

Prudential®

ICM Interface

BP021- RAFM Projects



|  |  |  |
| --- | --- | --- |
| Author | : | *Irram Sherwani, Michael Dmytrenko* |
| Manager | : | *Sander De Rooij* |
| Document ID | : | *BP022 – RAFM Projects* |
| Date last updated | : | *9 June 2022* |
| Version | : | *1.8.3* |
| Status | : | *Final* |
| Document Name | : | *BP022 RAFM Projects\_v1.8.3.docx* |
| Classification | : | *Confidential* |

Document Information

|  |  |
| --- | --- |
| **Organization** | Prudential |
| **Author** | Irram Sherwani, Michael Dmytrenko |
| **Manager** | Sander De Rooij |
| **Title** | ICM Interface |
| **Subtitle** | BP022 RAFM Projects |
| **Document ID** | BP022 |
| **Date last updated** | 9 juni 2022 |
| **Version** | 1.8.3 |
| **Status** | *Final* |
| **Document Name** | BP022 RAFM Projects\_v1.8.3.docx |
| **Classification** | *Confidential* |

Version History

|  |  |  |  |
| --- | --- | --- | --- |
| **Version** | **Date** | **Status** | **Comments** |
| 1.0 | 23/4/2019 | Draft | Published version for Release 6.4.0.0 |
| 1.1 | 22/05/2019 | Draft | Updates based on comments from clients. |
| 1.2 | 24/06/2019 | Draft | Updates based on comments from clients. |
| 1.3 | 28/06/2019 | Final | Updates on Merged project valdation upon modification. |
| 1.4 | 28/08/2020 | Draft | Published version for Release 6.8.0.0 |
| 1.4.1 | 13-10-2020 | Draft | Published version for release 6.8.5.1 including changes after revision by the client |
| 1.5 | 10-11-2020 | Final | Published version for Release 6.9.0.2 |
| 1.6 | 29-01-2021 | Final | Draft version for Release 7.0.0.0 |
| 1.7 | 15-08-2021 | Draft | Draft version for Release 8.1.0.0 |
| 1.7.1 | 28-09-2021 | Final | Published version for Release 8.1.1.0 |
| 1.8 | 20-05-2022 | Draft | Draft version for Release 8.2.0.0 |
| V1.8.1 | 27-05-2022 | Draft | Feedback from Prudential included |
| V1.8.2 | 01-06-2022 | Draft | 2nd draft version for Release 8.2.0.0 |
| V1.8.3 | 01-06-2022 | Draft | Published version for Release 8.2.0.0 |

References

|  |  |  |
| --- | --- | --- |
| **Doc-ID** | **Document** | **Version** |
|  |  |  |

Open and Closed Issues

|  |  |  |
| --- | --- | --- |
| **#** | **Issue** | **Status** |
|  |  |  |

Target readership

|  |  |  |
| --- | --- | --- |
| **Role** | **Purpose** | **Comments** |
| Manager Group Risk |  |  |
| Project Team Implementation |  |  |

Review table

Please mark an **X** on the document version you have reviewed.

This will enable the Project Board approval process to be sped up.

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Role** | **Name** | **1.4.1** | **1.5** | **1.6** | **1.7** | **1.7.1** | **1.8** | **1.8.3** |
| Capital Management and Modelling, Risk Modelling, Head Of | Stephen Hainsworth |  | x | x | x |  |  |  |
| Capital Management and Modelling, Risk Modelling, Senior Manager | David Van Der Merwe |  | x | x | x |  |  | x |
| Capital Management and Modelling, Interim Actuarial Consultant | Cassandra Moore |  | x | x | x |  |  | x |
| Project Manager | Ola Olaopa |  | x | x | x |  |  | x |
| Solution Architect | Petre Todorov | x | x | x | x | x | x | x |
| Lead Developer | Laurent Severac | x | x |  | x | x | x | x |
| Developer | Agnes Regeni |  |  |  | x | x |  |  |
| Project manager | Sander de Rooij | x | x | x | x | x |  | x |
| Business Analyst | Irram Sherwani | x | x | x |  |  |  |  |
| Custom Projects Solution Owner | Michael Dmytrenko | x | x | x | x | x | x | x |
| Senior Test Analyst | Ruchi Patni | x | x | x | x | x |  | x |

