregation request is created. <div style="border: 1px solid black; padding: 5px; margin-top: 10px;"> <p>Note: If you do not specify any value, by default, the parameter value is set to BILLING.</p> </div>
Chunk Size	Yes	Used to specify the number of transactions you want to execute in each work unit.
Maximum Batch Count	Yes	Used to specify the maximum number of transactions after which the data must be committed in the database.
Thread Pool Name	No	Used to specify the thread pool on which you want to execute the batch.

Note:

At present, the batch business date is not used (or considered) while executing the **Pending Bill Deletion (C1-PNBD)** batch.

If the **Pending Bill Deletion (C1-PNBD)** batch fails or aborts due to some reason, you can restart the batch over and over again with the same set of parameters.

Post Execution Check/Clean Up:

On successful completion of this batch, the BILL_ACT_CD column corresponding to a pending bill in the CI_DELETE_BILL_DETAIL table is set to either 10 or 20. 10 indicate that the pending bill contains no bill segments in the **Pending Cancel, Frozen, or Cancel** status, and therefore can be deleted. However, 20 indicate that the pending bill contains bill segments in the **Pending Cancel, Frozen, or Cancel** status, and therefore cannot be deleted. If the BILL_ACT_CD column corresponding to a pending bill is set to 10, the system deletes the pending bill.

1.15 Assign Sequential Bill Numbers (ASSGNSBN)

The **Assign Sequential Bill Numbers (ASSGNSBN)** batch is used to generate alternate sequential bill numbers. However, you can only generate alternate sequential bill numbers for bills that are complete and for which the alternate sequential bill numbers are not yet generated.

You need to configure this batch such that it is executed at regular intervals. Note that you need to schedule this batch only when you want to generate alternate sequential bill numbers for completed bills (i.e. when the **Use Sequential Bill Numbers** check box is selected in the **Billing** tab of the **Installation Options** screen).

This batch invokes the following algorithm:

If the alternate sequential bill numbers must be....	Then the batch...
Unique throughout the system	Invokes an algorithm created using the C1_ALTBLLSYS or C1_ALTBLSYS1 algorithm type which is attached to the Sequence Generation Algorithm spot in the Billing tab of the Installation Options screen.
Unique within the division	Invokes an algorithm created using the C1_ALTBLLDIV or C1_ALTBLLDIV1 algorithm type which is attached to the Sequential Bill Number Generation algorithm spot of the division.

This batch is a single-threaded batch. You can specify the following parameters while executing this batch:

Parameter Name	Mandatory (Yes or No)	Description
Override Maximum Number of Errors	No	Used to override the maximum number of errors after which the batch must be terminated.
Division	No	Used when you want to generate alternate sequential bill numbers for completed bills which belong to a particular division.

Note: If the **Assign Sequential Bill Numbers (ASSGNSBN)** batch fails or aborts due to some reason, you can restart the batch over and over again with the same set of parameters.

Post Execution Check/Clean Up:

On successful completion of this batch, an alternate sequential bill number is generated for the completed bills and stamped corresponding to the bill in the **CI_BILL** table. A To Do is generated for a bill when error occurs while generating alternate sequential bill number for the bill.

2. Payment

This section provides detail information about the following batches:

- Payment Upload (PUPL)
- Automatic Payment Creation (APAYCRET)
- Activate Automatic Payments (ACTVTAPY)
- Extract Automatic Payments (APAYACH)
- Freeze or Cancel Automatic Payments (APAYRA)
- To Do Creation for Automatic Payment Exception Records (APAYUPTD)
- Create Tender Controls for Automatic Payments (BALAPY)
- Payment Request Periodic Monitor (C1-PAYRQ)

2.1 Payment Upload (PUPL)

The **Payment Upload (PUPL)** batch is used to create the deposit control, tender controls, payment events, tenders, payments, and payment segments using payment records in the staging area. In addition, the financial transactions are created when the payments are frozen. The system creates these objects only when the accounting date (i.e. date received in case of the **Payment Upload** process) of the payment record is earlier than or equal to the batch business date.

For the **Payment Upload** process, you need to configure this batch such that it is executed at regular intervals. It 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. You can specify the following parameters while executing this batch:

Parameter Name	Mandatory (Yes or No)	Description
Override Maximum Number of Errors	No	Used to override the maximum number of errors after which the batch must be terminated.
Thread Pool Name	No	Used to specify the thread pool on which you want to execute the batch.
Staging Header ID	No	Used to indicate the payment data file for which you want to create payments.

Note:

When the batch is automatically invoked at predefined intervals, the system does not specify any value in the above mentioned parameters.

If the **Payment Upload (PUPL)** batch fails or aborts due to some reason, you can restart the batch over and over again with the same set of parameters.

Post Execution Check/Clean Up:

On successful completion of this batch, the records are added in the following tables:

- CI_DEP_CTL
- CI_TNDR_CTL
- CI_PAY_EVENT
- CI_PAY_TNDR
- CI_PAY
- CI_PAY_CHAR (if the characteristics exist for the payment)
- CI_FT
- CI_FT_GL

In addition, the status of the corresponding deposit control, tender control, payment tender, and payment staging records is set to **Complete**. If an error occurs while creating a payment event and/or payment, an exception is added in the CI_PAY_EVT_EXCP and CI_PAY_EXCP tables, respectively. If any error occurs while creating payment event for a payment record, the status of the deposit control, tender control, payment tender, and payment staging records is changed to **Error**. In addition, the status of the payment record in the pre-staging area is changed to **Error in Staging**.

Traditionally, the **Payment Upload (PUPL)** batch considered staging records which are in the **Error** or **Pending** status. If a staging record is in the **Error** status, the system used to reset the status to **Pending** and then further processed the staging record. This system behavior still exists for staging records which are not uploaded through the **Payment Upload** process. If the staging records are uploaded through the **Payment Upload** process, you need to manually correct the staging records and change the status of deposit control, tender control, payment tender, and payment staging records to **Pending**. The corrected staging records will then be processed when the **Payment Upload (PUPL)** batch is invoked at subsequent interval.

2.2 Automatic Payment Creation (APAYCRET)

When the automatic payment facility is used for an account, the system calculates the automatic payment amount and extracts date during the bill completion and stamps these details against the bill in the CI_BILL_ACH table. The **Automatic Payment Creation (APAYCRET)** batch is used to create automatic payments and payment events for such bills. It also creates a clearing record for each automatic payment in the CI_APAY_CLR_STG table. The payment freeze date in the clearing record is set to NULL. The extract date is calculated and stamped on the clearing record. This extract date is later on used to activate the clearing record.

You must execute the **Automatic Payment Creation (APAYCRET)** batch in the specified sequence when the **Autopay Creation Option** field in the installation options is set to the following:

Autopay Creation Option	Batch Execution Sequence
Create On Extract Date	<ol style="list-style-type: none"> 1. Automatic Payment Creation (APAYCRET) 2. Activate Automatic Payments (ACTVTAPY) 3. Extract Automatic Payments (APAYACH) 4. Distribute and Freeze Automatic Payments (APAYDSFR) 5. Create Tender Controls for Automatic Payments (BALAPY)
Freeze Payment on Notification	<ol style="list-style-type: none"> 1. Automatic Payment Creation (APAYCRET) 2. Activate Automatic Payments (ACTVTAPY) 3. Extract Automatic Payments (APAYACH) 4. Freeze or Cancel Automatic Payments (APAYRA) 5. To Do Creation for Automatic Payment Exception Records (APAYUPTD) 6. Create Tender Controls for Automatic Payments (BALAPY)

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

Parameter Name	Mandatory (Yes or No)	Description
Batch Business Date	No	Used to identify the bills whose extract date is earlier than or equal to the specified date. The batch then creates automatic payments for such bills. Note: If you do not specify any date, the batch business date is set to the current date.
Override Maximum Number of Errors	No	Used to override the maximum number of errors after which the batch must be terminated.
Thread Pool Name	No	Used to specify the thread pool on which you want to execute the batch.
Division	No	Used when you want to create automatic payments for accounts belonging to a particular division.

Note: If the **Automatic Payment Creation (APAYCRET)** batch fails or aborts due to some reason, you can restart the batch over and over again with the same set of parameters.

Post Execution Check/Clean Up:

On successful completion of this batch, the records are added in the following tables:

- CI_APAY_CLR_STG

- CI_PAY
- CI_PAY_EVENT
- CI_PAY_TNDR

While adding the clearing record in the CI_APAY_CLR_STG table, the APAY_DIST_FRZ_DT column is set to NULL. In addition, the extract date is added in the SCHED_EXTRACT_DT column of the CI_APAY_CLR_STG table.

2.3 Activate Automatic Payments (ACTVTAPY)

Unless you activate a clearing record, you cannot extract the clearing record in a flat file. The **Activate Automatic Payments (ACTVTAPY)** batch is used to activate a clearing record by stamping the **APAYACH** batch control and incremental batch run number on the clearing record. Only those clearing records whose extract date is earlier than or equal to the batch business date are activated through this batch.

You must execute the **Activate Automatic Payments (ACTVTAPY)** batch in the specified sequence when the **Autopay Creation Option** field in the installation options is set to the following:

Autopay Creation Option	Batch Execution Sequence
Create At Bill Completion	<ol style="list-style-type: none"> 1. Activate Automatic Payments (ACTVTAPY) 2. Extract Automatic Payments (APAYACH) 3. Create Tender Controls for Automatic Payments (BALAPY)
Create On Extract Date	<ol style="list-style-type: none"> 1. Automatic Payment Creation (APAYCRET) 2. Activate Automatic Payments (ACTVTAPY) 3. Extract Automatic Payments (APAYACH) 4. Distribute and Freeze Automatic Payments (APAYDSFR) 5. Create Tender Controls for Automatic Payments (BALAPY)
Freeze Payment on Notification	<ol style="list-style-type: none"> 1. Automatic Payment Creation (APAYCRET) 2. Activate Automatic Payments (ACTVTAPY) 3. Extract Automatic Payments (APAYACH) 4. Freeze or Cancel Automatic Payments (APAYRA) 5. To Do Creation for Automatic Payment Exception Records (APAYUPTD) 6. Create Tender Controls for Automatic Payments (BALAPY)

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

Parameter Name	Mandatory (Yes or No)	Description
Batch Business Date	No	Used to identify the clearing records whose extract date is earlier than or equal to the specified date. Note: If you do not specify any date, the batch business date is set to the current date.
Override Maximum Number of Errors	No	Used to override the maximum number of errors after which the batch must be terminated.
Thread Pool Name	No	Used to specify the thread pool on which you want to execute the batch.
Division	No	Used when you want to activate clearing records of accounts belonging to a particular division.

Note: If the **Activate Automatic Payments (ACTVTAPY)** batch fails or aborts due to some reason, you can restart the batch over and over again with the same set of parameters.

Post Execution Check/Clean Up:

On successful completion of this batch, the **APAYACH** batch control is added in the **BATCH_CD** column and the next batch run number is added in the **BATCH_NBR** column corresponding to the clearing record in the **CI_APAY_CLR_STG** table.

2.4 Extract Automatic Payments (APAYACH)

The **Extract Automatic Payments (APAYACH)** batch is used to extract the clearing records with the latest batch run number in a flat file. This flat file is then sent to the auto clearing house for review.

You must execute the **Extract Automatic Payments (APAYACH)** batch in the specified sequence when the **Autopay Creation Option** field in the installation options is set to the following:

Autopay Creation Option	Batch Execution Sequence
Create At Bill Completion	<ol style="list-style-type: none"> 1. Activate Automatic Payments (ACTVTAPY) 2. Extract Automatic Payments (APAYACH) 3. Create Tender Controls for Automatic Payments (BALAPY)
Create On Extract Date	<ol style="list-style-type: none"> 1. Automatic Payment Creation (APAYCRET) 2. Activate Automatic Payments (ACTVTAPY) 3. Extract Automatic Payments (APAYACH) 4. Distribute and Freeze Automatic Payments (APAYDSFR) 5. Create Tender Controls for Automatic Payments (BALAPY)

Autopay Creation Option	Batch Execution Sequence
Freeze Payment on Notification	<ol style="list-style-type: none"> 1. Automatic Payment Creation (APAYCRET) 2. Activate Automatic Payments (ACTVTAPY) 3. Extract Automatic Payments (APAYACH) 4. Freeze or Cancel Automatic Payments (APAYRA) 5. To Do Creation for Automatic Payment Exception Records (APAYUPTD) 6. Create Tender Controls for Automatic Payments (BALAPY)

This batch is a single-threaded batch. You can specify the following parameters while executing this batch:

Parameter Name	Mandatory (Yes or No)	Description
Output Directory	No	Used to specify the path where you want to save the flat file. Note: If you do not specify the path, the flat file is, by default, saved in the <code>sploutput</code> directory.
Output File Name	No	Used to specify the name which you want to use for the flat file. Note: If you do not specify the name, the file is, by default, named as ACHMMDDYYHHSS. Here, MMDDYY indicates the date and HHSS indicates the time when the file is extracted.
Override Maximum Number of Errors	No	Used to override the maximum number of errors after which the batch must be terminated.
Thread Pool Name	No	Used to specify the thread pool on which you want to execute the batch.
Division	No	Used when you want to extract clearing records of accounts belonging to a particular division.

Note: If the **Extract Automatic Payments (APAYACH)** batch fails or aborts due to some reason, you can restart the batch over and over again with the same set of parameters.

Post Execution Check/Clean Up:

On successful completion of this batch, the flat file is created at the specified location. If you have not specified the path and name of the output file, by default, the file is saved with the ACHMMDDYYHHSS name in the `sploutput` directory.

2.5 Freeze or Cancel Automatic Payments (APAYRA)

The **Freeze or Cancel Automatic Payments (APAYRA)** batch is used to freeze or cancel the automatic payments based on the reason code. It checks whether the reason code (i.e. upload reason) and sub reason code (i.e. payment cancellation reason or NOC reason) specified in the clearing staging record exist in the system. A clearing staging record which is in the **Pending** status and for which payment freeze date is specified in the clearing record is considered during the batch execution.

