Oracle® Financial Services Funds Transfer Pricing Cloud Service Break Identification Process

User Guide

Release 23.12.01
F93255-01
February 2024

ORACLE

Oracle Financial Services Funds Transfer Pricing Cloud Service Break Identification Process User Guide, Release 23.12.01

F93255-01

Contents

1

Break Identification Processes

Breaks are associated with Assets and Liabilities that have fixed maturities and have experienced a full prepayment or pre-closure, partial prepayment, or restructuring. Any event that causes the bank to receive a change to scheduled contractual cash flows on a fixed maturity instrument results in a Break Funding Event and should be evaluated. Transactions that could cause a change in future cash flows would include full loan prepayments, partial loan prepayments, early withdrawal of term deposits, or a change in maturity tenor, payment amount, payment frequency, or other contractual terms.

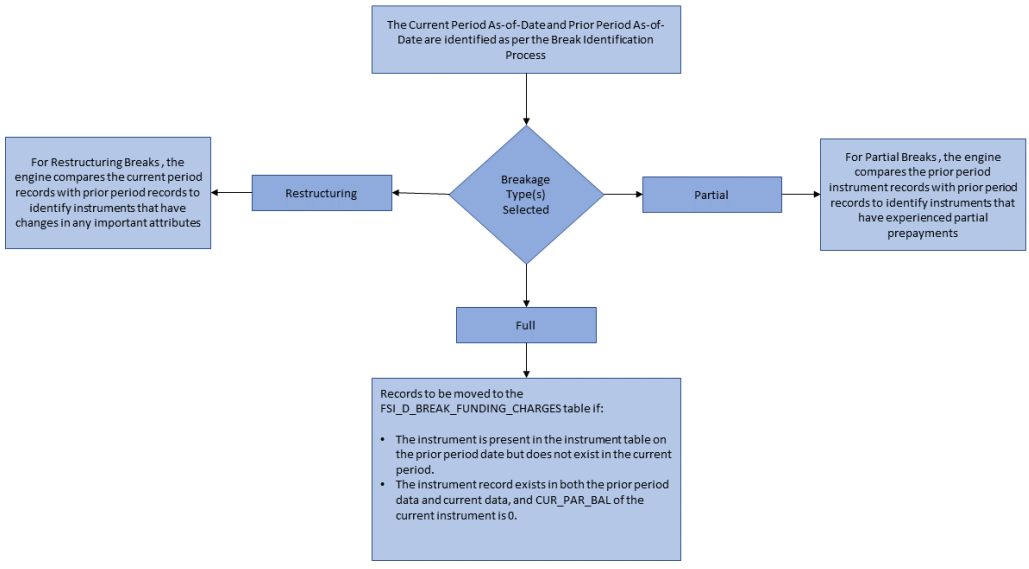
The Break Identification Process allows you to perform the following tasks:

- Determine the data that you want to process (Product Selection block).
- Specify the parameters for the process. The parameters include break types like a full break, partial break, and change in attributes.
- Execute or Run the Break Identification Request and generate results (Break Identification Process Summary Page).

Break Identification processing should be run if automatic break detection is the preferred approach to populating the break events table (FSI_D_BREAK_FUNDING_CHARGES). The Break Funding Charges table is the source table for calculating breakage charges.

The following figure shows the overview of the Break Identification process:

Figure 1-1 Break Identification Process Flow



1.1 Full Breaks

The following accounts are considered as full breaks, fully repaid, or terminated:

- The Instrument is present in the instrument table on the prior period date but does not exist in the current period and the maturity date of the prior period record is greater than the current period as of the date.
- The Instrument record exists in both the prior period data and current data and the CUR_PAR_BAL of the current instrument is 0 and the account open flag is NO.

For the above scenarios, the Break Identification Process populates the source and Break Funding Charges instrument tables with the following:

- BREAKAGE_FLG = 2 (*Source record only) (External break will populate flag=1)
- BREAKAGE_TYPE_CD = 1
- BREAKAGE_AMOUNT = CUR_PAR_BAL (prior period)

Additionally, the FSI_D_BREAK_FUNDING_CHARGES table is populated as follows:

- • If Prior record NEXT_PRIN_PAYMENT_DATE > Current AS_OF_DATE then:
  - CUR_PAR_BAL = Prior Period CUR_PAR_BAL
  - CUR_BOOK_BAL = Prior Period CUR_BOOK_BAL
- If Prior record NEXT_PRIN_PAYMENT_DATE <= Current AS_OF_DATE then:
  - IF AFTER_PAYMENT_BALANCE is Not Null
    CUR_PAR_BAL = AFTER_PAYMENT_BALANCE

    CUR_BOOK_BAL = AFTER_PAYMENT_BALANCE
  - IF AFTER_PAYMENT_BALANCE is Null
    CUR_PAR_BAL = Prior Period CUR_PAR_BAL

    CUR_BOOK_BAL = Prior Period CUR_BOOK_BAL
- AS_OF_DATE = AS_OF_DATE defined in Application Preferences
- BREAKAGE_AMOUNT = Prior Period CUR_PAR_BAL
- BREAKAGE_TYPE_CD = 1
- All additional fields are carried forward from the prior period record.