Approvals

|  |  |  |  |
| --- | --- | --- | --- |
| **Role** | **Name** | **Signature** | **Date** |
| Internal Capital Model Oversight Committee (ICMOC) |  |  |  |

Table of Contents

[Use Cases 4](#_Toc105686692)

[1.1 RAFM001 View RAFM Project 4](#_Toc105686693)

[1.2 RAFM002 Create RAFM Project 12](#_Toc105686694)

[1.3 RAFM003 Share Base Engine RAFM Project with other Geographies 25](#_Toc105686695)

[1.4 RAFM011 Share Merged Project ICM RAFM Project with other Geographies 26](#_Toc105686696)

[1.5 RAFM004 Validate RAFM Project 27](#_Toc105686697)

[1.6 RAFM005 Modify RAFM Project 28](#_Toc105686698)

[1.7 RAFM006 Delete RAFM Project 41](#_Toc105686699)

[1.8 RAFM010 Copy RAFM Projects 42](#_Toc105686700)

[1.9 RAFM007 Download RAFM Project 43](#_Toc105686701)

[1.10 RAFM008 Filter RAFM Project 43](#_Toc105686702)

[1.11 RAFM009 Filter by Tags 46](#_Toc105686703)

[1.12 RAFM010 Copy RAFM Projects 47](#_Toc105686704)

[1.13 RAFM012 Rename RAFM Projects 48](#_Toc105686705)

[Appendix 49](#_Toc105686706)

[I. Code Blocks BU Bridge Aggregator codebase 49](#_Toc105686707)

[II. WDF Versions of the RAFM Project 49](#_Toc105686708)

[III. Identification of codebases and consistency check based on codemetadata.json 50](#_Toc105686709)

[IV. Identification of codebases and consistency check based on cpp files 51](#_Toc105686710)

Use Cases

## RAFM001 View RAFM Project

RAFM Projects

### Use Case

|  |  |
| --- | --- |
| **Constraints** | |
| Pre-condition | User should have necessary role(s) and permission(s) |
| Pre-condition | At least one RAFM Project is available in the system i.e. RAFM002- “Create RAFM Project” use case is complete. |
| Post-condition | RAFM Project is displayed in the summary grid. |

|  |  |
| --- | --- |
| **Scenarios** | |
| **Basic Path** |  |
| Basic Path | 1. User navigates to RAFM Projects tab*.* 2. RAFM Projects is displayed in the following panes:   - Summary Grid;  - Code;  - Details for;  - Properties for;  - Selected Tags; |