If the reason code and sub reason code exist in the system, the system executes the algorithms attached to the reason code in the specified sequence. If the type of reason code is **Cancel**, the attached algorithm cancels the automatic payment and payment event. If the type of reason code is **Success**, the attached algorithm freezes the automatic payment on the payment freeze date. However, if the type of reason code is **Success** and the sub reason code is specified along with the reason code, the attached algorithm freezes the automatic payment on the payment freeze date, stores sub reason codes (i.e. NOC reasons) as payment tender characteristics and then creates a To Do entry to notify user about these NOC reasons. In addition, the status of the clearing staging record is changed to **Complete**.

If the reason code and/or sub reason code (i.e. upload reason, payment cancellation reason, or NOC reason) specified in the clearing staging record does not exist in the system, the status of the clearing staging record is changed to **Error**. In addition, an exception is logged in the CI_APAY_STGUP_EXC table.

You must execute the **Freeze or Cancel Automatic Payments (APAYRA)** batch in the specified sequence when the **Autopay Creation Option** field in the installation options is set to the following:

Autopay Creation Option	Batch Execution Sequence
Freeze Payment on Notification	<ol style="list-style-type: none"> 1. Automatic Payment Creation (APAYCRET) 2. Activate Automatic Payments (ACTVTAPY) 3. Extract Automatic Payments (APAYACH) 4. Freeze or Cancel Automatic Payments (APAYRA) 5. To Do Creation for Automatic Payment Exception Records (APAYUPTD) 6. Create Tender Controls for Automatic Payments (BALAPY)

This batch is a single-threaded batch. You can specify the following parameters while executing this batch:

Parameter Name	Mandatory (Yes or No)	Description
Override Maximum Number of Errors	No	Used to override the maximum number of errors after which the batch must be terminated.
Thread Pool Name	No	Used to specify the thread pool on which you want to execute the batch.
Division	No	Used when you want to freeze or cancel automatic payments of accounts belonging to a particular division.

Parameter Name	Mandatory (Yes or No)	Description
Validate Payment Freeze Date (Y or N)	No	<p>Used to indicate whether you want to validate the payment freeze date before freezing the automatic payments. The valid values are:</p> <ul style="list-style-type: none"> Y N <div style="border: 1px solid black; padding: 5px; margin-top: 10px;"> <p>Note:</p> <p>If you set this parameter to Y, the system considers only those automatic payment clearing records whose payment freeze date is earlier than or equal to the batch business date. However, if you set this parameter to N, the system considers automatic payment clearing records irrespective of whether the payment freeze date is set to a past date, current date, or future date.</p> <p>If you do not specify any value, by default, the parameter value is set to N.</p> </div>

Post Execution Check/Clean Up:

On successful completion of this batch, the status of the automatic payment clearing staging record is updated in the **APAY_UP_STATUS_FLG** column of the **CI_APAY_STAGE_UP** table. The status of automatic payment is updated in the **PAY_STATUS_FLG** column of the **CI_PAY** table. The **APAY_DIST_FRZ_DT** column of the corresponding automatic payment clearing record in the **CI_APAY_CLR_STG** table is set to **NULL**.

If the upload reason, payment cancelation reason, or NOC reason does not exist in the system, an exception is logged in the **CI_APAY_STGUP_EXC** table.

2.6 To Do Creation for Automatic Payment Exception Records (APAYUPTD)

The **To Do Creation for Automatic Payment Exception Records (APAYUPTD)** batch is used to create a To Do using the **APAYUPTD** To Do type when an exception is logged for an automatic payment clearing staging record in the **CI_APAY_STGUP_EXC** table. The system creates one To Do entry for all exceptions that have occurred for an automatic payment clearing staging record.

Only those exceptions where review date does not exist in the **CI_APAY_STGUP_EXC** table are considered during the batch execution. You must execute the **To Do Creation for Automatic Payment Exception Records (APAYUPTD)** batch in the specified sequence when the **Autopay Creation Option** field in the installation options is set to the following:

Autopay Creation Option	Batch Execution Sequence
Freeze Payment on Notification	<ol style="list-style-type: none"> 1. Automatic Payment Creation (APAYCRET) 2. Activate Automatic Payments (ACTVTAPY) 3. Extract Automatic Payments (APAYACH) 4. Freeze or Cancel Automatic Payments (APAYRA) 5. To Do Creation for Automatic Payment Exception Records (APAYUPTD) 6. Create Tender Controls for Automatic Payments (BALAPY)

This batch is a single-threaded batch. You can specify the following parameters while executing this batch:

Parameter Name	Mandatory (Yes or No)	Description
Override Maximum Number of Errors	No	Used to override the maximum number of errors after which the batch must be terminated.
Thread Pool Name	No	Used to specify the thread pool on which you want to execute the batch.

Note: If the **To Do Creation for Automatic Payment Exception Records (APAYUPTD)** batch fails or aborts due to some reason, you can restart the batch over and over again with the same set of parameters.

Post Execution Check/Clean Up:

On successful completion of this batch, the records are added in the **CI_TD_ENTRY** table. Also, the date when the To Do is generated is added in the **REVIEW_DT** column of the **CI_APAY_STGUP_EXC** table.

2.7 Distribute and Freeze Automatic Payments (APAYDSFR)

The **Distribute and Freeze Automatic Payments (APAYDSFR)** batch is used to distribute and freeze automatic payment records. It considers only those automatic payment clearing staging records from the **CI_APAY_CLR_STG** table whose distribution and freeze date is earlier than or equal to batch business date.

You must execute the **Distribute and Freeze Automatic Payments (APAYDSFR)** batch in the specified sequence when the **Autopay Creation Option** field in the installation options is set to the following:

Autopay Creation Option	Batch Execution Sequence
Create On Extract Date	<ol style="list-style-type: none"> 1. Automatic Payment Creation (APAYCRET) 2. Activate Automatic Payments (ACTVTAPY) 3. Extract Automatic Payments (APAYACH) 4. Distribute and Freeze Automatic Payments (APAYDSFR) 5. Create Tender Controls for Automatic Payments (BALAPY)

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

Parameter Name	Mandatory (Yes or No)	Description
Batch Business Date	No	<p>Used to identify the clearing staging records whose distribution date is earlier than or equal to the specified date.</p> <div> Note: If you do not specify any date, the batch business date is set to the current date. </div>
Override Maximum Number of Errors	No	Used to override the maximum number of errors after which the batch must be terminated.
Thread Pool Name	No	Used to specify the thread pool on which you want to execute the batch.
Division	No	Used when you want to distribute and freeze automatic payments of accounts belonging to a particular division.

Note: If the **Distribute and Freeze Automatic Payments (APAYDSFR)** batch fails or aborts due to some reason, you can restart the batch over and over again with the same set of parameters.

Post Execution Check/Clean Up:

On successful completion of this batch, the status of automatic payment is updated in the **PAY_STATUS_FLG** column of the **CI_PAY** table.

2.8 Create Tender Controls for Automatic Payments (BALAPY)

The **Create Tender Controls for Automatic Payments (BALAPY)** batch is used to create tender and deposit controls for each automatic payment which is frozen, but not yet linked to any tender control. One tender control is created for the **APAYACH** batch control and batch run number combination. The payment tenders of these automatic payments are then linked to the tender control. This batch also balances the open tender control records and changes the status of the tender and deposit controls, accordingly.

You must execute the **Create Tender Controls for Automatic Payments (BALAPY)** batch in the specified sequence when the **Autopay Creation Option** field in the installation options is set to the following:

Autopay Creation Option	Batch Execution Sequence
Create At Bill Completion	<ol style="list-style-type: none"> 1. Activate Automatic Payments (ACTVTAPY) 2. Extract Automatic Payments (APAYACH) 3. Create Tender Controls for Automatic Payments (BALAPY)
Create On Extract Date	<ol style="list-style-type: none"> 1. Automatic Payment Creation (APAYCRET) 2. Activate Automatic Payments (ACTVTAPY) 3. Extract Automatic Payments (APAYACH) 4. Distribute and Freeze Automatic Payments (APAYDSFR) 5. Create Tender Controls for Automatic Payments (BALAPY)
Freeze Payment on Notification	<ol style="list-style-type: none"> 1. Automatic Payment Creation (APAYCRET) 2. Activate Automatic Payments (ACTVTAPY) 3. Extract Automatic Payments (APAYACH) 4. Freeze or Cancel Automatic Payments (APAYRA) 5. To Do Creation for Automatic Payment Exception Records (APAYUPTD) 6. Create Tender Controls for Automatic Payments (BALAPY)

This batch is a single-threaded batch. You can specify the following parameters while executing this batch:

Parameter Name	Mandatory (Yes or No)	Description
Override Maximum Number of Errors	No	Used to override the maximum number of errors after which the batch must be terminated.
Thread Pool Name	No	Used to specify the thread pool on which you want to execute the batch.
Division	No	Used when you want to create tender controls for automatic payments of accounts belonging to a particular division.

Note: If the **Create Tender Controls for Automatic Payments (BALAPY)** batch fails or aborts due to some reason, you can restart the batch over and over again with the same set of parameters.

Post Execution Check/Clean Up:

On successful completion of this batch, the records are added in the following tables:

- CI_DEP_CTL
- CI_TNDR_CTL

The status is updated in the **DEP_CTL_STATUS_FLG** and **TNDR_CTL_ST_FLG** columns of the **CI_DEP_CTL** and **CI_TNDR_CTL** tables, respectively. In addition, the tender control ID is stamped in the **TNDR_CTL_ID** column of the **CI_PAY_TNDR** table.

2.9 Payment Request Periodic Monitor (C1-PAYRQ)

The **Payment Request Periodic Monitor (C1-PAYRQ)** batch is used to monitor or check whether there are any payment requests in the **Deferred Distribution** status. If there is a payment request in the **Deferred Distribution** status, the batch changes the status of the payment request to **Distributed**. Then, the algorithm attached to the **Distributed** status is triggered which distributes the tender or transfer amount.

Note: On distributing the tender amount, the payment event, payments, payment segments, and payment tender are created. The payments are created in the **Freezable** status. If any error occurs while creating a payment, the payment is created in the **Error** status. If all payments of a payment event are in the **Freezable** status, the status of the payment event is set to **Balanced**. However, if any payment of a payment event is in the **Error** status, the status of the payment event is set to **Unbalanced**. On distributing the payment amount (during transfer), the new payments are created in the **Freezable** status. If any error occurs while creating a new payment, the payment is created in the **Error** status. The status of the payment event is set to **Unbalanced**. The old payments in the payment event that you want to transfer remains in the **Frozen** or **Error** status.

This batch is a single-threaded batch. You can specify the following parameters while executing this batch:

Parameter Name	Mandatory (Yes or No)	Description
Maintenance Object	Yes	Used to indicate that you want to monitor payment requests created using the business objects of a particular maintenance object. Note: By default, the parameter value is set to C1-PAYREQ.
Restrict By Batch Code	No	Used when you want to monitor the payment requests whose current status is linked to the batch code. The valid value is true .
Restrict By Payment Request Type	No	Used when you want to monitor the payment requests which are created using a particular payment request type.

Parameter Name	Mandatory (Yes or No)	Description
Restrict By Business Object	No	Used when you want to monitor the payment requests which are created using a particular business object.
Restrict By Status Code	No	Used when you want to monitor the payment requests 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.
Thread Pool Name	No	Used to specify the thread pool on which you want to execute the batch.

Note: If the **Payment Request Periodic Monitor (C1-PAYRQ)** batch fails or aborts due to some reason, you can restart the batch over and over again with the same set of parameters.

Post Execution Check/Clean Up:

On successful completion of this batch, the status of the payment request is changed from **Deferred Distribution** to **Distributed**.

2.10 Upload Lockbox Payment and Remittance Advices (C1-PUPSG)

The **Upload Lockbox Payment and Remittance Advices (C1-PUPSG)** batch is used to read the 820 EDI lockbox file received from the bank containing the payment and remittance details for accounts, and upload the payment data in the payment upload staging tables.

It uses the source system customer number or taxpayer identification number specified in the payment or remittance record to identify the customer for whom the payment is made. If the customer is found in ORMB, it checks whether the identified customer has an account. If the identified customer has one account, the batch executes the algorithm created using the **Pay Tender Staging Account Distribution - Pay Oldest Bill First (C1-CCPYTSADS)** algorithm type and attached to the **Payment Tender Staging Distribution** system event of the customer class to which the account belongs.

Note:

This algorithm does the following:

>> If the **Check Binder Payment** parameter is set to **Y**, the algorithm checks whether the payment is the first payment for the account and the tender used for the payment is not automatic payment. If so, the payment is applied against the contract which is created using the contract type specified in the payment distribution algorithm of the Binder Payment match type.

>> If the **Check Promise To Pay** parameter is set to **Y**, the algorithm checks whether there is an active promise to pay for the account. If so, the payment is applied against the contract which is created using the contract type specified in the payment distribution algorithm of the On Account match type.

>> If the **Check Payment Agreement** parameter is set to **Y**, the algorithm checks whether there is an active payment agreement request for the account. If so, the payment is applied against the contract which is created using the contract type specified in the payment distribution algorithm of the On Account match type.

However, if the **Check Binder Payment**, **Check Promise To Pay**, and **Check Payment Agreement** parameters are set to **N**, the payment is applied to the account's unpaid bills in the order of the due date (i.e. oldest bill first). If the payment amount is greater than the account's billed balance plus overpayment threshold amount, the entire payment amount is applied against the excess credit contract of the account.

However, if the payment amount is greater than the account's billed balance, but less than account's billed balance plus overpayment threshold amount, the payment is first applied against the unpaid bills of the account and the overpayment amount is then applied against the excess credit contract of the account.

