# Understanding supply valuation pool

The **supply valuation pool** simplifies the process of MTM (Mark-to-Market) valuation by providing a centralized framework for configuring and managing market indices or curves. MTM ensures that asset and liability values are updated to reflect current market conditions, maintaining the accuracy of financial records.

This module streamlines the management of valuation data, allowing users to efficiently define, organize, and adjust valuation pools. By using this system, businesses can gain actionable insights into market values, enhance operational efficiency, and ensure financial transparency.

## Procedure to configure supply valuation pool

Configuring a supply valuation pool involves accessing the configuration screen, creating categories for better organization, and managing key details such as valuation definitions and pipeline points.

### Prerequisites

* You must have the permissions to *add* or *update* the **Valuation pool configuration** screen.

### Process diagram

Below is a visual representation of the **Valuation pool configuration** process flow:

@startuml  
  
scale max 1000 width  
  
skinparam backgroundColor #F5F5F3  
skinparam Shadowing 10  
skinparam BoundaryBackgroundColor #ffffff  
skinparam BoundaryBorderColor #4ca0d9  
skinparam DiagramBorderColor #F5F5F3  
skinparam BoundaryFontColor #0079c9  
skinparam BoundaryFontStyle bold  
skinparam defaultTextAlignment center  
skinparam SequenceMessageAlign center  
skinparam ReferenceBorderThickness 2  
skinparam padding 2  
skinparam roundCorner 20  
skinparam ActorShadowing false  
skinparam TitleFontColor #000000  
skinparam TitleFontSize 18  
skinparam BoxPadding 10  
  
skinparam default {  
 FontName Inter-SemiBold, SansSerif, sans-serif  
 FontStyle plain  
 FontSize 12  
}  
  
skinparam actorStyle awesome  
  
skinparam sequence {  
  
'Users and boundries (DBS for example)  
 ActorBorderThickness 1  
 ActorBorderColor ##0079c9  
 ActorBackgroundColor #0079c9  
 ActorFontColor #0079c9  
 ActorFontName Roboto, SansSerif, sans-serif  
 ActorFontStyle bold  
  
 AgentBackgroundColor red  
  
'Arrows  
 ArrowFontColor #616161  
 ArrowColor #616161  
 ArrowThickness 2  
  
 RectangleBorderThickness 3  
  
  
'Capability  
 BoxFontColor #000000  
 BoxFontName Roboto, SansSerif, sans-serif  
 BoxFontStyle bold  
 BoxFontSize 14  
  
 BoxBackgroundColor #e9eaeb  
 BoxBorderColor transparent  
  
 BoundaryBackgroundColor #1281ac  
 BoundaryBorderColor #1281ac  
 BoundaryFontColor #1281ac  
 BoundryFontName Roboto, SansSerif, sans-serif  
 BoundaryStereotypeFontColor #616161  
  
 ClassBorderColor #1281ac  
 ComponentBorderColor #1281ac  
  
' Lifelines and dividers/separators  
 DividerBackgroundColor #f5f5f5  
 DividerBorderColor #a2a2a2  
 DividerBorderThickness 1  
 DividerFontColor #616161  
 DividerFontName Roboto, SansSerif, sans-serif  
  
 LifeLineBorderColor #4ca0d9  
  
' Loops but not the title text  
 GroupBorderColor #4CA0D9  
 GroupBorderThickness 1  
 GroupFontName Roboto, SansSerif, sans-serif  
 GroupFontColor #0079c9  
 GroupBackgroundColor #D9EBF6  
 GroupBodyBackGroundColor #FFFFFF  
  
'Title text for loops  
 GroupHeaderFontColor #000000  
 GroupHeaderFontName Roboto, SansSerif, sans-serif  
 GroupHeaderFontStyle bold  
  
'Services  
 ParticipantBorderColor #4ca0d9  
 ParticipantBackgroundColor white  
 ParticipantFontSize 12  
 ParticipantFontStyle bold  
 ParticipantFontName Roboto, SansSerif, sans-serif  
 ParticipantFontColor #0079c9  
 ParticipantBorderThickness 1  
 StereotypeFontColor #616161  
  
 ReferenceBorderThickness 2  
  
  
' SwimlaneBorderThickness 2  
  
 TitleFontColor #1281ac  
 TitleFontName Roboto, SansSerif, sans-serif  
 TitleFontSize 18  
 TitleBorderThickness 5  
  