### Requirements

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **RQ-2001 User should be able to view all the available RAFM Project versions in the summary grid table.** | | | | | | | | | | |
| The following fields shall be displayed on the summary grid table for all the  RAFM Project records:   * **Name**: name of the RAFM project. * **Version**: Version of the RAFM project. * **Status**: Status of the RAFM project (“In Review” or “Validated”) * **Type**: Type of RAFM project (“Base Engine” or “Standard ICM RAFM Project” or Merged ICM RAFM Project). * **Description**: Description of the RAFM project. * **Geographical Rights**: user group that owns the RAFM Project; * **Last modification local date**: Timestamp of the RAFM Project create action or modify as per local time zone of user. * **Archive status**: The Archive Status of RAFM Project | **Status** | | | **:** | | | | | **Open** | |
| **Type** | | | **:** | | | | | **Functional** | |
| **Phase** | | | **:** | | | | | **6.1.0.0** | |
| **From** | | | **:** | | | | | **Irram Sherwani** | |
| **RQ-2002 The System shall display the “Properties for < [RAFM Project Name]\_[version no.]>” panel in the dashboard.** | | | | | | | | | | |
| The system shall display the following fields in the properties panel for the selected RAFM project version.  **Name:** name of the RAFM project version selected;  **RAFM project File**: Name of the .zip file uploaded for the selected RAFM project.    **RAFM project software version**:  ICM shall recognize and display RAFM project software version. Depends on RAFM engine generation version it should be parsed from:  For RAFM Projects that contain codebases in a stand alone .cpp files:  RAFM Project – ICM.msproj file. The data in this file is held in an XML tag structure. The tag that holds the RAFM Project software version is:  <ProjectFileVersion></ProjectFileVersion>  Example:  <ProjectFileVersion>2.6.603.0</ProjectFileVersion>  Note that all historical projects already in the ICM have used version 2.6.603.0. Therefore they have been updated with the RAFM Project software version: 2.6.603.0  For RAFM Projects that contain codebases in a codemetadata.json file:  Project software version is stored in the version property, e.g.: "Version": "2.9.0.1431"  **Note:** If the codemetadata.json file is present, the system shall use it to perform all actions related to defining, checking, validating, recognizing, displaying codebases. If the file is missing, the system performs the above actions with standalone \*.cpp files, each of which is responsible for information about a particular codebase.  **Status:** Status of the selected RAFM project;  **Tag**: System shall display the following tags in the field:   * All ICM RAFM Project version when they are created will receive the tag ‘New’ * Standard ICM RAFM Project inherits the entity set/assumption set tags they are assigned through their codebase to lite models’ assignment. * Base Engine ICM RAFM Project inherits the scenario set tags it is assigned. * When a selected lite model or aggregation rules codebase of an ICM RAFM Project is un-assigned from Entity Set (for Standard ICM RAFM Project), or from selected scenario set that it was assigned to (for Base Engine RAFM Project) then relevant tag will be disappeared * When an ICM RAFM Project is un-assigned to all lite models/aggregation rules (Standard), all scenario sets it was assigned to (Base Engine) then it will receive the tag ‘Rejected’ automatically   **WDF Version:** version of the selected RAFM project. See more details in the Appendix II.  The value of this property is extracted from ICM codebase of the RAFM project, using the following logic:   * If ICM codebase contains a keyword with the following structure: [**New\_WDF\_Format\_vXXX**] then the WDF version of the RAFM project will be **XXX**. * If there is no such keyword it will get a WDF version value **0**.   Existing RAFM projects already uploaded in the system will get WDF version value **0**.  **Allowed Geographies:** list of geographies that can access the RAFM projects (owner user group and user groups the RAFM Projects has been shared with).  **Base Engine**: This property displays the Base Engine assigned to ICM Standard or Merged RAFM Project. **Assumption Set:** Nested Assumption set version assigned to RAFM Project. Applicable only for Merged Projects.  **Last modified by:** User id of the last user that has updated the selected RAFM project.  **Last modified local date:** timestamp of the last modification of the RAFM Project as per local time zone of user;  **Last modified system date**: timestamp of the last modification of the RAFM Project as per system time zone.  The Property names in the pane shall be displayed in alphabetical order. | | **Status** | | | **:** | | | | | **Open** |
| **Type** | | | **:** | | | | | **Functional** |
| **Phase** | | | **:** | | | | | **6.1.0.0** |
| **From** | | | **:** | | | | | **Irram Sherwani** |
| **RQ-2003 The system shall display the Selected Tags panel on the left hand side of the screen (Collapsible window).** | | | | | | | | | | |
| This summary table of the selected tags panel shall contain all tags created in the system (created under System Administration>Tag management) which are been made available for filter. | | **Status** | | | **:** | | | | | **Open** |
| **Type** | | | **:** | | | | | **Functional** |
| **Phase** | | | **:** | | | | | **6.0.0.0** |
| **From** | | | **:** | | | | | **Irram Sherwani** |
| **RQ-2004 The system shall display the Code panel in the dashboard** | | | | | | | | | | |