For Full Breaks, the Next Principle/Interest Payment Date gets rolled forward till the Maturity Date. Break record has AS_OF_DATE = NEXT_PRIN_PAYMENT_DATE.

If NEXT_PRIN_PAYMENT_DATE <= AS_OF_DATE, then Next Principle Payment Date = Next Principle Payment Date + Payment Frequency/Multiplier. This should not exceed the Maturity Date.

If NEXT_PRIN_PAYMENT_DATE <= AS_OF_DATE for the current period, then Break Amount = AFTER_PAYMENT_BAL from the prior record. This is used when you do not want to pay a break charge against the current period's scheduled principal payment.

1.2 Partial Breaks

For partial breaks, the engine compares the prior period-instrument records with current period records to identify instruments that have experienced a partial prepayment.

The Break Identification Process uses Identity Code to determine which record is current and which is prior. It is assumed the Prior record Identity Code will be < Current Period.

The following accounts are considered as partial breaks:

Case 1:

If the NEXT_PRIN_PAYMENT_DATE on the prior period record is less than or equal to the AS_OF_DATE of the current period record, then:

- To detect a partial break, compare the AFTER_PAYMENT_BALANCE of the prior period record with the CUR_PAR_BAL of the current period record. If the difference is more than the MINIMUM_BREAK_AMOUNT then the instrument is classified as a Partial Break. For this scenario, the Break Identification Process populates the source and Break Funding Charges instrument tables with the following:
  - BREAKAGE_FLG = 2 (*Source - Current record only) (External break will populate flag=1)
  - BREAKAGE_TYPE_CD = 2
  - BREAKAGE_AMOUNT = AFTER_PAYMENT_BALANCE of prior record – CUR_PAR_BAL of the current record Additionally, the FSI_D_BREAK_FUNDING_CHARGES table is populated as follows:
  - Both the prior period record and the current record are copied into the FSI_D_BREAK_FUNDING_CHARGES table. The as of date for the prior record is changed to the current as of date.
  - The BREAKAGE_AMOUNT of the prior record =0
  - The BREAKAGE_AMOUNT of the current record = AFTER_PAYMENT_BALANCE of prior record – CUR_PAR_BAL of the current record
  - BREAKAGE_TYPE_CD = 2
  - Specific Fields updated on prior break funding record:
    * Next Interest Payment Date: If Next Interest Payment Date <= AS_OF_DATE then Next Interest Payment Date +Payment Frequency/Multiplier
    * Last Interest Payment Date: If Next Interest Payment Date <= AS_OF_DATE then Next Interest Payment Date
    * Next Principle Payment Date: If Next Principle Payment Date <= AS_OF_DATE then Next Principle Payment Date +Payment Frequency/Multiplier
    * Last Principle Payment Date: If Next Principle Payment Date <= AS_OF_DATE then Next Principle Payment Date
    * Next Reprice Date: If Next Reprice Date <= AS_OF_DATE then Next Reprice Date + Reprice Frequency/Multiplier
    * Last Reprice Date: If Next Reprice Date <= AS_OF_DATE then Next Reprice Date

    - Remaining Number of Payments: If Next Principle Payment Date <= AS_OF_DATE then Remaining Number of Payments -1
  - All additional fields are carried forward from the prior period record.

Case 2:

If the NEXT_PRIN_PAYMENT_DATE on the prior period record is greater than the As-of-Date of the current period, then:

- To detect a partial break, compare the CUR_PAR_BAL of the prior period record with the CUR_PAR_BAL of the current record. If the difference is more than the Minimum Break Amount, then the instrument is classified as a partial break.
  For this scenario, the Break Identification Process populates the source and Break Funding Charges instrument tables with the following:
  - BREAKAGE_FLG = 2 (*Source - Current Record only)
  - BREAKAGE_TYPE_CD = 2
    - If both Partial and Change in Attributes are detected, then BREAKAGE_TYPE_CD = 5
  - BREAKAGE_AMOUNT = CUR_PAR_BAL of the prior period record – CUR_PAR_BAL of the current record

  Additionally, the FSI_D_BREAK_FUNDING_CHARGES table is populated as follows:
  - Both the prior period record and the current record are copied into the FSI_D_BREAK_FUNDING_CHARGES table. The as of date for the prior record is changed to the current As-of-Date.
  - The BREAKAGE_AMOUNT of the Prior Record =0
  - The BREAKAGE_AMOUNT of the Current Record = CUR_PAR_BAL of prior record – CUR_PAR_BAL of the Current Record
  - BREAKAGE_TYPE_CD = 2

    Note:

    If it is NULL, 0, or 1, the breakage charge calculation will treat as a full break and will not correctly calculate BREAK_FUNDING_AMT_CHG.
  - RECORD_IND = -1 for Prior Record and 1 for Current Record.
  - All additional fields are carried forward from the Prior Period Record.