However, if the identified customer has multiple accounts, the batch identifies the account using either of the following (in the specified order):

1. **Invoice ID** – It is used to find the account for which the invoice is created.
2. **MICR** - It is used find the payment where the MICR is stamped as a characteristic. Once a payment is identified, the batch finds the account for which the payment is created. In this way, the batch derives the account to which the payment must be applied. However, the MICR is used to derive the account only when the **Search Account Using MICR (Y or N)** parameter in the batch is set to **Y**.

Once the account is identified, the batch executes the algorithm created using the **Pay Tender Staging Account Distribution - Pay Oldest Bill First (C1-CCPYTSADS)** algorithm type and attached to the **Payment Tender Staging Distribution** system event of the customer class to which the account belongs. However, if there are no accounts for the identified customer, or account could not be found in ORMB, or the customer could not be found in ORMB, the payment is applied against the suspense contract defined on the tender source which is associated with the external source (lockbox) ID.

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

Parameter Name	Mandatory (Yes or No)	Description
File Directory	Yes	Used to specify the path from where you want to upload the EDI 820 files.
File Name Prefix	Yes	Used when you want to upload EDI 820 files with a particular prefix from the specified path. Note: By default, the parameter value is set to usb_brms_lbx1_* .

Parameter Name	Mandatory (Yes or No)	Description
File Line Separator	Yes	Used to indicate the character specified in the 820 EDI files to represent the end of line. Note: By default, the parameter value is set to ~.
Record Field Separator	Yes	Used to indicate the character specified in the 820 EDI files to represent the end of data record. Note: By default, the parameter value is set to *.
Search Account Using MICR (Y or N)	No	Used to indicate whether you want to search the account to which the payment must be applied using the MICR code. The valid values are: <ul style="list-style-type: none"> • Y • N Note: By default, the parameter value is set to Y.
Customer Class	No	Used when you want to apply payments for accounts which belong to a particular customer class. Note: You can specify comma separated values for this parameter.
Division	No	Used when you want to apply payments for accounts which belong to a particular division.
General Suspense Match Type	Yes	Used to specify the match type using which you want to apply the payment against the suspense contract defined on the tender source. Note: By default, the parameter value is set to SUSPENSE .
On Account Match Type	Yes	Used to specify the match type using which you want to apply the payment against the contract of the account. Note: By default, the parameter value is set to SUSPENSE .

Parameter Name	Mandatory (Yes or No)	Description
Source System Customer Number Characteristic Type	Yes	<p>Used to specify the characteristic type which you want to use to store the source system customer number. This type of characteristic is defined on the payment upload staging record when the account to which the payment is applied is derived using the source system customer number.</p> <p>Note: By default, the parameter value is set to C1SRCSID. You can change the characteristic type, if required. However, you must specify a characteristic type where the characteristic entity is set to Payment.</p>
Remittance Invoice ID Characteristic Type	Yes	<p>Used to specify the characteristic type which you want to use to store the invoice ID. This type of characteristic is defined on the payment upload staging record when the account to which the payment is applied is derived using the invoice ID.</p> <p>Note: By default, the parameter value is set to C1RMIVID. You can change the characteristic type, if required. However, you must specify a characteristic type where the characteristic entity is set to Payment.</p>
Remittance Payment Amount Characteristic Type	Yes	<p>Used to specify the characteristic type which you want to use to store the payment amount. This type of characteristic is defined on the payment upload staging record when the account to which the payment is applied is derived using the invoice ID.</p> <p>Note: By default, the parameter value is set to C1RMPYMT. You can change the characteristic type, if required. However, you must specify a characteristic type where the characteristic entity is set to Payment.</p>

Parameter Name	Mandatory (Yes or No)	Description
Payment Review Reason Characteristic Type	Yes	<p>Used to specify the characteristic type which you want to use to store the payment reason. This type of characteristic is defined on the payment upload staging record when the payment is applied either against the suspense contract defined on the tender source or against the contract of the account.</p> <p>Note: By default, the parameter value is set to C1PYRVWR. You can change the characteristic type, if required. However, you must specify a characteristic type where the characteristic entity is set to Payment.</p>
Payment Review Reason for Multiple Accounts	Yes	<p>Used to specify the characteristic value that must be specified for the C1PYRVWR characteristic type when the customer has multiple accounts.</p> <p>Note: By default, the parameter value is set to MULTCUAC.</p>
Taxpayer Identification Number Characteristic Type	Yes	<p>Used to specify the characteristic type which you want to use to store the taxpayer identification number. This type of characteristic is defined on the payment upload staging record when the account to which the payment is applied is derived using the taxpayer identification number.</p> <p>Note: By default, the parameter value is set to C1TXPYID. You can change the characteristic type, if required. However, you must specify a characteristic type where the characteristic entity is set to Payment.</p>
Excess Credit Contract Type	Yes	<p>Used to specify the contract type using which the excess credit contract should be created if it does not exist for the account.</p> <p>Note: By default, the parameter value is set to EXCSCRED. You can change the parameter value, if required. However, you must specify a contract type which is already defined in the system.</p>

Parameter Name	Mandatory (Yes or No)	Description
Person Identifier Type	Yes	<p>Used to indicate the person identifier type. The system derives the customer or person using the person identifier type and person identifier combination.</p> <p>Note: By default, the parameter value is set to SSN. You can change the parameter value, if required. However, you must specify a person identifier type which is already defined in the system.</p>
Remittance Source System Customer Number Characteristic Type	Yes	<p>Used to specify the characteristic type which you want to use to store the remittance source system customer number. This type of characteristic is defined on the payment upload staging record when the account to which the payment is applied is derived using the remittance source system customer number.</p> <p>Note: By default, the parameter value is set to C1RMSCID. You can change the characteristic type, if required. However, you must specify a characteristic type where the characteristic entity is set to Payment.</p>
Remittance Taxpayer Identification Number Characteristic Type	Yes	<p>Used to specify the characteristic type which you want to use to store the remittance taxpayer identification number. This type of characteristic is defined on the payment upload staging record when the account to which the payment is applied is derived using the remittance taxpayer identification number.</p> <p>Note: By default, the parameter value is set to C1RMTXID. You can change the characteristic type, if required. However, you must specify a characteristic type where the characteristic entity is set to Payment.</p>

Parameter Name	Mandatory (Yes or No)	Description
Archive Files (Y or N)	Yes	Used to indicate whether you want to archive the files once the payment records are added in the payment upload staging tables. The valid values are: <ul style="list-style-type: none">• Y• N <div>Note: By default, the parameter value is set to N.</div>
Archive Directory	No	Used to specify the path where you want to move the files once the payment records are added in the payment upload staging tables.
Thread Pool Name	No	Used to specify the thread pool on which you want to execute the batch.

Note: If the **Upload Lockbox Payment and Remittance Advices (C1-PUPSG)** batch fails or aborts due to some reason, you can restart the batch over and over again with the same set of parameters.

Post Execution Check/Clean Up:

On successful completion of this batch, the records are added in the following tables:

- CI_DEP_CTL_ST
- CI_TNDR_CTL_ST
- CI_PAY_ST
- CI_PAY_ST_CHAR

3. Financial Transaction

This section provides detail information about the following batches:

- Foreign Exchange Loss Gain (C1-FXLG)
- Assign GL Account to Financial Transaction (C1-GLASN)
- Insert Records in CI_FTTEMP (GLASSGN1)
- Assign GL Account to Financial Transaction (GLASSGN2)
- GL Download Staging (GLS)
- GL Download Extract (GLDL)

3.1 Foreign Exchange Loss Gain (C1-FXLG)

The **Foreign Exchange Loss Gain (C1-FXLG)** batch is used to calculate foreign exchange gain or loss on financial transactions, such as payments and adjustments. This gain or loss occurs due to fluctuations in the exchange rate at different point in time. During the batch execution, the system considers the following types of financial transactions:

- Payments (i.e. Pay Segments)
- Credit adjustments which are created against a bill after the bill completion
- Write-off adjustments which are created against a bill
- Write Up adjustments which are automatically created when you match a payment against a bill using the Bill Weighted match type
- Write Down adjustments which are automatically created when you match a payment against a bill using the Bill Weighted match type

However, the system considers payment and adjustment FTs when:

- Division's Base Currency is different from the Payment or Adjustment Currency
- Payment or Adjustment Freeze Date is different from the Bill Completion Date
- Match Event ID exists in the financial transaction
- FXLG_CALC_STATUS column corresponding to the financial transaction in the CI_FT table is set to **N** or **NULL**

This batch is a multi-threaded batch. The multi-threading is based on financial transaction ID. You can specify the following parameters while executing this batch:

Parameter Name	Mandatory (Yes or No)	Description
Division	Yes	Used to calculate foreign exchange gain or loss for financial transactions, which are created for accounts, belonging to a particular division.

Parameter Name	Mandatory (Yes or No)	Description
Distribution Code for FXLG	Yes	Used to indicate the distribution code against which you want to book the foreign exchange gain or loss.
Distribution Code for Contract Balancing	No	Used to indicate the distribution code against which you want to balance the foreign exchange gain or loss. Note: If you do not specify any distribution code, by default, the distribution code associated with the FT's contract is used to balance the foreign exchange gain or loss.
Thread Pool Name	No	Used to specify the thread pool on which you want to execute the batch.

Note:

You must execute the **Foreign Exchange Loss Gain (C1-FXLG)** batch prior to the **Assign GL Account to Financial Transaction (C1-GLASN)** or **Insert Records in CI_FTTEMP (GLASSGN1)** batch.

If the **Foreign Exchange Loss Gain (C1-FXLG)** batch fails or aborts due to some reason, you can restart the batch over and over again with the same set of parameters.

Post Execution Check/Clean Up:

On successful completion of this batch, two entries are added corresponding to each financial transaction (for which foreign exchange gain or loss is calculated) in the CI_FT_GL and CI_FT_GL_EXT tables. One entry for foreign exchange gain or loss is booked against the distribution code for FXLG and another entry for foreign exchange gain or loss is booked against the distribution code which is used for contract balancing. The foreign exchange loss is recorded as positive entry and the foreign exchange gain is recorded as negative entry.

In the CI_FT_GL table, the foreign exchange gain or loss is recorded in the financial transaction currency. And, in the CI_FT_GL_EXT table, the foreign exchange gain or loss is recorded in the division's base currency. The FXLG_CALC_STATUS column corresponding to the financial transaction in the CI_FT table is set to I.

3.2 Assign GL Account to Financial Transaction (C1-GLASN)

The **Assign GL Account to Financial Transaction (C1-GLASN)** batch is used to assign GL account to trial and actual financial transactions. It allows you to validate the GL account before it is assigned to financial transactions. If the GL account is valid, the date when the GL account is validated is added in the GLA_VAL_DT column of the CI_TRL_FT_GL or CI_FT_GL table, respectively, depending on whether the GL account is assigned to a trial or actual financial transaction. However, the column is not updated when the GL account is invalid or when the GL account is not validated. This batch also facilitates you to assign GL account only to frozen or to both frozen and freezable financial transactions depending on the requirement.

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

Parameter Name	Mandatory (Yes or No)	Description
Process Only Frozen FTs	No	Used to indicate whether you want to assign GL account only to frozen financial transactions. If you set the value of this parameter to Y , GL account is assigned only to frozen financial transactions. However, if you want to assign GL account to both frozen and freezable financial transactions, you must leave this parameter blank.
Chunk Size	No	Used to specify the number of financial transactions to which you want to assign GL account in each work unit.
Thread Pool Name	No	Used to specify the thread pool on which you want to execute the batch.
Division	No	Used when you want to assign GL account to financial transactions, which are created for accounts, belonging to a particular division.
Bill Generation Type	No	Used to indicate whether you want to assign GL account to trial or actual financial transactions. The valid values are: <ul style="list-style-type: none"> • Trial • Regular <div>Note: If you do not specify any value, the bill generation type is set to Regular.</div>
Trial Billing Batch Run Number	No	Used to indicate the trial billing batch run number whose trial bills' financial transactions you want to consider for assigning GL account. <div>Note: This parameter is used only when you are assigning GL account to trial financial transactions.</div>
Validate GL Account	No	Used to indicate whether you want to validate the GL account before it is assigned to the financial transaction. If you set the value of this parameter to Y , the GL account is validated. However, if you do not want to validate the GL account, you must leave this parameter blank.

Parameter Name	Mandatory (Yes or No)	Description
Generate Conditional GL Entries Switch	No	Used to indicate whether you want to create additional FT GL entries for the financial transactions. The valid value is Y . If you set the value of this parameter to Y , the system will invoke the C1-GLCE algorithm to create additional FT GL entries. However, if you do not want to create additional FT GL entries for the financial transactions, you must leave this parameter blank.

Note: If the **Assign GL Account to Financial Transaction (C1-GLASN)** batch fails or aborts due to some reason, you can restart the batch over and over again with the same set of parameters.

Post Execution Check/Clean Up:

On successful completion of this batch, the following columns are updated in the CI_TRL_FT_GL or CI_FT_GL table depending on whether the GL account is assigned to a trial or actual financial transaction:

- GL_ACCT
- GLA_VAL_DT
- VALIDATE_SW

The GLA_VAL_DT column is updated when the GL account is valid and not when the GL account is invalid.

3.3 Insert Records in CI_FTTEMP (GLASSGN1)

The **Insert Records in CI_FTTEMP (GLASSGN1)** batch is used to search and add the frozen and/or freezable financial transactions (FTs), where the GL account is not yet assigned, in the CI_FTTEMP table. The data in this table is then used for creating chunks for multi-threading while executing the **Assign GL Account to Financial Transaction (GLASSGN2)** batch.

This batch is a single-threaded batch. You can specify the following parameters while executing this batch:

Parameter Name	Mandatory (Yes or No)	Description
Override Maximum Number of Errors	No	Used to override the maximum number of errors after which the batch must be terminated.