}  
skinparam interface {  
 BackgroundColor #FFFFFF  
 BorderColor #1281ac  
 BorderThickness 1  
 FontColor #1281ac  
 FontName Roboto, SansSerif, sans-serif  
 FontSize 14  
 FontStyle bold  
  
}  
  
skinparam note {  
  
 BackgroundColor #d9ebf6  
 BorderColor #1281ac  
 BorderThickness 1  
  
 RoundCorner 0  
  
 StartColor #00548c  
 BarColor #00548c  
 EndColor #00548c  
  
 FontColor #000000  
 FontStyle plain  
 FontSize 12  
 FontName Roboto, SansSerif, sans-serif  
  
}  
  
skinparam activity {  
  
 BackgroundColor #FFFFFF  
 DiamondBackgroundColor #FFFFFF  
 DiamondBorderColor #1281ac  
 BorderColor #1281ac  
 BorderThickness 4  
  
 DiamondFontColor #1281ac  
 DiamondFontName Roboto, SansSerif, sans-serif  
 DiamondFontSize 12  
  
 StartColor #1281ac  
 BarColor #1281ac  
 EndColor #1281ac  
  
 FontColor #1281ac  
 FontName Roboto, SansSerif, sans-serif  
 FontSize 14  
 FontStyle bold  
  
}  
  
skinparam rectangle {  
  
 BorderColor #4ca0d9  
 BackgroundColor white  
 FontSize 12  
 FontStyle bold  
 FontName Roboto, SansSerif, sans-serif  
 FontColor #0079c9  
 BorderThickness 1  
  
 StartColor #00548c  
 BarColor #00548c  
 EndColor #00548c  
  
}  
  
skinparam cloud {  
  
 BackgroundColor #FFFFFF  
  
 BorderColor #1281ac  
 BorderThickness 4  
  
 StartColor #1281ac  
 BarColor #1281ac  
 EndColor #1281ac  
  
 FontColor #1281ac  
 FontName Roboto, SansSerif, sans-serif  
 FontSize 14  
 FontStyle bold  
  
}  
  
skinparam class {  
  
 BackgroundColor #FFFFFF  
  
 BorderColor #1281ac  
 BorderThickness 1  
  
 StartColor #1281ac  
 BarColor #1281ac  
 EndColor #1281ac  
  
 FontColor #1281ac  
 FontName Roboto, SansSerif, sans-serif  
 FontSize 14  
 FontStyle bold  
  
  
  
}  
  
'For services  
skinparam component {  
  
 BorderColor #4ca0d9  
 BackgroundColor white  
 StereotypeFontColor #616161  
  
 FontSize 12  
 FontStyle bold  
 FontName Roboto, SansSerif, sans-serif  
 FontColor #0079c9  
 BorderThickness 1  
  
}  
  
  
  
'databases  
skinparam database {  
  
 BackgroundColor #ffffff  
  
 BorderColor #4ca0d9  
 BorderThickness 1  
  
 FontColor #0079c9  
 FontName Roboto, SansSerif, sans-serif  
 FontSize 12  
 FontStyle bold  
  
}  
  
  
'For Capabilities  
skinparam frame {  
  
 StereotypeFontColor #616161  
  
 BorderColor #4CA0D9  
 BorderThickness 1  
 FontName Roboto, SansSerif, sans-serif  
 FontColor #0079c9  
 FontStyle bold  
 BackgroundColor #FFFFFF  
 BodyBackGroundColor #FFFFFF  
}  
  
' External systems in architecture diagrams  
skinparam node {  
  
 BorderThickness 1  
 BackgroundColor #FFFFFF  
 BorderColor #4ca0d9  
  
 FontColor #0079c9  
 FontName Roboto, SansSerif, sans-serif  
 FontSize 12  
 FontStyle bold  
  
}  
  
skinparam package {  
 BorderThickness 1  
 BackgroundColor #FFFFFF  
 BorderColor #1281ac  
  
 FontColor #1281ac  
 FontName Roboto, SansSerif, sans-serif  
 FontSize 12  
 FontStyle bold  
  
}  
  
' Activity diagrams, for different capabilities  
  
skinparam partition {  
  
 BackgroundColor #f3f3f3  
  
 BorderColor #1281ac  
 BorderThickness 2  
  
 FontColor #1281ac  
 FontName Roboto, SansSerif, sans-serif  
 FontSize 12  
 FontStyle bold  
  