    Note:

    Partial Breaks are also detected when there is an increase in balance.

  Example: Partial Break:
  - Prior Record:
    - As-of-Date = 30-SEP-2012

    - Next_Reprice_Date = 01-OCT-2012
    - Last_Reprice_Date = 01-SEP-2012
    - REPRICE_FREQ = 1M
  - Current Record:
    - As-of-Date = 01-OCT-2012
    - Next_Reprice_Date = 01-NOV-2012
    - Last_Reprice_Date = 01-OCT-2012
    - REPRICE_FREQ = 1M

  The break occurs on 01-OCT-2012 and a break is detected on this date. This is also the Application Preferences As-of-Date given by the user.

  Records moved to FSI_D_BREAK_FUNDING_CHARGES will be:

  - Prior Record:
    - Since Next_Reprice_Date (1-Oct-2012) <= active As-of-Date (1-Oct-2012),
    - As_of_Date = 01-OCT-2012
    - Next_Reprice_Date = 01-NOV-2012
    - Last_Reprice_Date = 01-OCT-2012
  - Current Record:
    - As_of_Date = 01-OCT-2012
    - Next_Reprice_Date = 01-NOV-2012
    - Last_Reprice_Date = 01-OCT-2012
    - The engine can only process the modified record in FSI_D_BREAK_FUNDING_CHARGES when Last_Reprice_Date <= As_of_Date < Next_Reprice_Date.

1.3 Change in Attributes Break

For Change in Attributes Break, the engine compares the current period records with the prior period records to determine if any changes are made to critical attributes (other than balances). If any of the critical attributes have changed then it qualifies as a break event.

The Break Identification Process uses Identity Code to determine which record is current and which is prior. It is assumed the Prior record Identity Code will be < Current Period.

The following key attributes are compared when analyzing data for Change in Attributes Break:

- ADJUSTABLE_TYPE_CD
- AMRT_TYPE_CD
- COMPOUND_BASIS_CD
- CUR_NET_RATE
- MATURITY_DATE
- NEG_AMRT_AMT
- NEG_AMRT_EQ_DATE

- NEG_AMRT_EQ_FREQ
- NEG_AMRT_EQ_MULT
- NEG_AMRT_LIMIT
- INT_PMT_FREQ
- INT_PMT_FREQ_MULT
- REPRICE_FREQ
- REPRICE_FREQ_MULT
- RESIDUAL_AMOUNT
- ACCRUAL_BASIS_CD
- PRIN_PMT_FREQ
- PRIN_PMT_FREQ_MULT

The list of seeded attributes can be viewed in the FSI_BRK_DETECTION_COLUMN_LIST table. Note that some attributes such as CUR_PAYMENT are not in the seeded list, but are referred to conditionally along with other columns.

If Prior Record Adjustable Type Cd = 0 and Current Record Adjustable Type Cd = 0 and Prior Cur Payment <> Current Cur Payment, then consider a break event.

If you want to consider any change in current payment as a break event, regardless of adjustable type code, then Cur Payment can be appended to the list of seeded values.

If any of these fields are identified as changed while comparing current and prior records, the records are flagged as a break.

For this scenario, the Break Identification Process populates the source and Break Funding Charges instrument tables with the following:

- BREAKAGE_FLG = 2 (Source – Current record only)
- BREAKAGE_TYPE_CD = 3
  - If both Partial and Change in Attributes are detected, then BREAKAGE_TYPE_CD = 5
    - BREAKAGE_AMOUNT = CUR_PAR_BAL

Additionally, the FSI_D_BREAK_FUNDING_CHARGES table is populated as follows:

- Both the prior period record and the current record are copied into the FSI_D_BREAK_FUNDING_CHARGES table. The As of Date for the prior record is changed to the current as of date.
- BREAKAGE_TYPE_CD = 3

> **Note:**
>
> If it is NULL, 0, or 1, the breakage charge calculation will treat as a full break and will not correctly calculate BREAK_FUNDING_AMT_CHG.