Parameter Name	Mandatory (Yes or No)	Description
Process Only Frozen FTs	No	Used to indicate whether you want to add only frozen financial transactions in the CI_FTTEMP table. If you set the value of this parameter to Y , only frozen financial transactions are added in the CI_FTTEMP table. However, if you want to add both frozen and freezable financial transactions, you must leave this parameter blank.
Division	No	Used when you want to add financial transactions, which are created for accounts, belonging to a particular division.
Bill Generation Type	No	Used to indicate whether you want to add trial or actual financial transactions in the CI_FTTEMP table. The valid values are: <ul style="list-style-type: none"> • Trial • Regular <div> Note: If you do not specify any value, the bill generation type is set to Regular. </div>
Trial Billing Batch Run Number	No	Used to indicate the trial billing batch run number whose trial bills' financial transactions you want to add in the CI_FTTEMP table. <div> Note: This parameter is used only when you are adding trial financial transactions in the CI_FTTEMP table. </div>

Note:

You must specify same parameters in the GLASSGN1 and GLASSGN2 batches. Otherwise, erroneous results might occur.

If the **Insert Records in CI_FTTEMP (GLASSGN1)** batch fails or aborts due to some reason, you can restart the batch over and over again with the same set of parameters.

Post Execution Check/Clean Up:

On successful completion of this batch, the financial transactions from CI_TRL_FT_GL or CI_FT_GL table are added in the CI_FTTEMP table.

3.4 Assign GL Account to Financial Transaction (GLASSGN2)

The **Assign GL Account to Financial Transaction (GLASSGN2)** batch is used to assign GL account to frozen and/or freezable financial transactions which are stored in the CI_FTTEMP table. It allows you to validate the GL account before it is assigned to financial transactions. If the GL account is valid, the date when the GL account is validated is added in the GLA_VAL_DT column of the CI_TRL_FT_GL or CI_FT_GL table, respectively, depending on whether the GL account is assigned to a trial or actual financial transaction. However, the column is not updated when the GL account is invalid or when the GL account is not validated.

If the GL account is not successfully assigned any financial transaction, an error is logged and To Do is created for the corresponding financial transaction.

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

Parameter Name	Mandatory (Yes or No)	Description
Override Maximum Number of Errors	No	Used to override the maximum number of errors after which the batch must be terminated.
Process Only Frozen FTs	No	Used to indicate whether you want to assign GL account only to frozen financial transactions. If you set the value of this parameter to Y , GL account is assigned only to frozen financial transactions. However, if you want to assign GL account to both frozen and freezable financial transactions, you must leave this parameter blank.
Division	No	Used when you want to assign GL account to financial transactions, which are created for accounts, belonging to a particular division.
Bill Generation Type	No	Used to indicate whether you want to assign GL account to trial or actual financial transactions. The valid values are: <ul style="list-style-type: none"> • Trial • Regular <div style="border: 1px solid black; padding: 5px; margin-top: 10px;"> Note: If you do not specify any value, the bill generation type is set to Regular. </div>

Parameter Name	Mandatory (Yes or No)	Description
Trial Billing Batch Run Number	No	Used to indicate the trial billing batch run number whose trial bills' financial transactions you want to consider for assigning GL account. Note: This parameter is used only when you are assigning GL account to trial financial transactions.
Validate GL Account	No	Used to indicate whether you want to validate the GL account before it is assigned to the financial transaction. If you set the value of this parameter to Y , the GL account is validated. However, if you do not want to validate the GL account, you must leave this parameter blank.
Generate Conditional GL Entries Switch	No	Used to indicate whether you want to create additional FT GL entries for the financial transactions. The valid value is Y . If you set the value of this parameter to Y , the system will invoke the C1-GLCE algorithm to create additional FT GL entries. However, if you do not want to create additional FT GL entries for the financial transactions, you must leave this parameter blank.

Note:

You must specify same parameters in the GLASSGN1 and GLASSGN2 batches. Otherwise, erroneous results might occur.

If the **Assign GL Account to Financial Transaction (GLASSGN2)** batch fails or aborts due to some reason, you can restart the batch over and over again with the same set of parameters.

Post Execution Check/Clean Up:

On successful completion of this batch, the following columns are updated in the CI_TRL_FT_GL or CI_FT_GL table depending on whether the GL account is assigned to a trial or actual financial transaction:

- GL_ACCT
- GLA_VAL_DT
- VALIDATE_SW

The GLA_VAL_DT column is updated when the GL account is valid and not when the GL account is invalid.

3.5 GL Download Staging (GLS)

Unless you activate a FT GL record, you cannot extract the record in a flat file. The **GL Download Staging (GLS)** batch is used to activate trial or actual financial transaction against which the GL account is assigned. Here, activating means stamping the GLDL batch control and incremental batch run number on the FT GL record.

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

Parameter Name	Mandatory (Yes or No)	Description
Override Maximum Number of Errors	No	Used to override the maximum number of errors after which the batch must be terminated.
Division	No	Used when you want to activate FT GL records of accounts belonging to a particular division.
Bill Generation Type	No	Used to indicate whether you want to activate trial or actual FT GL records. The valid values are: <ul style="list-style-type: none"> • Trial • Regular <div>Note: If you do not specify any value, the bill generation type is set to Regular.</div>
Trial Billing Batch Run Number	No	Used to indicate the trial billing batch run number whose trial bills' FT GL records you want to activate. <div>Note: This parameter is used only when you are activating trial FT GL records.</div>
Thread Pool Name	No	Used to specify the thread pool on which you want to execute the batch.

Note: If the **GL Download Staging (GLS)** batch fails or aborts due to some reason, you can restart the batch over and over again with the same set of parameters.

Post Execution Check/Clean Up:

On successful completion of this batch, the following columns are updated in the CI_TRL_FT_PROC or CI_FT_PROC table depending on whether trial or actual FT GL record is activated:

- BATCH_CD
- BATCH_NBR

3.6 GL Download Extract (GLDL)

The **GL Download Extract (GLDL)** batch is used to extract trial or actual FT GL records with the latest batch run number in a flat file. This flat file is then used for uploading data in the General Ledger application.

This batch is a single-threaded batch. You can specify the following parameters while executing this batch:

Parameter Name	Mandatory (Yes or No)	Description
Output Directory Path	Yes	Used to specify the path where you want to save the flat file.
Output File Name	Yes	Used to specify the name which you want to use for the flat file.
Override Maximum Number of Errors	No	Used to override the maximum number of errors after which the batch must be terminated.
Division	No	Used when you want to extract FT GL records of accounts belonging to a particular division.
Bill Generation Type	No	Used to indicate whether you want to extract trial or actual FT GL records. The valid values are: <ul style="list-style-type: none"> • Trial • Regular <div>Note: If you do not specify any value, the bill generation type is set to Regular.</div>
Trial Billing Batch Run Number	No	Used to indicate the trial billing batch run number whose trial bills' FT GL records you want to extract. <div>Note: This parameter is used only when you are extracting trial FT GL records.</div>
Thread Pool Name	No	Used to specify the thread pool on which you want to execute the batch.

Note: If the **GL Download Extract (GLDL)** batch fails or aborts due to some reason, you can restart the batch over and over again with the same set of parameters.

Post Execution Check/Clean Up:

On successful completion of this batch, the flat file is created at the specified location.

4. Pricing Management

This section provides detail information about the following batches:

- Identify Accounts for Repricing (C1-REPC1)
- Process Repricing Request and Persist Pricing for an Account (C1-REPC2)
- Process Fees Request and Persist Fees for an Account (C1-ACFEE)

4.1 Identify Accounts for Repricing (C1-REPC1)

On defining and editing the price item pricing of a price list which is assigned to an account or a person, the system creates an entry for the price item pricing in the CI_REPRC_ENTITY_DTL table. The **Identify Accounts for Repricing (C1-REPC1)** batch is used to identify the following in the specified order:

1. Price list on which the price item pricing (for which an entry is created in the CI_REPRC_ENTITY_DTL table) is assigned.
2. Account or Person on which the price list is assigned.
3. All accounts which belong to the person and its child persons when the price list is assigned to a person.

Once the accounts are identified, the system checks whether the identified accounts are eligible for repricing. If one or more accounts are eligible for repricing, a repricing request is created for the account, price item, and effective date combination in the CI_REPRC_REQ_DTL table. If the price assignment end date is specified, the system creates two repricing requests – one request where the effective date is set to the price assignment start date and another request where the effective date is set to the price assignment end date + 1 Day.

Note: Only accounts where the person and child person are the main customers are considered for repricing.

This batch considers only those entries in the CI_REPRC_ENTITY_DTL table which are in the **Pending (P)** or **Error (E)** status and whose effective date is earlier than or equal to the batch business date. This batch is a multi-threaded batch and the multi-threading is based on the account ID. You can specify the following parameters while executing this batch:

Parameter Name	Mandatory (Yes or No)	Description
Batch Business Date	No	Used to identify the entries in the CI_REPRC_ENTITY_DTL table which must be considered during the batch execution. <div style="border: 1px solid black; padding: 5px; margin-top: 10px;"> <p>Note:</p> <p>All entries whose effective date is earlier than or equal to the batch business date are considered during batch execution.</p> <p>If you do not specify any date, the batch business date is set to the current date.</p> </div>

Parameter Name	Mandatory (Yes or No)	Description
Thread Pool Name	No	Used to specify the thread pool on which you want to execute the batch.

Note: If the **Identify Accounts for Repricing (C1-REPC1)** batch fails or aborts due to some reason, you can restart the batch over and over again with the same set of parameters.

Post Execution Check/Clean Up:

On successful completion of this batch, a repricing request is created for the account, price item, and effective date combination in the CI_REPRC_REQ_DTL table. The BO_STATUS_CD column corresponding to the entry in the CI_REPRC_ENTITY_DTL table is set to either **Complete (C)** or **Error (E)** depending on whether the repricing request was created successfully or not. The date and time when the BO_STATUS_CD column is updated is stamped in the UPDT_DTTM column. The information about the batch which is executed is stamped in the BATCH_CD and BATCH_NBR columns.

4.2 Process Repricing Request and Persist Pricing for an Account (C1-REPC2)

The **Process Repricing Request and Persist Pricing for an Account (C1-REPC2)** batch is used to calculate and persist the rate for the account in the CI_PRCE_CALC table. It must be executed when a repricing request is created on:

- Editing the attributes and characteristics of an account
- Adding and deleting the usage amount and counter details of an account
- Editing the attributes and characteristics of a person
- Adding and deleting the usage amount and counter details of a person
- Assigning a price list to an account or editing the price list assignment details of an account
- Assigning a price list to a person or editing the price list assignment details of a person
- Defining or editing the price item pricing of an account
- Defining or editing the price item pricing of a person
- Defining and editing the price item pricing of a price list which is assigned to an account or a person

Note: In the latter case, this batch must be executed after executing the **Identify Accounts for Repricing (C1-REPC1)** batch.

It invokes the C1-PriceAccount business service which determines effective price item pricing for an account and persist calculated rates for price items whose price item type is set to **Rate**.

This batch considers only those repricing requests in the CI_REPRC_REQ_DTL table which are in the **Pending (P)** or **Error (E)** status and whose effective date is earlier than or equal to the batch business date. This batch is a multi-threaded batch and the multi-threading is based on the repricing request ID. You can specify the following parameters while executing this batch:

Parameter Name	Mandatory (Yes or No)	Description
Batch Business Date	No	Used to identify the repricing requests in the CI_REPRC_REQ_DTL table which must be considered during the batch execution. Note: All repricing requests whose effective date is earlier than or equal to the batch business date are considered during batch execution. If you do not specify any date, the batch business date is set to the current date.
Account ID	No	Used when you want to process repricing requests created for a particular account.
Thread Pool Name	No	Used to specify the thread pool on which you want to execute the batch.

Note: If the **Process Repricing Request and Persist Pricing for an Account (C1-REPC2)** batch fails or aborts due to some reason, you can restart the batch over and over again with the same set of parameters.

Post Execution Check/Clean Up:

On successful completion of this batch, the rate is calculated and persisted for the rate type price items in the CI_PRCE_CALC_K, CI_PRCE_CALC, CI_PRCE_CALC LN, and CI_PRCE_CALC_PARAMS tables.

The BO_STATUS_CD column corresponding to the repricing request in the CI_REPRC_REQ_DTL table is set to either **Complete (C)** or **Error (E)** depending on whether the rate was successfully calculated and persisted for the account, price item, and effective date combination. The date and time when the BO_STATUS_CD column is updated is stamped in the UPDT_DTTM column. The information about the batch which is executed is stamped in the BATCH_CD and BATCH_NBR columns.

4.3 Process Fees Request and Persist Fees for an Account (C1-ACFEE)

The **Process Fees Request and Persist Fees for an Account (C1-ACFEE)** batch is used to calculate and persist the fees for the account in the C1_FEES table. It must be executed when a user wants to store fees.

It invokes the C1-AccountFees business service which determines effective price item pricing for an account and persist calculated fees for price items of type **Only Pricing** whose type is set to **Fees**.

This batch considers all those accounts which satisfy the given input parameters. While doing price assignment user can specify the pricing frequency to be used from the already define schedule periods. If the Batch business date falls in between the schedule period and the data corresponding to that has not been already persisted then batch persists the fees based upon the advance pricing switch and price assignment type.

Advance Pricing Switch conditions:

Y- Batch business date should be on or after effective start date and on or before effective end date.

N- Batch business date should be on effective end date.

Note:

Effective start date - Price assignment start date and schedule period start date whichever is later.

Effective end date - Price assignment end date and schedule period end date whichever is earlier.

This batch is a multi-threaded batch and the multi-threading is based on the account Id. You can specify the following parameters while executing this batch:

Parameter Name	Mandatory (Yes or No)	Description
Product Code	No	Used when you want to calculate account fees for all accounts having this product linked.
Price Item Code	No	Used when you want to calculate account fees for given price item only.
Division	No	Used when you want to calculate account fees for all accounts of a specific division.
Thread Pool Name	No	Used to specify the thread pool on which you want to execute the batch.

Note: Fees for an account will be calculated for POST processing price assignment types only.