| The system shall display the code of currently selected codebase in the “Details” panel.   * The system must display code of a selected Lite model or Aggregation Rules or Bridge Aggregator codebase. * The system must not display code of a selected API or RSG codebase. In that case the “Code” panel must be empty for any type of RAFM Project. * Depending on RAFM Generation engine version (to be more precise, depending on whether the file codemetadata.json is present or absent) the code shall be populated:   + For RAFM Projects with only .cpp codebase files and nocodemetadata.json:   + ‘as is’ it stated in the relevant codebase file {codebase\_file\_name}.cpp   + For RAFM Projects with codemetadata.json codebases (irrespective of .cpp files being present):   + The strings contained in the Names of the items in the CodeManager -> ModelClasses correspond to the name of a codebase, i.e. a codebase will be created for each entry in the ModelClasses section.   + For each entry in the “ExternalFunctions” (if any exist) concatenate the “Name” entry followed by the entry from the “Formula” attribute.   + For each entry in the “Scalars” (if any exist) concatenate the “Name” entry and the value from the “Formula” attribute.   + For each entry in the “TempTables” (if any exist) concatenate the “Name” entry and the value from the “Formula” attribute.   + For each entry in the “Variables” (if any exist) concatenate the “Name” entries.   + For each entry in the “Columns” element (if any exist) concatenate “Name” entry and the value from the “Formula” attribute .   + Between each section a heading should be inserted to indicate which section is concatenated, e.g. “Columns section”   + Entries within the .json file are arranged alphabetically, firstly by modelclass name and then by Column, ExternalFunction,Scalar, TempTable andVariableames. The alphabetical order within each modelclass codebase should be preserved when creating the concatenated text. | | **Status** | | | **:** | | | | | **Open** |
| **Type** | | | **:** | | | | | **Functional** |
| **Phase** | | | **:** | | | | | **6.1.0.0** |
| **From** | | | **:** | | | | | **Irram Sherwani** |
| RQ-2005 The system shall display “Events” tab of the “Details” table in the dashboard | | | | | | | | | | |
| The system shall display the following information:  **Type**: contains type of action that was performed for the RAFM Project  **Version**:shows the version of the RAFM Project after the action has been performed.  **User:** user-id of user who performed action.  **Local Date**:date and time on which the action was performed in the local time zone of user.  **System Date**: The system shall show the list of events that have been performed on the RAFM Project for the selected version.  Types of possible events:  **Create**;  **Modify**:the system shall provide **system comment** with extra information:   * Latest validated version of the RAFM Project that is currently available and modifying (“Generated from *%version*. Modified attributes: *[%what’s\_new]*);   **Share**: the system generates event “Share” if a RAFM project has been shared;  **Validate;**  This table is by default empty, the system shall populate it when a version of a RAFM project in the summary table is selected. | | | **Status** | | | **:** | | **Open** | | |
| **Type** | | | **:** | | **Functional** | | |
| **Phase** | | | **:** | | **6.0.0.0** | | |
| **From** | | | **:** | | **Irram Sherwani** | | |
| **RQ-2006 The system shall display the Scenario set in a tab of the “Details” table in the dashboard** | | | | | | | | | | |
| This table shall be displayed with the following information regarding usage of Base Engine RAFM Project in Scenario Set(s).  **Name**: Name of the Scenario set that uses the selected Base Engine RAFM Project  **Version**: Version of the Scenario set that uses the selected Base Engine RAFM Project  **Base Date**: Base date of the Scenario set;  **Description**: description of the Scenario set;  **Type**: type of the Scenario set;  **Purpose**: purpose of the Scenario set;  **Session Date**: Session date of the Scenario set.  **Status**: Status of Scenario set.  **Last Run Local Date**: timestamp of Scenario Set run as per Local time zone of user  **Last Run System Date**: system timestamp of the Scenario Set run.  **Last Run Status**: Status of last Run.  **Last Run Triggered by**: user-id that triggered Scenario Set run.  The information in the columns: Last Run Local Date, Last Run System Date, Last Run Status, Last Run Triggered by is populated when the run is triggered to an Scenario set  By default this table is empty. The system shall populate it when a version of the Base Engine RAFM Project  is selected in the summary table is selected. This table shall be empty if user selects any Standard ICM RAFM Project | | | **Status** | | | **:** | | **Open** | | |
| **Type** | | | **:** | | **Functional** | | |
| **Phase** | | | **:** | | **6.0.0.0** | | |
| **From** | | | **:** | | **Irram Sherwani** | | |
| **RQ-2007 The system shall display the Usage tab of the “Details” table in the dashboard** | | | | | | | | | | |
| This table shows in which assumption set the selected version of the RAFM Project has been assigned to.   * For Base engine : assumption set versions using a scenario set using that base engine * For Standard Project: assumption sets using LM/AR from that standard project (through the entity set) * For Merged project: assumption sets having the merged project assigned to them   For each assumption set  the system shall show the following information:  **Name**;  **Version**;  **Description**;  **Type**;  **Status**;  **Last Run Local Date**;  **Last Run System Date**;  **Last Run Status**;  **Last Run Triggered by**.   By default this table is empty. The system shall populate it when a version of the RAFM Project in the summary table is selected.   The information in these columns: Name, Version, Description, Type and Status is populated at the moment a RAFM Project is used in an assumption set through assigning of Entity Set.   The information in the columns: Last Run Local Date, Last Run System Date, Last Run Status, Last Run Triggered by is populated when the run is triggered to an assumption set. Status | | | **Status** | | | **:** | | **Open** | | |