- RECORD_IND = -1 for Prior Record and 1 for Current Record

- All additional fields are carried forward from the Prior Period Record If prior record's Next_Reprice_Date <= active As_of_Date,
- Next Interest Payment Date: If Next Interest Payment Date <= AS_OF_DATE then Next Interest Payment Date +Payment Frequency/Multiplier
- Last Interest Payment Date: If Next Interest Payment Date <= AS_OF_DATE then Next Interest Payment Date
- Next Principle Payment Date: If Next Principle Payment Date <= AS_OF_DATE then Next Principle Payment Date +Payment Frequency/Multiplier
- Last Principle Payment Date: If Next Principle Payment Date <= AS_OF_DATE then Next Principle Payment Date
- Next Reprice Date: If Next Reprice Date <= AS_OF_DATE then Next Reprice Date + Reprice Frequency/Multiplier
- Last Reprice Date: If Next Reprice Date <= AS_OF_DATE then Next Reprice Date
- Remaining Number of Payments: If Next Principle Payment Date <= AS_OF_DATE then Remaining Number of Payments -1

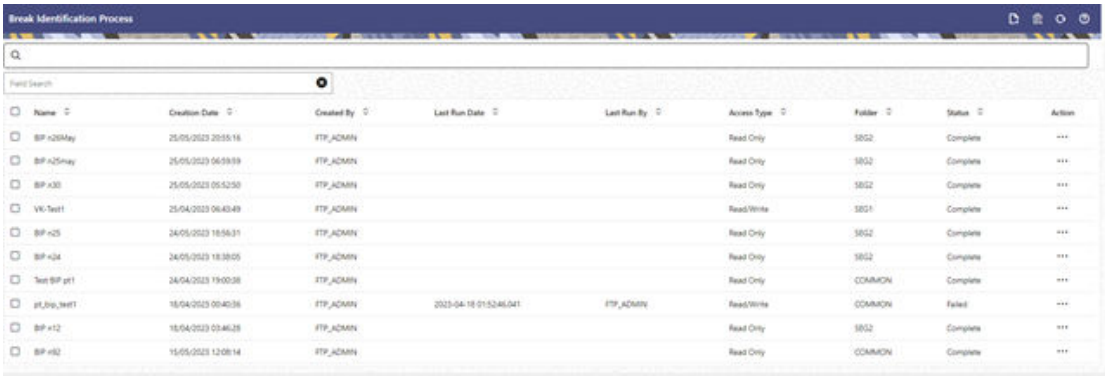
1.4 Summary and Detail Screens

To open the Break Identification Process summary page, select **Operations and Processes** and select **Break Identification Process**.

The Break Identification Process summary page is displayed showing a set of Break Identification Process definitions.

Using the search criteria, you can control set of definitions displayed. When you Add, Edit, or View a definition, the application displays a detailed screen.

Figure 1-2 Break Identification Process Summary page



Navigation in Summary Screen

When you navigate to the Break Identification Process summary screen, the existing definitions are presented in a summary table. The Break Identification Process summary screen has two panes: Search and Break Identification Process summary table.

The title bar of the summary page provides several actions for the user. They are:

- **Add**: Click Add icon to build a new Break Identification Process definition. The Add icon is disabled if any rows in the table are selected.
- **Multiple Delete**: Select one or more drivers in the table and then click the (-) icon at the top right of the summary page to delete more than one rule at the same time.
- **Refresh**: Click Refresh to refresh the summary page.
- **Help**: Click Help icon to view the Break Identification Process help.

Search

There are two Search options provided to search the Break Identification Process definitions on the Summary Page.

To search the Break Identification Process definitions:

1. Click the **Search** icon on the Search pane to collapse (display) the Criteria window.
2. Enter the definition **Name** or **Description**.
3. Click **Cancel** to remove the filter criteria on the Search window and refresh the window.
4. Click **Search** after entering the search criteria.
   The search results are displayed in a table containing all the Break Idenftiication Process definitions that meet the search criteria.
5. The other method to search is using the **Field Search** option. The Field Search is an inline wildcard search that allows you to enter value partially or fully and the rows that match the entered string in any of its column is fetched in the Summary table.

Summary Table

This section of the Breal Identification Process summary screen presents a table containing all of the already created Break Identification Process definitions.

The Summary Table displays the following details:

- **Name**: Displays the given name for the Break Identification Process definition.
- **Creation Date**: Displays the date and time at which a Break Identification Process definition was created.
- **Created By**: Displays the name of the user who created the Break Identification Process definition.
- **Last Run Date**: Displays the recent date on which the Break Identification Process definition was run.
- **Last Run By**: Displays the name of the user who ran the Break Identification Process definition.
- **Access Type**: Displays the “Read/Write” or “Read Only” property of a Break Identification Process definition. Only the creator of a rule may change its Access Type.
- **Folder**: Displays the folder in which the definition is created.