Post Execution Check/Clean Up:

On successful completion of this batch, the fees is calculated and persisted for the fees type price items in the C1_FEES, C1_FEES_CALC, C1_FEES_CALC_LN, C1_FEES_CALC_LN_CHAR, C1_FEES_CHAR, C1_FEES_K, and C1_FEES_PARAM, C1_FEES_SQ tables.

5. Funding Request

This section provides detail information about the Funding Request Periodic Monitor (C1-FNDRQ) batch.

5.1 Funding Request Periodic Monitor (C1-FNDRQ)

The **Funding Request Periodic Monitor (C1-FNDRQ)** batch process invokes the monitoring rules associated with the current state of the funding request.

By default, the process periodically monitors funding requests which are deferred for background processing.

Batch parameters govern whether the processing is further restricted by batch code, business object and status. You can specify the following parameters while executing this batch:

Parameter Name	Mandatory (Yes or No)	Description
Maintenance Object	Yes	Used to specify the name of maintenance object whose business objects are used for creating the funding requests.
Restrict By Batch Code	No	Used to specify a value as true to restrict processing of requests whose current status is linked to this batch code.
Restrict By Business Object	No	Used to specify a business object code to limit the process for the requests linked to this business object.
Restrict By Status Code	No	Used to specify a status code to limit the process for the requests in this status.
Thread Pool Name	No	Used to specify the thread pool on which you want to execute the batch.

Post Execution Check/Clean Up:

On successful completion of this batch, for debit bills, payment events are created or the records are inserted in the CI_BILL_ACH table and for credit bills, adjustments are created and inserted in C1_FUNDING_REQ_DTLS_ADJ table.

The status of the records is either set to PROCESSED or ERROR in the C1_FUNDING_REQ_DTLS table depending on whether the payments or adjustments are created successfully or not. Error messages corresponding to the record are inserted in C1_FUNDING_REQ_DTLS_EXCP table. The BO_STATUS_CD column corresponding to the funding request in the C1_FUNDING_REQ table is set to COMPLETE.

6. Offset Request

This section provides detail information about the Offset Request Periodic Monitor (C1-OFSRQ) batch.

6.1 Offset Request Periodic Monitor (C1-OFSRQ)

The **Offset Request Periodic Monitor (C1-OFSRQ)** batch process invokes monitoring rules associated with the current state of the offset request. By default, the process periodically monitors offset requests which are deferred for background processing.

Batch parameters govern whether the processing is further restricted by batch code, business object and status. You can specify the following parameters while executing this batch:

Parameter Name	Mandatory (Yes or No)	Description
Maintenance Object	Yes	Used to specify the name of maintenance object whose business objects are used for creating the offset requests.
Restrict By Batch Code	No	Used to specify a value as true to restrict processing of requests whose current status is linked to this batch code.
Restrict By Business Object	No	Used to specify a business object code to limit the process for the requests linked to this business object.
Restrict By Status Code	No	Used to specify a status code to limit the process for the requests in this status.
Thread Pool Name	No	Used to specify the thread pool on which you want to execute the batch.

Post Execution Check/Clean Up:

On successful completion of this batch, offset distribution algorithm will be invoked which will create the transfer adjustments. Adjustment IDs will be updated in C1_OFFSET_REQ_ADJ table. The BO_STATUS_CD column corresponding to the offset request in the C1_OFFSET_REQ table is set to PROCESSED or DRAFT depending on whether the adjustments were successfully created or some error was encountered. In case of error, in creating adjustments, adjustments will not be created and the error will be logged in Offset Request Logs.

7. Hold Request

This section provides detail information about the following batches:

- Hold Request Periodic Monitor (C1-HLDRQ)
- Hold Request Monitor (C1-HLMON)
- Delete Pending Bill segments and Bills (C1-DELBI)

7.1 Hold Request Periodic Monitor (C1-HLDRQ)

The **Hold Request Periodic Monitor (C1-HLDRQ)** batch is used to monitor or check whether there are any hold requests in the **Deferred Processing** status. If there is a hold request in the **Deferred Processing** status, the batch changes the status of the hold request to **Active**. Then, the algorithm attached to the **Active** status is triggered.

This batch is a multi-threaded batch. The multi-threading is based on hold request ID and chunks for multi-threading are created based on numerical distribution of hold request ID. However, we recommend you to execute this batch in the single-threaded mode. Otherwise, erroneous results might occur. While executing this batch in the single-threaded mode, the following fields must be set to zero:

- Thread Count
- Override Nbr Records to Commit

You can specify the following parameters while executing this batch:

Parameter Name	Mandatory (Yes or No)	Description
Maintenance Object	Yes	Used to indicate that you want to monitor hold requests which are created using the business objects of a particular maintenance object. Note: By default, the parameter value is set to C1-HOLDREQ.
Batch Control	No	Used when you want to monitor hold requests whose current status is linked to the Hold Request Periodic Monitor (C1-HLDRQ) batch. The valid value is true .
Business Object	No	Used when you want to monitor hold requests which are created using a particular business object.
Status	No	Used when you want to monitor hold requests 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.

Parameter Name	Mandatory (Yes or No)	Description
Thread Pool Name	No	Used to specify the thread pool on which you want to execute the batch.

Note: If the **Hold Request Periodic Monitor (C1-HLDRQ)** batch fails or aborts due to some reason, you can restart the batch over and over again with the same set of parameters.

Post Execution Check/Clean Up:

On successful completion of this batch, the status of the hold request is changed from **Deferred Processing** to **Active**. If any error occurs while executing the batch, the status of the hold request is changed from **Deferred Processing** to **Draft** and a message appears in the **Log** tab of the **Hold Request** screen which indicates the error that occurred during batch execution.

7.2 Hold Request Monitor (C1-HLMON)

The **Hold Request Monitor (C1-HLMON)** batch is used to monitor or check whether there are any hold requests in the **Active** or **Released** status. If there is a hold request in the **Active** status, the batch does the following:

If...	Then...
The hold request is created for an account, the bill generation process is on hold, and the entity start date in the hold request is earlier than or equal to the batch business date	<p>The entity end date or process end date whichever is earlier is set in the Bill After field for the respective account.</p> <p>Note: The Bill After field appears in the Main tab of the Account screen.</p>
The hold request is created for an account, the overdue process is on hold, and the entity start date in the hold request is earlier than or equal to the batch business date	<p>The entity end date or process end date whichever is earlier is set in the Postpone Credit Review Until field for the respective account.</p> <p>Note: The Postpone Credit Review Until field appears in the C & C tab of the Account screen.</p>
The hold request is created for an account, the auto pay process is on hold, and the entity start date in the hold request is earlier than or equal to the batch business date	<p>The entity end date or process end date whichever is earlier is set in the Defer Auto Pay Date field for the respective account.</p> <p>Note: The Defer Auto Pay Date field appears in the Auto Pay tab of the Account screen.</p>
The hold request is created for an account, the auto pay process is on hold, and the entity end date in the hold request is earlier than the batch business date	The automatic payment details are updated in the CI_BILL_ACH table.

If...	Then...
The hold request end date is earlier than the batch business date	The status of the hold request is changed to Released . In addition, the entity end date and process end date is set to the hold request end date when the entity end date or process end date is either not specified or is a future date.

If there is a hold request in the **Released** status, the batch does the following:

If...	Then...
The hold request is created for an account, the auto pay process is on hold, and the entity end date in the hold request is earlier than or equal to the batch business date	The automatic payment details are updated in the CI_BILL_ACH table.

This batch is a multi-threaded batch. The multi-threading is based on hold request ID and chunks for multi-threading are created based on numerical distribution of hold request ID. However, we recommend you to execute this batch in the single-threaded mode. Otherwise, erroneous results might occur. While executing this batch in the single-threaded mode, the following fields must be set to zero:

- Thread Count
- Override Nbr Records to Commit

You can specify the following parameters while executing this batch:

Parameter Name	Mandatory (Yes or No)	Description
Thread Pool Name	No	Used to specify the thread pool on which you want to execute the batch.
Number of Release Days	No	Used when you do not want to monitor or check hold requests which were released earlier than the specified number of days. For example, if the batch business date is set to 02-20-2017 and the number of release days is set to 10, then this batch will not consider hold requests which were released earlier than 02-11-2017.
Release Status Reason	No	Used to indicate the reason why you want to release the hold request. <div style="border: 1px solid black; padding: 5px;"> Note: You need to ensure that the specified release status reason is already defined for the C1-HoldRequest business object in the Status Reason screen. </div>

Parameter Name	Mandatory (Yes or No)	Description
Alert Type Code	No	<p>Used to indicate the type of alert whose end date must be updated on the account when the hold request is released.</p> <div> <p>Note:</p> <p>The alert end date is set to the date when the hold request is released.</p> <p>You need to ensure that the alert type used while activating and releasing the hold requests must be same. Therefore, always specify the same alert type in the following entities - C1-HOLDACTV, C1-RELENTITY, and C1-HLMON.</p> </div>

Note:

If you do not specify the batch business date, it is set to the current date.

If the **Hold Request Monitor (C1-HLMON)** batch fails or aborts due to some reason, you can restart the batch over and over again with the same set of parameters.

Post Execution Check/Clean Up:

On successful completion of this batch, the following actions take place for:

- **Active hold requests** - The date is updated in the **Bill After**, **Postpone Credit Review Until**, and **Defer Auto Pay Date** fields depending on the process which is kept on hold for the account. The automatic payment details are updated in the **CI_BILL_ACH** table when the account's auto pay process is kept on hold and the entity end date in the hold request is earlier than the batch business date. In addition, the status of the active hold requests is changed to **Released** when the hold request end date is earlier than the batch business date.
- **Released hold requests** - The automatic payment details are updated in the **CI_BILL_ACH** table when the account's auto pay process is kept on hold and the entity end date in the hold request is earlier than or equal to the batch business date.

7.3 Delete Pending Bill Segments and Bills (C1-DELBI)

The **Delete Pending Bill Segments and Bills (C1-DELBI)** batch is used to delete the following for accounts' whose bill generation process is kept on hold through a hold request:

- All bill segments of the pending bills which are in the **Freezable** or **Error** status
- All pending bills

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

Parameter Name	Mandatory (Yes or No)	Description
Chunk Size	Yes	Used to specify the number of bill segments you want to execute in each work unit.
Thread Pool Name	No	Used to specify the thread pool on which you want to execute the batch.

Note: If the **Delete Pending Bill Segments and Bills (C1-DELBI)** batch fails or aborts due to some reason, you can restart the batch over and over again with the same set of parameters.

Post Execution Check/Clean Up:

On successful completion of this batch, all bill segments of the pending bill which are in the **Freezable** or **Error** status are deleted along with the pending bill of the account whose bill generation process is kept on hold through a hold request. The batch run number through which the pending bill and the corresponding bill segments are deleted is stored corresponding to the hold request in the **CI_BILL_DEL_REQUEST** table.

8. Upload Request

This section provides detail information about the **Upload Request Periodic Monitor (C1-UPLRQ)** batch.

8.1 Upload Request Periodic Monitor (C1-UPLRQ)

The **Upload Request Periodic Monitor (C1-UPLRQ)** batch is used to monitor or check upload requests or adjustment upload requests depending on whether the **Maintenance Object** parameter is set to **C1-UPLOADREQ** or **C1-UPLREQ**.

The following table describes the batch in detail:

If the Maintenance Object parameter is set to...	Then...
C1-UPLOADREQ	<p>The Upload Request Periodic Monitor (C1-UPLRQ) batch is used to monitor or check whether there are any upload requests in the Deferred Processing status. If there is an upload request in the Deferred Processing status, the batch changes the status of the upload request to Processing. Then, the algorithm attached to the Processing status is triggered which creates the entities and accordingly changes the status of the upload request to Processed.</p> <p>This batch is a multi-threaded batch. The multi-threading is based on upload request ID and chunks for multi-threading are created based on numerical distribution of upload request ID. However, we recommend you to execute this batch in the single-threaded mode. Otherwise, erroneous results might occur. While executing this batch in the single-threaded mode, the following fields must be set to zero:</p> <ul style="list-style-type: none"> • Thread Count • Override Nbr Records to Commit
C1-UPLREQ	<p>The Upload Request Periodic Monitor (C1-UPLRQ) batch is used to monitor or check whether there are any adjustment upload requests in the Deferred status. If there is an adjustment upload request in the Deferred status, the batch changes the status of the adjustment upload request to Creating Adjustment. Then, the algorithm attached to the Creating Adjustment status is triggered which creates the adjustments.</p> <p>This batch is a single-threaded batch.</p>

You can specify the following parameters while executing this batch:

Parameter Name	Mandatory (Yes or No)	Description
Maintenance Object	Yes	<p>Used to indicate that you want to monitor upload requests or adjustment upload requests created using the business objects of a particular maintenance object.</p> <div> Note: If you want to monitor upload requests, you need to set the value of this parameter to C1-UPLOADREQ. And, if you want to monitor adjustment upload requests, you need to set the value of this parameter to C1-UPLREQ. By default, the parameter value is set to C1-UPLOADREQ. </div>
Batch Control	No	Used when you want to monitor upload requests or adjustment upload requests whose current status is linked to the Upload Request Periodic Monitor (C1-UPLRQ) batch. The valid value is true .
Business Object	No	Used when you want to monitor upload requests or adjustment upload requests which are created using a particular business object.
Status	No	<p>Used when you want to monitor upload requests or adjustment upload requests which are in a particular status.</p> <div> Note: This parameter is useful when this batch is invoked from more than one status in the lifecycle of the business object. </div>
Override Maximum Number of Errors	No	Used to override the maximum number of errors after which the batch must be terminated.
Upload Request Type	No	Used when you want to monitor upload requests or adjustment upload requests which are created using a particular upload request type.
Thread Pool Name	No	Used to specify the thread pool on which you want to execute the batch.

Note: If the **Upload Request Periodic Monitor (C1-UPLRQ)** batch fails or aborts due to some reason, you can restart the batch over and over again with the same set of parameters.