| **Type** | | | **:** | | **Functional** | | |
| **Phase** | | | **:** | |  | | |
| **From** | | | **:** | | **Irram Sherwani** | | |
| **RQ-2008 The system shall display the Lite Model in a tab of the “Details” table in the dashboard** | | | | | | | | | | |
| This table shows in which Lite Model the selected version of the RAFM Project has been assigned to.  For each Lite Model  the system shall provide the following information:  **Name**;  **Version**;  **Status**;  **Base Date**.  **Go To:** link to the Lite Model page. (the Lite Model which uses the selected RAFM project) This link shall be available for all users which has access to the Lite Model entity which uses the selected RAFM Project.  Clicking on the link will open the Lite Model Tab with the Lite Model (in summary table) which uses the RAFM project already selected and filtered to just highlight this one entity.  By default this table is empty. The system shall populate it when a version of the RAFM Project in the summary table is selected. | | | **Status** | | | **:** | | **Open** | | |
| **Type** | | | **:** | | **Functional** | | |
| **Phase** | | | **:** | | **6.1.0.0** | | |
| **From** | | | **:** | | **Irram Sherwani** | | |
| **RQ-2009 The system shall display the Aggregated Rules in a tab of the “Details” table in the dashboard** | | | | | | | | | | |
| This table shows in which Aggregation rules the selected version of the RAFM Project has been assigned to.  For each Aggregation Rules  the system shall show the following information:  **Name**;  **Version**;  **Status**;  **Base Date;**  **Go To:** link to the Aggregation Rule page. (the Aggregation Rule which uses the selected RAFM project) This link shall be available for all users which has access to the Aggregation Rule Entity which uses the selected RAFM Project.  Clicking on the link will open the Aggregation Rules Tab with the Aggregation Rule (in summary table) which uses the RAFM project already selected and filtered to just highlight this one entity.  By default this table is empty. The system shall populate it when a version of the RAFM Project in the summary table is selected. | | | **Status** | | | | **:** | **Open** | | |
| **Type** | | | | **:** | **Functional** | | |
| **Phase** | | | | **:** | **6.1.0.0** | | |
| **From** | | | | **:** | **Irram Sherwani** | | |
| **RQ-2010 The system shall update the codebase MetaData in a tab of the “Details” table in the dashboard** | | | | | | | | | | |
| This table shows the codebases of selected RAFM project. The table is populated for each version of a RAFM Project. The table is updated upon Creation/Modification of the RAFM Project.  Following fields shall be populated for each RAFM project in the table.  **Name**: name of the codebases of the selected version of the RAFM project.  **Type**: Type of the codebase (either RSG or API or Lite Model Or Aggregation Rule)  **Version**: Version of the codebase. Refer to RQ-2011 for more details on the version of the codebases. | | | **Status** | | | | **:** | **Open** | | |
| **Type** | | | | **:** | **Functional** | | |
| **Phase** | | | | **:** | **6.0.0.0** | | |
| **From** | | | | **:** | **Irram Sherwani** | | |
| **RQ-2011 The system shall create a new version of the codebase Metadata based on following rules.** | | | | | | | | | | |
| Upon Creation of new RAFM project, all the codebases of the RAFM project shall be assigned to version 1.0 (with an exception for codebase type ‘BU Bridge Aggregators’ for RAFM project type ‘Base Engine’. For this combination the system shall not keep the version number upon creation and modification):  Upon modification of the RAFM Project which is in status “In Review” , the system shall update the codebase version numbers as per below rules:   * If the codebase in the new RAFM project zip file does not exist in the existing zip file, then a new codebase will be created with version 1.0. * If a codebase from the new RAFM project zip file already exists in the RAFM project version, then the system must not increment the version of the codebase and must replace the old codebase with the new codebase.   Note: Only codebases which are present in the new version of the RAFM project zip file must be included in the new version of the RAFM project. For confirmation, codebases which are not a part of a new version of RAFM Project must not be copied from previous version of a RAFM Project to the new version of the RAFM Project.     * If a codebase is removed from the project zip file, which has previously been assigned to a LM, then the system must prevent the user from uploading the zip file.   Upon modification of the RAFM Project which is in status “Validated” , the system shall update the codebases/version numbers as per below rules:   * If the codebase name in the new RAFM package does not exist in the existing project package then new codebase will be created and populated in the updated RAFM Project with codebase version 1.0. * If in the uploading RAFM package there is a codebase with the name, which is already exists in the latest validated version of the RAFM project, the system shall make codebase code comparison (comparison between existing codebase and new codebase of the same model): * If the new codebase content is the same with existing codebase content, the system must not increment the version of the codebase. The system just replaces old codebase with the new one. * If the new codebase content differs from the existing codebase content, the system must increment the version of the codebase by 1.0 | | | **Status** | | | | **:** | **Open** | | |
| **Type** | | | | **:** | **Functional** | | |
| **Phase** | | | | **:** | **6.1.0.0** | | |
| **From** | | | | **:** | **Irram Sherwani** | | |