- **Status**: Before executing a Break Identification Process definition for the first time, the Status is blank. After executing a driver rule the appropriate status of the rule is displayed among In Progress, Complete, Success, or Failed.
- **Action**: Displays the following list of actions that can be performed on the rule.
  - **View**: Click the View icon to view the contents of a Break Identification Process definition on a read-only basis.
  - **Edit**: Click the Edit icon to modify a previously saved definition.
  - **Delete**: Click Delete to delete the Break Identification Process definition you have selected.
  - **Save As**: Click on this option to create a copy of an existing Break Identification Process definition. The Save As pop-up window allows you to enter the Name, Description, Folder, and Access Type for the copy definition.
  - **Run**: To submit the definition for processing.
  - **Execution Logs**: To see the execution log details of the selected definition.
  - **Check Dependency**: To check the dependency of the selected definition on other processes.

1.4.1 Navigation in Summary Screen

When you navigate to the Break Identification Process summary screen, the existing definitions are presented in a summary table. The Break Identification Process summary screen has two panes: Search and Break Identification Process summary table.

The title bar of the summary page provides several actions for the user. They are:

- **Add**: Click Add icon to build a new Break Identification Process definition. The Add icon is disabled if any rows in the table are selected.
- **Multiple Delete**: Select one or more drivers in the table and then click the (-) icon at the top right of the summary page to delete more than one rule at the same time.
- **Refresh**: Click Refresh to refresh the summary page.
- **Help**: Click Help icon to view the Break Identification Process help.

1.4.1.1 Search

There are two Search options provided to search the Break Identification Process definitions on the Summary Page.

To search the Break Identification Process definitions:

1. Click the **Search** icon on the Search pane to collapse (display) the Criteria window.
2. Enter the definition **Name** or **Description**.
3. Click **Cancel** to remove the filter criteria on the Search window and refresh the window.
4. Click **Search** after entering the search criteria.
   The search results are displayed in a table containing all the Break Idenftiication Process definitions that meet the search criteria.
5. The other method to search is using the **Field Search** option. The Field Search is an inline wildcard search that allows you to enter value partially or fully and the rows that match the entered string in any of its column is fetched in the Summary table.

1.4.1.2 Break Identification Process Summary Table

This section of the Breal Identification Process summary screen presents a table containing all of the already created Break Identification Process definitions.

The Summary Table displays the following details:

- **Name**: Displays the given name for the Break Identification Process definition.
- **Creation Date**: Displays the date and time at which a Break Identification Process definition was created.
- **Created By**: Displays the name of the user who created the Break Identification Process definition.
- **Last Run Date**: Displays the recent date on which the Break Identification Process definition was run.
- **Last Run By**: Displays the name of the user who ran the Break Identification Process definition.
- **Access Type**: Displays the “Read/Write” or “Read Only” property of a Break Identification Process definition. Only the creator of a rule may change its Access Type.
- **Folder**: Displays the folder in which the definition is created.
- **Status**: Before executing a Break Identification Process definition for the first time, the Status is blank. After executing a driver rule the appropriate status of the rule is displayed among In Progress, Complete, Success, or Failed.
- **Action**: Displays the list of actions that can be performed on the rule.

The **Action** column on the Break Identification Process summary page offers the following actions that allow you to perform different functions:

- **View**: Click the View icon to view the contents of a Break Identification Process definition on a read-only basis.
- **Edit**: Click the Edit icon to modify a previously saved definition.
- **Delete**: Click Delete to delete the Break Identification Process definition you have selected.
- **Save As**: Click on this option to create a copy of an existing Break Identification Process definition. The Save As pop-up window allows you to enter the Name, Description, Folder, and Access Type for the copy definition.
- **Run**: To submit the definition for processing.
- **Execution Logs**: To see the execution log details of the selected definition.
- **Check Dependency**: To check the dependency of the selected definition on other processes.

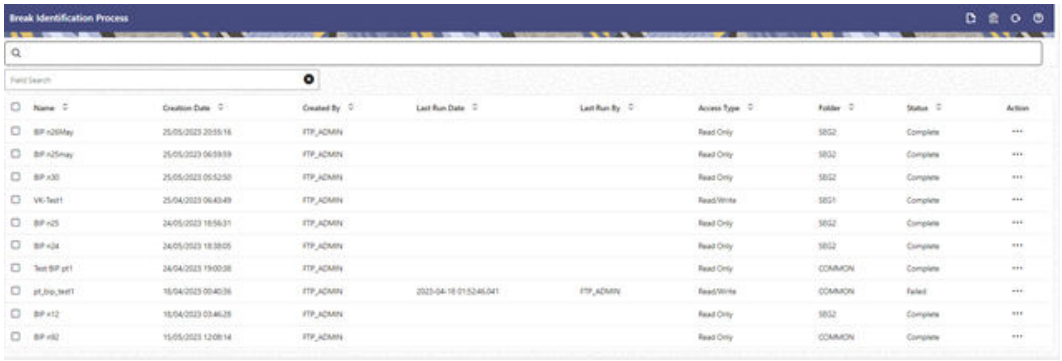
1.5 Create a Break Identification Process

To define and execute a Break Identification Process. The Break Identification Process will compare current period-instrument data with prior period-instrument data to identify break events. When breaks are detected, the related instrument records are copied into the FSI_D_BREAK_FUNDING_CHARGES table. This table then becomes the source table for further FTP Add-on Rate Rule > Breakage Charge calculations.