Post Execution Check/Clean Up:

The following table explains what happens on successful completion of this batch:

If the Maintenance Object parameter is set to...	Then...
C1-UPLOADREQ	<ol style="list-style-type: none">1. The status of the upload request is changed from Deferred Processing to Processing.4. The entities are created using the entity business object specified in the respective upload request type.5. Finally, the status of the upload request is changed to Processed.
C1-UPLREQ	<ol style="list-style-type: none">1. The status of the adjustment upload request is changed from Deferred to Creating Adjustment.6. The adjustment is created for each adjustment record in the adjustment upload request.7. Finally, the status of the adjustment upload request is changed to Completed.

9. Inbound Message

This section provides detail information about the following batches:

- Customer Inbound Message Periodic Monitor (C1-CUSIN)
- Membership Inbound Message Periodic Monitor (C1-MEMIN)

9.1 Customer Inbound Message Periodic Monitor (C1-CUSIN)

The **Customer Inbound Message Periodic Monitor (C1-CUSIN)** batch is used to monitor or check whether there are any customer inbound messages in the **Pending** status. If there is a customer inbound message in the **Pending** status, the batch changes the status of the customer inbound message to **Validation**. Then, the algorithm attached to the **Validation** status is triggered. If the validation is successful, the batch changes the status of the customer inbound message to **Processing**. Then, the algorithm attached to the **Processing** status is triggered. If the processing is successful, the batch creates or updates the entities, such as person, account, policy, policy plan, and/or billable charge (as required), and then changes the status of the customer inbound message to **Processed**. However, if an error occurs while validating or processing a customer inbound message, the batch changes the status of the customer inbound message to **Rejected** and the algorithm attached to the **Rejected** status is triggered.

A To Do is created using the To Do type specified in the **C1-TDCRINMSG** algorithm and assigned to the users with the To Do role specified in the **C1-TDCRINMSG** algorithm. The system checks whether the maximum retry attempts are met. If the retry attempt is less than or equal to the value defined in the **Maximum Retry** parameter of the **C1-TODORETRY** algorithm, the status of the inbound message is changed to **Pending**. However, if the retry attempt is greater than the value defined in the **Maximum Retry** parameter of the **C1-TODORETRY** algorithm, the status of the inbound message remains as **Rejected**.

This batch is a multi-threaded batch. The multi-threading is based on external source ID and chunks for multi-threading are created based on numerical distribution of external source ID. However, we recommend you to execute this batch in the single-threaded mode. Otherwise, erroneous results might occur. While executing this batch in the single-threaded mode, the following fields must be set to zero:

- Thread Count
- Override Nbr Records to Commit

You can specify the following parameters while executing this batch:

Parameter Name	Mandatory (Yes or No)	Description
Batch Control	No	Used when you want to monitor customer inbound messages whose current status is linked to the Customer Inbound Message Periodic Monitor (C1-CUSIN) batch. The valid value is true .

Parameter Name	Mandatory (Yes or No)	Description
Inbound Message Type	No	Used when you want to monitor customer inbound messages which are created using a particular inbound message type.
Business Object	No	Used when you want to monitor customer inbound messages which are created using a particular business object.
Status	No	Used when you want to monitor customer inbound messages 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 Number of Errors	No	Used to override the maximum number of errors after which the batch must be terminated.
Thread Pool Name	No	Used to specify the thread pool on which you want to execute the batch.

Note: If the **Customer Inbound Message Periodic Monitor (C1-CUSIN)** batch fails or aborts due to some reason, you can restart the batch over and over again with the same set of parameters.

Post Execution Check/Clean Up:

If a customer inbound message is successfully validated and processed, the status of the customer inbound message is changed from **Pending** to **Processed**. If any error occurs while validating or processing a customer inbound message, the batch changes the status of the customer inbound message to **Rejected** and a message appears in the **Log** tab of the **Inbound Message** screen which indicates the error that occurred during batch execution.

9.2 Membership Inbound Message Periodic Monitor (C1-MEMIN)

The **Membership Inbound Message Periodic Monitor (C1-MEMIN)** batch is used to monitor or check whether there are any membership inbound messages in the **Pending** status. If there is a membership inbound message in the **Pending** status, the batch changes the status of the membership inbound message to **Validation**. Then, the algorithm attached to the **Validation** status is triggered. If the validation is successful, the batch changes the status of the membership inbound message to **Processing**. Then, the algorithm attached to the **Processing** status is triggered. If the processing is successful, the batch creates or updates the entities, such as membership and/or billable charge (as required), and then changes the status of the membership inbound message to **Processed**. However, if an error occurs while validating or processing a membership inbound message, the batch changes the status of the membership inbound message to **Rejected** and the algorithm attached to the **Rejected** status is triggered.

A To Do is created using the To Do type specified in the **C1-TDCRINMSG** algorithm and assigned to the users with the To Do role specified in the **C1-TDCRINMSG** algorithm. The system checks whether the maximum retry attempts are met. If the retry attempt is less than or equal to the value defined in the **Maximum Retry** parameter of the **C1-TODORETRY** algorithm, the status of the inbound message is changed to **Pending**. However, if the retry attempt is greater than the value defined in the **Maximum Retry** parameter of the **C1-TODORETRY** algorithm, the status of the inbound message remains as **Rejected**.

This batch is a multi-threaded batch. The multi-threading is based on external source ID and chunks for multi-threading are created based on numerical distribution of external source ID. However, we recommend you to execute this batch in the single-threaded mode. Otherwise, erroneous results might occur. While executing this batch in the single-threaded mode, the following fields must be set to zero:

- Thread Count
- Override Nbr Records to Commit

You can specify the following parameters while executing this batch:

Parameter Name	Mandatory (Yes or No)	Description
Batch Control	No	Used when you want to monitor membership inbound messages whose current status is linked to the Membership Inbound Message Periodic Monitor (C1-MEMIN) batch. The valid value is true .
Inbound Message Type	No	Used when you want to monitor membership inbound messages which are created using a particular inbound message type.
Business Object	No	Used when you want to monitor membership inbound messages which are created using a particular business object.
Status	No	Used when you want to monitor membership inbound messages 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.
Number of Characters (External Source ID)	No	Used when you want to consider membership inbound messages, whose first few characters of the external source ID are same, in the same work unit. For example, let us assume that there are four membership inbound messages with external source ID – 123456, 123654, 125647, and 123879. Now, when you specify the value for this parameter as 3, the batch considers the following membership inbound messages - 123456, 123654, and 123879 in one chunk and 125647 in another chunk.

Parameter Name	Mandatory (Yes or No)	Description
Override Maximum Number of Errors	No	Used to override the maximum number of errors after which the batch must be terminated.
Thread Pool Name	No	Used to specify the thread pool on which you want to execute the batch.

Note: If the **Membership Inbound Message Periodic Monitor (C1-MEMIN)** batch fails or aborts due to some reason, you can restart the batch over and over again with the same set of parameters.

Post Execution Check/Clean Up:

If a membership inbound message is successfully validated and processed, the status of the membership inbound message is changed from **Pending** to **Processed**. If any error occurs while validating or processing a membership inbound message, the batch changes the status of the membership inbound message to **Rejected** and a message appears in the **Log** tab of the **Inbound Message** screen which indicates the error that occurred during batch execution.

10. Payment Agreement Request

This section provides detail information about the following batches:

- Payment Agreement Request Periodic Monitor (C1-PAREQ)
- Generate Automatic Payment for Payment Agreement Request (C1-APPAB)

10.1 Payment Agreement Request Periodic Monitor (C1-PAREQ)

The **Payment Agreement Request Periodic Monitor (C1-PAREQ)** batch is used to monitor or check whether there are any payment agreement requests in the **Active** status. If there is a payment agreement request in the **Active** status, the system checks whether the total unpaid amount of the bills (included in the payment agreement request) is equal to zero and whether each bill is fully matched. If so, the status of the payment agreement request is changed to **Kept Promise**. However, if the total unpaid amount of the bills is not equal to zero, the system checks whether the batch business date is later than the schedule date and does not fall within the grace period. If so, the system checks whether the total unpaid amount is greater than the total future schedule amount. If so, the status of the payment agreement request is changed to **Broken Promise**. However, if the batch business date is earlier than the schedule date or falls within the grace period, or the total unpaid amount is less than the total future schedule amount, the status of the payment agreement request remains in **Active**.

This batch is a multi-threaded batch. The multi-threading is based on payment agreement request ID and chunks for multi-threading are created based on numerical distribution of payment agreement request ID. However, we recommend you to execute this batch in the single-threaded mode. Otherwise, erroneous results might occur. While executing this batch in the single-threaded mode, the following fields must be set to zero:

- Thread Count
- Override Nbr Records to Commit

You can specify the following parameters while executing this batch:

Parameter Name	Mandatory (Yes or No)	Description
Maintenance Object	Yes	Used to indicate that you want to monitor payment agreement requests which are created using the business objects of a particular maintenance object. <div>Note: By default, the parameter value is set to C1-PAREQ.</div>
Batch Control	No	Used when you want to monitor payment agreement requests whose current status is linked to the Payment Agreement Request Periodic Monitor (C1-PAREQ) batch. The valid value is true .

Parameter Name	Mandatory (Yes or No)	Description
Business Object	No	Used when you want to monitor payment agreement requests which are created using a particular business object.
Status	No	Used when you want to monitor payment agreement requests 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.
Thread Pool Name	No	Used to specify the thread pool on which you want to execute the batch.

Note: If the **Payment Agreement Request Periodic Monitor (C1-PAREQ)** batch fails or aborts due to some reason, you can restart the batch over and over again with the same set of parameters.

Post Execution Check/Clean Up:

On successful completion of this batch, the status of the payment agreement request is changed to either:

- **Kept Promise** (when the total unpaid amount of the bills (included in the payment agreement request) is equal to zero and each bill is fully matched)
- **Broken Promise** (when the total unpaid amount is greater than the total future schedule amount)

The status of the payment agreement request remains in the **Active** status when the batch business date is earlier than the schedule date or falls within the grace period, or the total unpaid amount is less than the total future schedule amount.

If any error occurs while executing the batch, a message appears in the **Log** tab of the **Payment Agreement Request** screen which indicates the error that occurred during batch execution.

10.2 Generate Automatic Payment for Payment Agreement Request (C1-APPAB)

The **Generate Automatic Payment for Payment Agreement Request (C1-APPAB)** batch is used to monitor or check whether there are any payment agreement requests in the **Active** status. If there is a payment agreement request in the **Active** status, the system checks whether the account for which the payment agreement request is created is eligible for automatic payment and the defer auto pay date (if any) defined for the account is earlier than the batch business date. If so, the system checks whether the extract date of the unpaid bill (with the earliest due date) is earlier than the schedule date. If so, the system creates the automatic payment for the unpaid bill on the schedule date. However, if the account is not eligible for automatic payment, or the defer auto pay date is equal to or later than batch business date, or the extract date is equal to or later than the schedule date, the automatic payment is not generated for the account.

This batch is a multi-threaded batch. The multi-threading is based on payment agreement request ID and chunks for multi-threading are created based on numerical distribution of payment agreement request ID.

You can specify the following parameters while executing this batch:

Parameter Name	Mandatory (Yes or No)	Description
Override Maximum Number of Errors	No	Used to override the maximum number of errors after which the batch must be terminated.
Thread Pool Name	No	Used to specify the thread pool on which you want to execute the batch.

Note: If the **Generate Automatic Payment for Payment Agreement Request (C1-APPAB)** batch fails or aborts due to some reason, you can restart the batch over and over again with the same set of parameters.

Post Execution Check/Clean Up:

On successful completion of this batch, the records are added in the following tables when the automatic payment is created for the account:

- CI_PAY
- CI_PAY_EVENT
- CI_PAY_TNDR

The status of the automatic payment is set to **Incomplete**.

11. Accruals

This section provides detail information about the following batches:

- Accrual Calculation (C1_ACCAL)
- Accrual Creation (C1_ACCRE)
- Accrual Reversal (C1_ACREV)
- Create To Do for Account without Accrual Cycle (C1_ACELG)

11.1 Accrual Calculation (C1_ACCAL)

The **Accrual Calculation (C1_ACCAL)** batch is used to calculate the accrual amount for accounts. It considers the batch business date as the accrual date and finds the accrual cycle within which the accrual date is present. Once the accrual cycle is identified, the batch finds all accounts where the accrual cycle is defined. It also identifies the accrual type which is effective for the division (to which the account belongs) on the batch business date. Based on the options defined in the accrual type, the system accordingly calculates the accrual amount for the accounts.

This batch executes the algorithms attached at the following spots in the accrual type in the specified order:

1. Accrual Eligibility
2. Accrual Pre Processing
3. Accrual Calculation
4. Accrual Post Calculation

If an algorithm created using the **C1_ACCR_ELG** algorithm type is attached to the **Accrual Eligibility** algorithm spot in the accrual type, the system checks whether the account is a member of an invoicing group or whether the account is included in an active construct. If the account is a member of an invoicing group or included in an active construct, the account is not considered for accrual calculation.

Note: The system checks whether the account is a member of an invoicing group only when the **includeIGAEligibility** parameter in an algorithm created using the **C1_ACCR_ELG** algorithm type is set to **Y**.

If an algorithm created using the **C1_ACCR_POCA** algorithm type is attached to the **Accrual Post Calculation** algorithm spot in the accrual type, the status of the accrual is changed to **Approval in Progress** once the accrual is calculated for the account. However, if there is no algorithm attached to the **Accrual Post Calculation** algorithm spot in the accrual type, the status of the accrual is changed to **Ready For Accrual** once the accrual is calculated for the account.

Note: This batch caches division related information, such as work calendar, currency conversion algorithms, rounding type, and accrual types, which is required for accrual calculation. If any of these attributes change for a division, you must execute the **Flush All Caches (F1-Flush)** batch before executing the **Accrual Calculation (C1_ACCAL)** batch. The **Flush All Caches (F1-Flush)** batch is used to clean the application cache.