## RAFM002 Create RAFM Project

RAFM Projects

### Use Case

|  |  |
| --- | --- |
| **Constraints** | |
| Pre-condition | User should have necessary role(s) and permission(s) |
| Pre-condition |  |
| Pre-condition | The new created RAFM Project is stored in the system and available to allowed users. |

|  |  |
| --- | --- |
| **Scenarios** | |
| **Basic Path** |  |
| Basic Path | 1. User navigates to “RAFM Projects” main tab 2. User selects the option "Create" under “Maintenance” menu button. 3. System displays a “Create RAFM Project” pop up dialog with the following fields for user to enter or view.  * Name: mandatory free text field; * Version: non-editable field populating the version of the RAFM project which is always 1.0 upon create action; * RAFM Project Type: mandatory drop-down list (possible values - Standard ICM RAFM Project; Base Engine; Merged ICM RAFM Project) * Description: Optional free text field; * Project package: Upload field.  1. User selects “RAFM Project Type” as “Base Engine” and enters other required fields. 2. User selects the button “Save” on “Create RAFM Project” dialog. 3. System checks for the validations of the Base Engine RAFM Project *(Refer* ***RQ-2016a*** *for more details).* 4. When all the validations are passed. 5. The system stores the RAFM Project and updates the RAFM project summary screen. 6. The system retrieves names of the codebase and shows them in “Details” panel of the RAFM Project |
| **Alternate 3.1** | |  |
| Alternate | | If user tries to upload Project package having extension anything other than .zip then system shall show a tooltip error message.  ***“The file has to have .zip extension”*** |
| **Alternate 4.1** |  |
| Alternate | 1. User enters the required fields and selects “RAFM Project Type” as “Standard ICM RAFM Project”. 2. System displays additional mandatory field “Base Engine” on the dialog for user to enter. 3. User enters/selects value in all required fields and the use case returns to basic path point no. 5. |
| **Alternate 4.2** |  |
| Alternate | 1. User enters the required fields and selects “RAFM Project Type” as “Merged Project”. 2. System displays additional mandatory fields on the dialog for user to enter:  * Assumption Set version: Find button which upon clicking displays pop up window displaying name and version of the published assumption sets. * Base Engine: Drop-down list;  1. User enters/selects value in all required fields and the use case returns to basic path point no. 5. |
| **Alternate 5.1** |  |
| Alternate | User choose the option “Cancel”, system aborts the process. |
| **Alternate 5.2** |  |
| Alternate | | If the name entered is not unique ( i.e. User has entered “Name” which already exists in the system either for Base Engine or Standard ICM RAFM Project) then system will show below error message and the use case is aborted  ***“Could not create the RAFM Project. Reason:****The RAFM project name ‘%name’ already exists in the system”* |
| **Alternate 5.3** | |  |
| Alternate | | The system shows an tooltip error message if any of the mandatory field is left blank by user |
| **Alternate 5.4** | | |  |
| Alternate | | 1. If a user tries to uploads a RAFM project having RAFM project version which is not configured in system (System Admin>RAFM configuration), then system shall give a warning message advising the user that the version of RAFM is not contained within the RAFM configuration page and user will be unable to run the RSG or assumption set until this has been updated. 2. The user will be given the option to continue with upload or abort the upload. |
| **Alternate 6.1** | |  |
| Alternate | | System checks for the validations of the Standard RAFM Project *(Refer* ***RQ-2016b*** *for more details).* Use Case returns to Basic path point no.7 |
| **Alternate 6.2** | |  |
| Alternate | | System checks for the validations of the Merged RAFM Project *(Refer****RQ-2016c***  *for more details).*  Use Case returns to Basic path point no.7 |
| **Alternate 7.1** | | **For Standard RAFM Project and Merged RAFM Project** |
|  | | If any of the consistency check validations fail, system shall popup an error message dialog with a link “Download Report”. Upon clicking the link system shall download a consistency check failure report (excel format) for the user. In the report, user can check which of consistency check validations are failed. |
| **Alternate 7.2** | | **For Base Engine RAFM Project** |
|  | | If any of the consistency check validations fail, system shall throw an error message and use case is aborted |