}  
  
  
skinparam file {  
 BorderThickness 1  
 BackgroundColor #FFFFFF  
 BorderColor #1281ac  
  
 FontColor #1281ac  
 FontName Roboto, SansSerif, sans-serif  
 FontSize 12  
 FontStyle bold  
  
}  
  
' ref (erences)  
skinparam sequenceReferenceBorderColor #4ca0d9  
skinparam sequenceReferenceBackgroundColor #ffffff  
skinparam sequenceReferenceHeaderBackgroundColor #d9ebf6  
  
skinparam queue {  
 BackgroundColor #ffffff  
 BorderThickness 1  
 BorderColor #4ca0d9  
 FontColor #0079c9  
 FontStyle italic  
}  
  
skinparam legend {  
BackgroundColor #ffffff  
}  
  
start  
:Log in to the \*\*nGenue\*\* application;  
:Click the Search icon and type \*\*Supply valuation pool\*\* in the search bar;  
note right: If no results appear,\nretry search.  
:Double-click \*\*Supply valuation pool\*\* in the search results;  
  
:Navigate to the screen with two sections:  
- \*\*Valuation pool selection\*\*  
- \*\*Valuation pool configuration\*\*;  
:Click the \*\*Add\*\* button in the Valuation pool configuration section;  
  
:Fill in the following details:  
- \*\*Val pool ID\*\* (auto-generated)  
- \*\*Valuation pool\*\* name  
- \*\*Description\*\*  
- \*\*Commodity\*\*  
- \*\*Market index/curve\*\*  
- \*\*Effective date\*\*  
- \*\*End date\*\*;  
note right: Ensure all fields are entered correctly.  
  
:Save the configuration;  
  
:Click \*\*Assign points\*\* field and select a pipeline;  
:Enable the \*\*Pipeline checkbox\*\* and select the pipeline;  
:Click \*\*Load location\*\* to display available locations;  
  
:Use forward arrows (>> or >) to add locations;  
note right: Use backward arrows (<< or <) if\nlocations need to be removed.  
  
:Click \*\*Accept\*\* to save changes;  
  
stop  
@enduml

### Process steps

#### Step 1: Navigate to Internal supply valuation pool screen

1. Log in to the **nGenue** application.
2. Click the **Search** icon and type \_Supply valuation pool\* in the search bar.
3. Double-click **Supply valuation pool** in the search results to open the **Menu configuration** screen. navigate\_supply\_valuation\_pool
4. The next screen has two main sections: **Valuation pool selection** and **Valuation pool configuration**. supply\_valuation\_pool
   1. **Valuation pool selection:** Displays a list of existing valuation pools.
   2. **Valuation Pool configuration:** Shows the details of the selected pool, allowing you to add, update, or edit configurations. The table below describes the available icons and their functions:

| Icons | Description |
| --- | --- |
| alt text | Add a new record |
| alt text | Edit the record detail. |
| alt text | Save the record. |
| alt text | Cancel the updates being made to the record. |
| alt text | Delete a record. |

#### Step 2: Configure a valuation pool

1. Open the **Valuation pool configuration** screen and click the **Add** button. add
2. Enter the following details under the **Valuation pool details** tab:
   * **Val pool ID:** The **ID** is auto-generated once you save the configuration.
   * **Valuation pool name:** Add a descriptive name for the pool.
   * **Description:** Include optional context details.
   * **Commodity:** Select the relevant option from the dropdown.
   * **Market index/curve:** Select from the dropdown list.
   * **Effective date:** Specify the date when the pool becomes active.
   * **End date:** Set the expiration date or leave it blank for indefinite use.

#### Step 3: Configure pipeline points

1. Click **Assign points to** field and select a pipeline from the dropdown menu. Assign\_points
2. Enable the **Pipeline** checkbox and select the pipeline from the dropdown.
3. Click **Load location** to display the available locations.
4. Add locations by:
   * Clicking the **forward arrow** (**>>**) key to include all locations.
   * Clicking the **forward arrow** (**>**) key to include selected locations.
5. Remove locations using the **backward arrows** (**<< or <**) if necessary.
6. Click **Accept** to save your changes. configuring\_pipeline\_point
7. The system will display the assigned valuation pool under the **Valuation pool** section on the **Location** screen.