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. You can specify the following parameters while executing this batch:

Parameter Name	Mandatory (Yes or No)	Description
Division	Yes	Used when you want to calculate accruals for accounts belonging to a particular division.
Accrual Cycle	No	Used when you want to calculate accruals for accounts having a particular accrual cycle.
Delete Manual Accrual	No	<p>Used when you want to delete the accrual which is manually created for the account on the accrual date. The valid values are:</p> <ul style="list-style-type: none"> • Y • N <div style="border: 1px solid black; padding: 5px; margin-top: 10px;"> <p>Note: If you do not specify any value, by default, the parameter value is set to N.</p> <p>If you set the parameter value to N, the batch will throw an error when the accrual in the Ready For Accrual, Processed, or Approval in Progress status already exists for the account on the accrual date.</p> </div>
Override Maximum Number of Errors	No	Used to override the maximum number of errors after which the batch must be terminated.
Thread Pool Name	No	Used to specify the thread pool on which you want to execute the batch.

Note: If the **Accrual Calculation (C1_ACCAL)** batch fails or aborts due to some reason, you can restart the batch over and over again with the same set of parameters.

Post Execution Check/Clean Up:

On successful completion of this batch, the accrual amount is calculated for the accounts that meet the search criteria. The status of the accrual is changed to either **Approval in Progress** or **Ready For Accrual** depending on whether the accrual post calculation algorithm is attached on the accrual type.

11.2 Accrual Creation (C1_ACCRE)

The **Accrual Creation (C1_ACCRE)** batch is used to check whether there are any accruals in the **Ready for Accrual** or **Processed** status. It considers only those accruals whose accrual date is earlier than or equal to the batch business date. If there is an accrual in the **Ready for Accrual** or **Processed** status, the accrual object (i.e. adjustment) is created for the account. Based on the adjustment type defined in the accrual type, the system accordingly creates the accrual adjustment for the account. Once the accrual adjustment is created for the account, the status of the accrual is changed to **Accrued**.

This batch executes the algorithms attached at the following spots in the accrual type in the specified order:

1. Accrual Creation
2. Accrual Post Processing

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

Parameter Name	Mandatory (Yes or No)	Description
Division	Yes	Used when you want to create accrual adjustment for accounts belonging to a particular division.
Accrual Cycle	No	Used when you want to create accrual adjustment for accounts having a particular accrual cycle.
Validation Required	No	Used to indicate whether the system should check if there are any accruals (whose accrual date is earlier than or equal to the batch business date) in the Approval in Progress status. The valid values are: <ul style="list-style-type: none"> • Y • N If you set the parameter value to Y , the batch will end in error when there are accruals (whose accrual date are earlier than or equal to the batch business date) in the Approval in Progress status. However, if you set the parameter value to N , the batch will skip the accruals (if any) in the Approval in Progress status and creates accrual adjustment for accruals in the Ready For Accrual or Processed status.
Thread Pool Name	No	Used to specify the thread pool on which you want to execute the batch.

Note: If the **Accrual Creation (C1_ACCRE)** batch fails or aborts due to some reason, you can restart the batch over and over again with the same set of parameters.

Post Execution Check/Clean Up:

On successful completion of this batch, the accrual adjustment is created for accounts and the status of the accrual is changed to **Accrued**.

11.3 Accrual Reversal (C1_ACREV)

The **Accrual Reversal (C1_ACREV)** batch is used to reverse the accrual for the accounts. It only considers the accruals where the batch business date is between the reversal start and end dates. On reversing the accrual, the accrual adjustments are cancelled and the status of the accrual is changed to **Reversed**.

This batch executes the algorithms attached at the following spots in the accrual type in the specified order:

1. Accrual Reversal Pre Processing
2. Accrual Reversal
3. Accrual Reversal Post Processing

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

Parameter Name	Mandatory (Yes or No)	Description
Division	Yes	Used when you want to reverse accrual for accounts belonging to a particular division.
Accrual Cycle	No	Used when you want to reverse accrual for accounts having a particular accrual cycle.
Thread Pool Name	No	Used to specify the thread pool on which you want to execute the batch.

Note: If the **Accrual Reversal (C1_ACREV)** batch fails or aborts due to some reason, you can restart the batch over and over again with the same set of parameters.

Post Execution Check/Clean Up:

On successful completion of this batch, the accrual is reversed for the accounts and the accrual adjustments are cancelled. In addition, the status of the accrual is changed to **Reversed**.

11.4 Create To Do for Account without Accrual Cycle (C1_ACELG)

The **Create To Do for Account without Accrual Cycle (C1_ACELG)** batch is used to create a To Do for all such accounts which are eligible for accrual, but the accrual cycle is not defined either at the account or division level.

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. You can specify the following parameters while executing this batch:

Parameter Name	Mandatory (Yes or No)	Description
Division	Yes	Used when you want to create To Do for accounts belonging to a particular division.
Thread Pool Name	No	Used to specify the thread pool on which you want to execute the batch.

Note: If the **Create To Do for Account without Accrual Cycle (C1_ACELG)** batch fails or aborts due to some reason, you can restart the batch over and over again with the same set of parameters.

Post Execution Check/Clean Up:

On successful completion of this batch, a To Do is created for an account which is eligible for accrual, but the accrual cycle is not defined either at the account or division level.

12. Earnings Credit Rate

This section provides detail information about the following batches:

- Financial Transaction Distribution Monitor (C1-FTDRD)
- Financial Transaction Distribution (C1-FTDTS)

12.1 Financial Transaction Distribution Monitor (C1-FTDRD)

The **Financial Transaction Distribution Monitor (C1-FTDRD)** batch is used to monitor or check whether there are any financial transaction distribution objects in the **Pending** status. If there is a financial transaction distribution object in the **Pending** status, the system validates the financial transaction distribution object. If the validation is successful, the status of the financial transaction distribution object is changed to **Validated**. Then, depending on the following condition which is met, the status of the financial transaction distribution object is changed from **Validated** to **Expire**, **Write Off**, or **Refund**.

If...	Then...
The credit or debit amount on the financial transaction distribution object is expired	The status of the financial transaction distribution object is changed to Expire and the expiration adjustment is created using the expiration adjustment type specified in the respective financial transaction distribution type. Once the expiration adjustment is created, the status of the financial transaction distribution object is changed to Complete .
The credit or debit amount on the financial transaction distribution object is less than or equal to the write off threshold amount	The status of the financial transaction distribution object is changed to Write Off and the write off adjustment is created using the write off adjustment type specified in the respective financial transaction distribution type. Once the write off adjustment is created, the status of the financial transaction distribution object is changed to Complete .
There is excess credit amount on the financial transaction distribution object at the end of the year and the source account belongs to a customer class where refund is allowed	The status of the financial transaction distribution object is changed to Refund and the refund adjustment is created using the refund adjustment type specified in the respective financial transaction distribution type. Once the refund adjustment is created, the status of the financial transaction distribution object is changed to Complete .

If none of the above conditions are met, the status of the financial transaction distribution object remains as **Validated**. However, if the validation fails for a financial transaction distribution object, the status of the financial transaction distribution object remains as **Pending**.

This batch is a multi-threaded batch. The multi-threading is based on financial transaction distribution ID and chunks for multi-threading are created based on numerical distribution of financial transaction distribution ID. However, we recommend you to execute this batch in the single-threaded mode. Otherwise, erroneous results might occur. While executing this batch in the single-threaded mode, the following fields must be set to zero:

- Thread Count
- Override Nbr Records to Commit

You can specify the following parameters while executing this batch:

Parameter Name	Mandatory (Yes or No)	Description
Maintenance Object	Yes	Used to indicate that you want to monitor financial transaction distribution objects which are created using the business objects of a particular maintenance object. Note: By default, the parameter value is set to C1-FTDIST .
Batch Control	No	Used when you want to monitor financial transaction distribution objects whose current status is linked to the Financial Transaction Distribution Monitor (C1-FTDRD) batch. The valid value is true .
Financial Transaction Distribution Type	No	Used when you want to monitor financial transaction distribution objects which are controlled through a particular financial transaction distribution type.
Business Object	No	Used when you want to monitor financial transaction distribution objects which are created using a particular business object.
Status	No	Used when you want to monitor financial transaction distribution objects 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 Number of Errors	No	Used to override the maximum number of errors after which the batch must be terminated.

Note: If the **Financial Transaction Distribution Monitor (C1-FTDRD)** batch fails or aborts due to some reason, you can restart the batch over and over again with the same set of parameters.

Post Execution Check/Clean Up:

On successful completion of this batch, the status of the financial transaction distribution object is changed to **Validated** when its validation is successful. However, if the validation fails, it remains in the **Pending** status.

Depending on the above mentioned condition that is met, the status of the financial transaction distribution object is changed from **Validated** to **Expire**, **Write Off**, or **Refund**. Accordingly, the expiration, write off, or refund adjustment is created. However, if none of the above mentioned conditions are met, the status of the financial transaction distribution object remains as **Validated**.

12.2 Financial Transaction Distribution (C1-FTDTS)

The **Financial Transaction Distribution (C1-FTDTS)** batch is used to monitor or check whether there are any financial transaction distribution objects in the **Validated** status. If there is a financial transaction distribution object in the **Validated** status, the system changes the status of the financial transaction distribution object to **Apply Rule** and executes the algorithm attached to the **Apply Rule** system event on the financial transaction distribution rule version which is effective on the batch business date. Once the credit or debit distribution array of accounts is created, the status of the financial transaction distribution object is changed to **Distributed** and then the algorithm attached to the **Distribute** system event on the financial transaction distribution rule version which is effective on the batch business date is executed.

On the credit or debit amount distribution, two adjustments are created – one against the source contract and another against the bill segment or adjustment's contract to which the amount is distributed. Once the credit or debit amount on the financial transaction distribution object is fully distributed, the status of the financial transaction distribution object is changed to **Complete**. However, if the credit or debit amount on the financial transaction distribution object is not yet fully distributed, the status of the financial transaction distribution object is changed to **Validated**.

This batch is a single-threaded batch. You can specify the following parameters while executing this batch:

Parameter Name	Mandatory (Yes or No)	Description
Account ID	No	Used when you want to distribute the credit or debit amount on the financial transaction distribution objects which are created for a particular account. You can specify comma separated values for this parameter.
Division	No	Used when you want to distribute the credit or debit amount on the financial transaction distribution objects which are created for accounts belonging to a particular division.

Parameter Name	Mandatory (Yes or No)	Description
Business Object	Yes	Used when you want to distribute the credit or debit amount on the financial transaction distribution objects which are created using a particular business object. Note: By default, the parameter value is set to C1-FTDIST .
Status	Yes	Used when you want to distribute the credit or debit amount on the financial transaction distribution objects which are in a particular status. Note: By default, the parameter value is set to VALIDATED .
Next Status	Yes	Used to indicate the next status to which the financial transaction distribution objects must be transitioned. Note: By default, the parameter value is set to APPLYRULE .
Thread Pool Name	No	Used to specify the thread pool on which you want to execute the batch.

Note: If the **Financial Transaction Distribution (C1-FTDTS)** batch fails or aborts due to some reason, you can restart the batch over and over again with the same set of parameters.

Post Execution Check/Clean Up:

On successful completion of this batch, two adjustments are created when the credit or debit amount on the financial transaction distribution object is distributed against a bill segment or an adjustment. One adjustment is created against the source contract and another adjustment is created against the bill segment or adjustment's contract to which the amount is distributed.

If the credit or debit amount on the financial transaction distribution object is fully distributed, the status of the financial transaction distribution object is changed to **Complete**. However, if the credit or debit amount on the financial transaction distribution object is not yet fully distributed, the status of the financial transaction distribution object is changed to **Validated**.

13. Information Lifecycle Management (ILM)

This section provides detail information about the following batches:

- ILM Crawler - Adjustments (C1-ADCRL)
- ILM Crawler - Bills and Bill Segments (C1-BLCRL)
- ILM Crawler - Billable Charges (C1-BCCRL)
- ILM Crawler - Transaction Rating Calculation Records (C1-TCCRL)
- ILM Crawler - Transaction Feed Management (C1-FMCRL)

13.1 ILM Crawler - Adjustments (C1-ADCRL)

The **ILM Crawler - Adjustments (C1-ADCRL)** batch is used to identify and mark the adjustments which are eligible for archival. It treats the system date minus retention period as the cutoff date. It uses the retention period which is defined for the **ADJUSTMENT** maintenance object. If the retention period is not defined for the **ADJUSTMENT** maintenance object, it uses the default retention period defined in the ILM master configuration.

It considers those adjustments where the **ILM Archival** flag is set to **N** and ILM date is earlier than or equal to the cutoff date, and then executes the ILM eligibility algorithm for each such record. The ILM eligibility algorithm checks whether:

- The adjustment creation date is earlier than or equal to the cutoff date.
- The adjustment is in the **Frozen** status.
- The adjustment is not linked to a pay tender or statement.
- All financial transactions related to the adjustment are in the **Frozen** status.
- The GL Distribution Status of the adjustment financial transaction is set to **D** (i.e. Distributed).
- The Foreign Exchange Gain Loss Calculation Status of the adjustment financial transaction is not set to **N** or **NULL**.
- If the account for which the adjustment is created belongs a customer class where the **Open Item Accounting** option is selected, the adjustment financial transaction is linked to a match event which is in the **Balanced** status.

If any of the above conditions fail, the adjustment is not eligible for archival. It also checks whether the bill on which the adjustment is swept has ILM date earlier than the adjustment's ILM date. If so, the adjustment's ILM date is set to the Bill's ILM date. In addition, it checks whether the adjustment is associated to any transfer adjustment. If the adjustment is associated to a transfer adjustment, it also verifies whether the transfer adjustment is eligible for archival. If so, it marks both the adjustments as eligible for archival. However, if the transfer adjustment is not yet eligible for archival, it moves both the adjustments on the same table partition and marks them as not eligible for archival.

Note: The system enables you to add additional conditions which should be considered while verifying whether the records are eligible for archival. You can define a custom algorithm and attach it to the **ILM Eligibility** system event of the **ADJUSTMENT** maintenance object.