### Requirements

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **RQ-2012 User should be able to create a RAFM Project in the system if user has necessary Role(s) and Permission(s)** | | | | | | | | | | | |
| Please refer to Appendix:” BP009 User manager: User Roles Permissions” for more details on relevant roles and permissions. | **Status** | | | **:** | | | | | | **Open** | |
| **Type** | | | **:** | | | | | | **Functional** | |
| **Phase** | | | **:** | | | | | | **6.1.0.0** | |
| **From** | | | **:** | | | | | | **Irram Sherwani** | |
| **RQ-2013 System shall allow a user to create a RAFM Project of Type “Base Engine”** | | | | | | | | | | | |
| System shall allow a user to create the RAFM project of Type” Base Engine”. Following field shall be displayed for creating the RAFM Project of Type “**Base Engine**”.   1. **Name**: User can specify the desired Name of RAFM Project Set in this field. Name should not contain any of the below special characters:   ^ : ~ | , ' . ` ! @ # $ £ % & \* ( ) + { } [ ] " ; < > ? / = and white spaces.  Name should be unique within all types of RAFM projects.   1. **Version:** non-editable field populating with the value 1.0 by system for each RAFM project that is to be created. 2. **RAFM Project type:** Possible values:  * Base Engine * Standard ICM RAFM Project * Merged ICM RAFM Project   For creating RAFM project of Type “Base Engine”, User should select the value “Base Engine” in this field.   1. **Description:** User can enter any free text in the description field or choose to leave it empty. 2. **Project Package:** User can select .zip file in this field | **Status** | | | **:** | | | | | | **Open** | |
| **Type** | | | **:** | | | | | | **Functional** | |
| **Phase** | | | **:** | | | | | | **6.0.0.0** | |
| **From** | | | **:** | | | | | | **Irram Sherwani** | |
| **RQ-2014 System shall allow a user to create a RAFM Project of Type “Standard ICM RAFM Project”.** | | | | | | | | | | | |
| System shall allow a user to create the RAFM project of Type” Standard Engine”. Following field shall be displayed for creating the RAFM Project of Type “**Standard ICM RAFM Project**”.   1. **Name**: User can specify the desired Name of RAFM Project Set in this field. Name should not contain any of the below special characters:   ^ : ~ | , ' . ` ! @ # $ Â£ % & \* ( ) + { } [ ] " ; < > ? / = and white spaces.  Name should be unique within all types of RAFM projects.   1. **Version:** non-editable field populating with the value 1.0 by system for each RAFM project that is to be created. 2. **RAFM Project type:** Possible values:  * Base Engine * Standard ICM RAFM Project * Merged ICM RAFM Project   For creating RAFM project of Type “Standard ICM RAFM Project”, User should select the value “Standard ICM RAFM Project” in this field.   1. **Base Engine:** this field consists of two drop-down lists where user can select a Base Engine and its version which was shared with the Geography who are creating the Standard ICM RAFM Project. 2. **Description:** User can enter any free text in this field or choose to leave it empty. 3. **Project Package:** User can select .zip file in this field. | | **Status** | | | **:** | | | | **Open** | | |
| **Type** | | | **:** | | | | **Functional** | | |
| **Phase** | | | **:** | | | | **6.0.0.0** | | |
| **From** | | | **:** | | | | **Irram Sherwani** | | |
| **RQ-2015 System shall allow a user to create a RAFM Project of Type “Merged”** | | | | | | | | | | | |
| System shall allow a user to create the RAFM project of Type ”Merged Project”. Following field shall be displayed for creating the RAFM Project of Type “**Merged ICM RAFM Project**”.   1. **Name**: User can specify the desired Name of RAFM Project Set in this field. Name should not contain any of the below special characters:   ^ : ~ | , ' . ` ! @ # $ £ % & \* ( ) + { } [ ] " ; < > ? / = and white spaces.  Name should be unique within all types of RAFM projects.   1. **Version:** non-editable field populating with the value 1.0 by system for each RAFM project that is to be created. 2. **RAFM Project type:** Possible values:  * Base Engine * Standard ICM RAFM Project * Merged ICM RAFM Project   For creating RAFM project of Type “Merged ICM RAFM Project”, User should select the value “Merged ICM RAFM Project” in this field.   