To create a Break Identification Process:

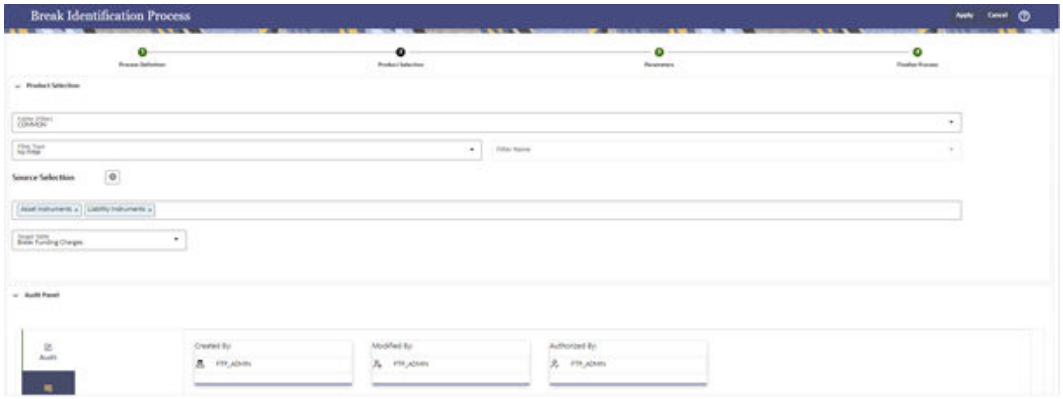
1. From the LHS menu, select **Operations and Processes** and select **Break Identification Process**.
   The Break Identification Process summary page is displayed.

Figure 1-3 Break Identification Process Summary page



2. Select the **Product Selection** block.

Figure 1-4 Break Identification Process Details



3. Enter or select the following details:

Table 1-1 Fields and Descriptions from the Break Identification Process Details page

Term	Description
Folder	The folder where you can save the definition. You can give other users, read/write, or read-only privileges.

Table 1-1 (Cont.) Fields and Descriptions from the Break Identification Process Details page

Term	Description
Filter	Filters allow you to restrict your data selection based on any attribute that exists within an instrument table. You define filters under Common Object Maintenance and reference your filter within the Product Selection block of your Process. The choice of the data filter would determine the instrument records that should be picked up from the As-of-Date and the prior period date for comparative analysis. The supported Filter Types are: • Attribute Filter • Data Filter • Hierarchy Filter • Group Filter **Note:** Data Filters with Expressions are not supported for Break Identification.
Source	Allows you to select one or more source Instrument tables to include in your process. Based on the Instrument Table(s) selected, the instrument records on the As-of-Date and the prior period date are chosen for comparison.
Target Table	Indicates the destination table where break event records will be posted. The default (seeded) table is the FSI_D_BREAK_FUNDING_CHARGES table. Users can additionally register user-defined tables for posting Breakage Funding records if needed.
Parameters	There are three types of break parameters for the accounts: • **Full Break**: Fully repaid or terminated accounts are considered as a Full Break. • **Partial Break**: Partly repaid accounts are considered as Partial Break. • **Change in Attributes**: Here a restructure of the Instrument record happens due to a change in critical attributes or terms other than Balance. You can execute these breaks individually or together.

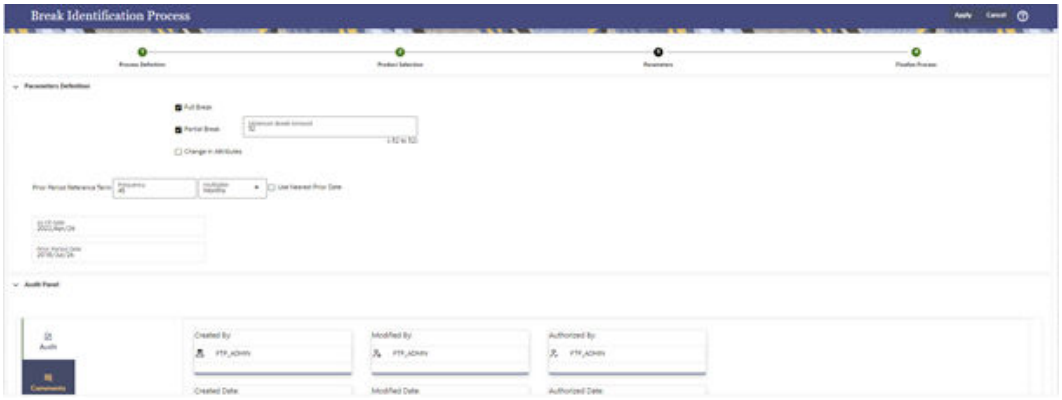
Table 1-1 (Cont.) Fields and Descriptions from the Break Identification Process Details page