This batch is a multi-threaded batch. The multi-threading is based on adjustment ID and chunks for multi-threading are created based on numerical distribution of adjustment ID.

You can specify the following parameters while executing this batch:

Parameter Name	Mandatory (Yes or No)	Description
Maintenance Object	Yes	Used to indicate that you want to verify the adjustments which are created using the business objects of a particular maintenance object. Note: By default, the parameter value is set to ADJUSTMENT .
Override Cutoff Date	No	Used when you want to override the calculated cutoff date (i.e. system date minus retention period). Note: You must specify a date which is earlier than or equal to the calculated cutoff date. You must specify the date in the YYYY-MM-DD format.
Thread Pool Name	No	Used to specify the thread pool on which you want to execute the batch.

Note: If the **ILM Crawler - Adjustments (C1-ADCRL)** batch fails or aborts due to some reason, you can restart the batch over and over again with the same set of parameters.

Post Execution Check/Clean Up:

On successful completion of this batch, the **ILM_ARCH_SW** column corresponding to the adjustments, which are eligible for archival, in the **CI_ADJ** table is set to **Y**.

13.2 ILM Crawler - Bills and Bill Segments (C1-BLCRL)

The **ILM Crawler - Bills and Bill Segments (C1-BLCRL)** batch is used to identify and mark the bills and their bill segments which are eligible for archival. It treats the system date minus retention period as the cutoff date. It uses the retention period which is defined for the **BILL** maintenance object. If the retention period is not defined for the **BILL** maintenance object, it uses the default retention period defined in the ILM master configuration.

It considers those bills where the **ILM Archival** flag is set to **N** and ILM date is earlier than or equal to the cutoff date, and then executes the ILM eligibility algorithm for each such bill. The ILM eligibility algorithm checks whether:

- The bill date is earlier than or equal to the cutoff date.
- The bill is in the **Complete** status.
- The bill is not a credit note bill or is not linked to a credit note bill.
- The bill is not linked to a statement.

- All financial transactions related to the bill and its bill segments are in the **Frozen** status.
- The GL Distribution Status of the bill and its bill segment financial transactions is set to **D** (i.e. Distributed)
- The Foreign Exchange Gain Loss Calculation Status of the bill and its bill segment financial transactions is not set to **N** or **NULL**.
- If the account for which the bill is created belongs a customer class where the **Open Item Accounting** option is selected, the bill and its bill segment financial transactions are linked to match events which are in the **Balanced** status.
- The Bill ID, Parent ID, and the Presentment Bill ID on the bill financial transaction are same.

If any of the above conditions fail, the bill and its bill segments are not eligible for archival. It also checks whether the end date of any billable charge which is swept on the bill is later than the cutoff date. If so, the bill's and bill segment's ILM date is set to the billable charge's ILM date.

Note: The system enables you to add additional conditions which should be considered while verifying whether the records are eligible for archival. You can define a custom algorithm and attach it to the **ILM Eligibility** system event of the **BILL** maintenance object.

This batch is a multi-threaded batch. The multi-threading is based on bill ID and chunks for multi-threading are created based on numerical distribution of bill ID.

You can specify the following parameters while executing this batch:

Parameter Name	Mandatory (Yes or No)	Description
Maintenance Object	Yes	Used to indicate that you want to verify the bills which are created using the business objects of a particular maintenance object. Note: By default, the parameter value is set to BILL .
Override Cutoff Date	No	Used when you want to override the calculated cutoff date (i.e. system date minus retention period). Note: You must specify a date which is earlier than or equal to the calculated cutoff date. You must specify the date in the YYYY-MM-DD format.
Thread Pool Name	No	Used to specify the thread pool on which you want to execute the batch.

Note: If the **ILM Crawler - Bills and Bill Segments (C1-BLCRL)** batch fails or aborts due to some reason, you can restart the batch over and over again with the same set of parameters.

Post Execution Check/Clean Up:

On successful completion of this batch, the **ILM_ARCH_SW** column corresponding to the bills and their bill segments, which are eligible for archival, in the **CI_BILL** and **CI_BSEG** tables, respectively, is set to **Y**.

13.3 ILM Crawler - Billable Charges (C1-BCCRL)

The **ILM Crawler - Billable Charges (C1-BCCRL)** batch is used to identify and mark the billable charges which are eligible for archival. It treats the system date minus retention period as the cutoff date. It uses the retention period which is defined for the **BILL CHARGE** maintenance object. If the retention period is not defined for the **BILL CHARGE** maintenance object, it uses the default retention period defined in the ILM master configuration.

It considers those billable charges where the **ILM Archival** flag is set to **N** and ILM date is earlier than or equal to the cutoff date, and then executes the ILM eligibility algorithm for each such billable charge. The ILM eligibility algorithm checks whether:

- The billable charge end date is earlier than or equal to the cutoff date.
- The billable charge is in the **Canceled** status and the **Recurring** flag is set to **NULL**.
- If the billable charge is in the **Billable** status and the **Recurring** flag is set to **NULL**, there should be bill segments and financial transactions associated with the billable charge.
- The status of the corresponding bill segment is not **Error**.
- All financial transactions related to the corresponding bill segment are in the **Frozen** status.
- The GL Distribution Status of the corresponding bill segment financial transactions is set to **D** (i.e. Distributed)
- The Foreign Exchange Gain Loss Calculation Status of the corresponding bill segment financial transactions is not set to **N** or **NULL**.
- If the account for which the bill is created belongs a customer class where the **Open Item Accounting** option is selected, the corresponding bill segment financial transactions are linked to match events which are in the **Balanced** status.
- The Bill ID, Parent ID, and the Presentment Bill ID on the corresponding bill segment financial transaction are same.

If any of the above conditions fail, the billable charge is not eligible for archival. It also checks whether the billable charge end date is later than the cutoff date. If so, the billable charge's ILM date is set to the billable charge's end date.

Note: The system enables you to add additional conditions which should be considered while verifying whether the records are eligible for archival. You can define a custom algorithm and attach it to the **ILM Eligibility** system event of the **BILL CHARGE** maintenance object.

This batch is a multi-threaded batch. The multi-threading is based on billable charge ID and chunks for multi-threading are created based on numerical distribution of billable charge ID.

You can specify the following parameters while executing this batch:

Parameter Name	Mandatory (Yes or No)	Description
Maintenance Object	Yes	Used to indicate that you want to verify the billable charges which are created using the business objects of a particular maintenance object. Note: By default, the parameter value is set to BILL CHARGE .
Override Cutoff Date	No	Used when you want to override the calculated cutoff date (i.e. system date minus retention period). Note: You must specify a date which is earlier than or equal to the calculated cutoff date. You must specify the date in the YYYY-MM-DD format.
Thread Pool Name	No	Used to specify the thread pool on which you want to execute the batch.

Note: If the **ILM Crawler - Billable Charges (C1-BCCRL)** batch fails or aborts due to some reason, you can restart the batch over and over again with the same set of parameters.

Post Execution Check/Clean Up:

On successful completion of this batch, the **ILM_ARCH_SW** column corresponding to the billable charges, which are eligible for archival, in the **CI_BILL_CHG** table is set to **Y**.

13.4 ILM Crawler - Transaction Feed Management (C1-FMCRL)

The **ILM Crawler - Transaction Feed Management (C1-FMCRL)** batch is used to identify and mark the transactions which are eligible for archival. It treats the system date minus retention period as the cutoff date. It uses the retention period which is defined for the **C1-TFMTXNDTL** maintenance object. If the retention period is not defined for the **C1-TFMTXNDTL** maintenance object, it uses the default retention period defined in the ILM master configuration.

It considers those transactions where the **ILM Archival** flag is set to **N** and the current system processing date (**CURR_SYS_PRCS_DT**) is earlier than or equal to the cutoff date, and then executes the ILM eligibility algorithm for each such transaction. The ILM eligibility algorithm checks whether:

- The status of the transaction is **Invalid**, **Ignored**, or **Cancelled**.
- The status of the transaction is **Compete** and its transaction legs are either in the **Ignore** status or belong to a frozen bill segment.

- The status of the transaction and its transaction legs is **Error**.

If the above conditions fail, the transactions are not eligible for archival.

Note: The system enables you to add additional conditions which should be considered while verifying whether the records are eligible for archival. You can define a custom algorithm and attach it to the **ILM Eligibility** system event of the **C1-TFMTXNDTL** maintenance object.

This batch is a multi-threaded batch. The multi-threading is based on transaction ID and chunks for multi-threading are created based on numerical distribution of transaction ID.

You can specify the following parameters while executing this batch:

Parameter Name	Mandatory (Yes or No)	Description
Maintenance Object	Yes	Used to indicate that you want to verify the transactions which are created using the business objects of a particular maintenance object. Note: By default, the parameter value is set to C1-TFMTXNDTL .
Override Cutoff Date	No	Used when you want to override the calculated cutoff date (i.e. system date minus retention period). Note: You must specify a date which is earlier than or equal to the calculated cutoff date. You must specify the date in the YYYY-MM-DD format.
Chunk Size	Yes	Used to specify the number of transactions you want to verify in each work unit. Note: By default, the parameter value is set to 500 .
Thread Pool Name	No	Used to specify the thread pool on which you want to execute the batch.

Note: If the **ILM Crawler - Transaction Feed Management (C1-FMCRL)** batch fails or aborts due to some reason, you can restart the batch over and over again with the same set of parameters.

Post Execution Check/Clean Up:

On successful completion of this batch, the **ILM_ARCH_SW** column corresponding to the transactions, which are eligible for archival, in the **CI_TXN_DETAIL** table is set to **Y**.

13.5 ILM Crawler - Transaction Rating Calculation Records (C1-TCCRL)

The **ILM Crawler - Transaction Rating Calculation Records (C1-TCCRL)** batch is used to identify and mark the transaction leg calculation records which are eligible for archival. It treats the system date minus retention period as the cutoff date. It uses the retention period which is defined for the **C1-TXNCALC** maintenance object. If the retention period is not defined for the **C1-TXNCALC** maintenance object, it uses the default retention period defined in the ILM master configuration.

It considers those transaction leg calculation records where the **ILM Archival** flag is set to **N** and ILM date is earlier than or equal to the cutoff date, and then executes the ILM eligibility algorithm for each such transaction leg calculation record. The ILM eligibility algorithm checks whether:

- The **ILM Archival** flag of the corresponding transaction leg is set to **Y**.

If the above condition fails, the transaction leg calculation records are not eligible for archival.

Note: The system enables you to add additional conditions which should be considered while verifying whether the records are eligible for archival. You can define a custom algorithm and attach it to the **ILM Eligibility** system event of the **C1-TXNCALC** maintenance object.

This batch is a multi-threaded batch. The multi-threading is based on transaction calculation ID and chunks for multi-threading are created based on numerical distribution of transaction calculation ID.

You can specify the following parameters while executing this batch:

Parameter Name	Mandatory (Yes or No)	Description
Maintenance Object	Yes	Used to indicate that you want to verify the transaction leg calculation records which are created using the business objects of a particular maintenance object. Note: By default, the parameter value is set to C1-TXNCALC .
Override Cutoff Date	No	Used when you want to override the calculated cutoff date (i.e. system date minus retention period). Note: You must specify a date which is earlier than or equal to the calculated cutoff date. You must specify the date in the YYYY-MM-DD format.

Parameter Name	Mandatory (Yes or No)	Description
Chunk Size	Yes	Used to specify the number of transaction leg calculation records you want to verify in each work unit. <div>Note: By default, the parameter value is set to 500.</div>
Thread Pool Name	No	Used to specify the thread pool on which you want to execute the batch.

Note: If the **ILM Crawler - Transaction Rating Calculation Records (C1-TCCRL)** batch fails or aborts due to some reason, you can restart the batch over and over again with the same set of parameters.

Post Execution Check/Clean Up:

On successful completion of this batch, the **ILM_ARCH_SW** column corresponding to the transaction leg calculation records, which are eligible for archival, in the **CI_TXN_CALC** table is set to **Y**.

14. Request

This section provides detail information about the **Generic Request Periodic Monitor (C1-GENRQ)** batch.

14.1 Generic Request Periodic Monitor (C1-GENRQ)

The **Generic Request Periodic Monitor (C1-GENRQ)** batch is used to monitor or check whether there are any requests in the status where this batch is configured for monitoring the process. If so, the batch transitions the requests to the default next status specified in the lifecycle and executes the algorithm (if any) attached to the default next status.

This batch is a multi-threaded batch. The multi-threading is based on request ID and chunks for multi-threading are created based on numerical distribution of request ID. However, we recommend you to execute this batch in the single-threaded mode. Otherwise, erroneous results might occur. While executing this batch in the single-threaded mode, the following fields must be set to zero:

- Thread Count
- Override Nbr Records to Commit

You can specify the following parameters while executing this batch:

Parameter Name	Mandatory (Yes or No)	Description
Maintenance Object	Yes	Used to indicate that you want to monitor requests which are created using the business objects of a particular maintenance object. Note: By default, the parameter value is set to C1-REQUEST .
Batch Control	Yes	Used when you want to monitor requests whose current status is linked to the Generic Request Periodic Monitor (C1-GENRQ) batch. The valid value is true .
Request Type	No	Used when you want to monitor requests which are created using a particular request type.
Business Object	No	Used when you want to monitor requests which are created using a particular business object.
Status	No	Used when you want to monitor requests 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 Number of Errors	No	Used to override the maximum number of errors after which the batch must be terminated.

Parameter Name	Mandatory (Yes or No)	Description
Thread Pool Name	No	Used to specify the thread pool on which you want to execute the batch.

Note: If the **Generic Request Periodic Monitor (C1-GENRQ)** batch fails or aborts due to some reason, you can restart the batch over and over again with the same set of parameters.

Post Execution Check/Clean Up:

On successful completion of this batch, the status of the request is changed to the default next status specified in the lifecycle of the business object. In addition, the algorithms (if any) attached to the default next status are executed in the specified sequence.