Term	Description
Minimum Break Amount	Minimum Break applies to both Positive and Negative breakage amounts. If the user enters the minimum break as 1000, it means that the minimum break amount ranges from -1000 to +1000. If the Breakage Amount that is calculated is less than or equal to the Minimum Break Amount, then it is not passed to the Break Funding Charges table.
Finalize Process	The finalize process screen you to review and finalize the selections made in the Process Definition Flow or to edit the selections.

- Select a Filter (optional) to constrain the data to be included in the process. The supported Filter Types are Attribute Filter, Data Filter, Hierarchy Filter, and Group Filter.
- Select the source table(s) that you want to include in the process.
- Select the target table, which is a Break Funding Instrument table.

4. Select the **Parameters** block.

Figure 1-5 Break Identification Process Parameters page

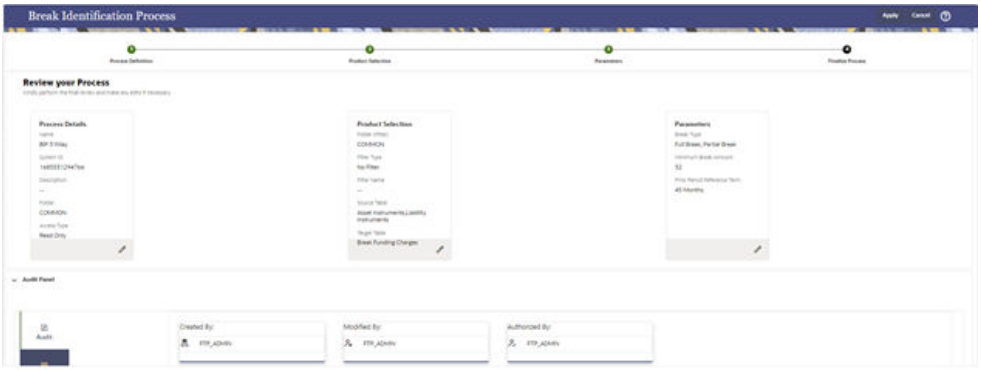


- Select the type of break that you would like to search for and fill in the related details. In the case of a Full Break, the filter will work only on the prior period-instrument record. In the case of a Partial Break or Change in Attributes, the filter will work on both the prior period and current period-instrument records.
- Input the **Minimum Break Amount** as a positive value. The engine will apply the absolute value of the amount of input ranging from - input amount to + input amount. For example, if the input is 100, then break amounts between -100 and +100 will be excluded. This input allows you to filter very small/insignificant break amounts, reducing the amount of data copied into the Break Funding Charges table.
  Note there are two approaches for determining the Prior Period Date. You can input the Prior Period Reference Term and based on the current As of Date, the Prior Period Date will be calculated, or you can select the “Use Nearest Prior Date” option, and the engine will then look back at the historical data (in the table

FSI_PROCESS_RUN_HISTORY) to determine the nearest prior As of Date and will use this as Prior Period Date.

5. Select the **Finalize Process** in the screen.
If you want to edit any of the prior steps, you can directly do it from this screen.

Figure 1-6 Finalize Process screen



6. Select **Apply** to complete the process.

1.6 Executing a Break Identification Process

You execute a Break Identification to compare the current period and prior period data to identify different break types.

To execute a Break Identification Process:

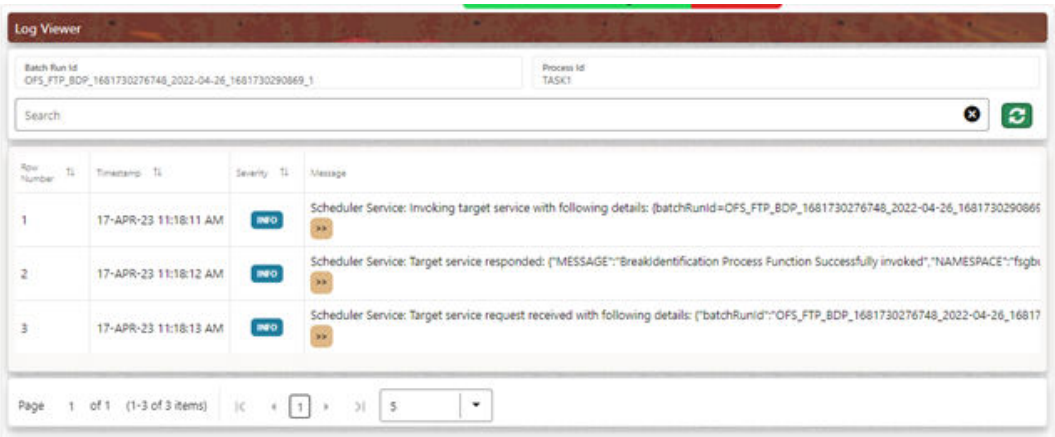
1. Perform the basic steps for Creating a Break Identification Process.
2. There are two approaches to execute the Break Identification Process:
   - Executing from the Summary UI
   - Executing using Batch Framework

Executing from the Summary UI

To execute the Break Identification Process from the Summary UI:

1. Navigate to the Break Identification Process summary Page.
2. Select a **Process** that you want to execute or Run.
The status column indicates whether a process can be Run. There are three possible status conditions:
   - **Failed**: Indicates the process is failed.
   - **Complete**: Indicates the process is fully defined and ready to be Run.
   - **Incomplete**: Indicates the process is partially defined and cannot be Run.
3. After executing the preceding process, select the **View Log** hyperlink.

Figure 1-7 View Log



4. Select the **Task ID** (also known as the Unique System Identifier) to view a report for any processing errors.

> **Note:**
>
> If significant processing errors exist, you must re-Run your process.

The Break Identification process is complete.

Executing using Batch Framework

To execute the Break Identification Process using Batch Framework:

1. Navigate to **Operations** and select **Batch Maintenance**.
2. Create a new batch.
3. Select the **Batch Name** to add the **Task**.
4. Click the **Add** button under the **Task Details** section.
5. Define the **Task ID** and **Description**.
6. Select Components as **Break Identification Process**.
7. Input the following required parameters:
   - Folder
   - Process Name
8. Save the Batch and execute.

1.6.1 Executing from the Summary UI

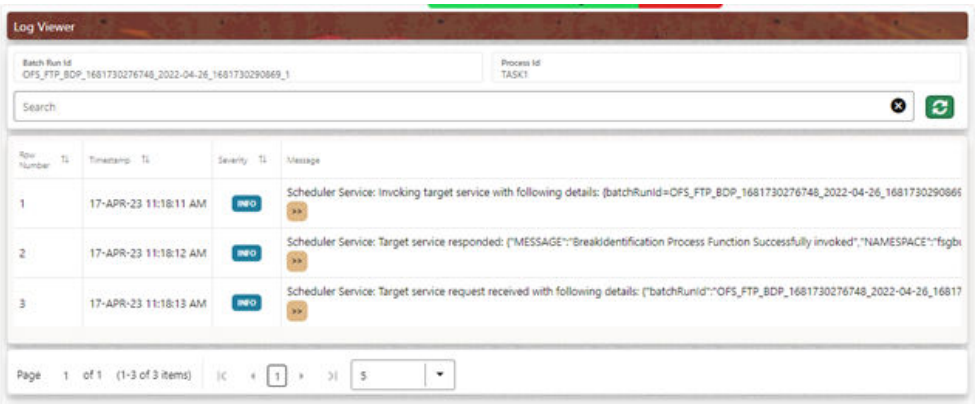
To execute the Break Identification Process from the Summary UI:

1. Navigate to the Break Identification Process summary Page.
2. Select a **Process** that you want to execute or Run.
   The status column indicates whether a process can be Run. There are three possible status conditions:

- **Failed**: Indicates the process is failed.
- **Complete**: Indicates the process is fully defined and ready to be Run.
- **Incomplete**: Indicates the process is partially defined and cannot be Run.

3. After executing the preceding process, select the **View Log** hyperlink.

Figure 1-8 View Log



4. Select the **Task ID** (also known as the Unique System Identifier) to view a report for any processing errors.

> **Note:**
>
> If significant processing errors exist, you must re-Run your process.

The Break Identification process is complete.

1.6.2 Executing using Batch Framework

You can also execute the Break Identification Processes using the Batch Framework.

To execute the Break Identification Process using Batch Framework:

1. Navigate to **Operations** and select **Batch Maintenance**.
2. Create a new batch.
3. Select the **Batch Name** to add the **Task**.
4. Click the **Add** button under the **Task Details** section.
5. Define the **Task ID** and **Description**.
6. Select Components as **Break Identification Process**.
7. Input the following required parameters:
   - Folder
   - Process Name
8. Save the Batch and execute.