1. **Assumption Set:** this field shall be populated with the list of published nested assumption sets owned by/shared with the current geography of the user. **Base Engine:** this field consists of two drop-down lists where user can select a Base Engine (1) and its version (2) which is owned by or shared with the Geography from which Merged ICM RAFM Project is to be created. 2. **Description:** User can enter any free text in the description field or choose to leave it empty. 3. **Project Package:** User can select .zip file in this field. | **Status** | | | **:** | | | | | | **Open** | |
| **Type** | | | **:** | | | | | | **Functional** | |
| **Phase** | | | **:** | | | | | | **6.1.0.0** | |
| **From** | | | **:** | | | | | | **Irram Sherwani** | |
| **RQ-2016a The System shall perform the following when a user selects “Save” on the “Create a RAFM project” dialog for Base Engine.** | | | | | | | | | | | |
| The System must perform the following checks when user selects “Save” on the “Create RAFM Project” dialog.  **Structural Checks:**   * + All the mandatory codebases corresponding to RSG/API codebase configured under System Administration->”Base Engine Code Base Management” must be present in a RAFM project package.   System shall only check the codebases in the RAFM project package for this validation.  Any codebases corresponding to RSG/API which are marked as “No” under the mandatory column in “System Administration” ->” Base Engine Code Base Management”, can also be present in the project package but there must be no other codebases contained in the project package which are not included in the “System Administration” -> ”Base Engine Code Base Management”.   * + All the mandatory codebases corresponding to BU Bridge aggregator codebases configured under “System Administration” -> ”BU Bridge Aggregator Code Base management” must be present in a Base Engine RAFM project package.   Any codebases corresponding to BU Bridge aggregator which are marked as “No” under the mandatory column in “System Administration” -> ”BU Bridge Aggregator Code Base Management”, can also be presenting the project package but there must be no BU Bridge aggregator codebases contained in the project package which are not included in the “System Administration” -> ”BU Bridge Aggregator Code Base Management”.   * + LM/AR codebases must not be present in RAFM project package. Please refer to RQ-2023 for more details on how the system identifies LM/AR codebases in the project package.   If the above structural checks are passes then system must create the Base Engine RAFM Project with status “In Review” and version 1.0.  If the above structural fails then system must throw an error message and RAFM project shall not be created in the system  **Note:** If the codemetadata.json file is present, the system shall use it to perform all actions related to defining, checking, validating, recognizing, displaying codebases. If the file is missing, the system performs the above actions with standalone \*.cpp files, each of which is responsible for information about a particular codebase. If no .json or .cpp file(s) are present an error message should be displayed. | | | **Status** | | **:** | | | | **Open** | | |
| **Type** | | **:** | | | | **Functional** | | |
| **Phase** | | **:** | | | | **6.1.0.0** | | |
| **From** | | **:** | | | | **Irram Sherwani** | | |
| **RQ-2016b The System shall perform the following when a user selects “Save” on the “Create a RAFM project” dialog for Standard ICM RAFM project.** | | | | | | | | | | | |
| The System must perform the following checks when user selects “Save” on the “Create RAFM Project” dialog for Standard RAFM project.  **Structural Codebase Checks:**  API and RSG codebases of the uploaded Standard ICM RAFM package are identical to the codebases of the selected Base Engine.   * + All the mandatory codebases corresponding to RSG/API codebase configured under “System Administration” -> ”Base Engine Code Base Management” must be present in a Standard ICM RAFM project package. Any codebases corresponding to RSG/API which are marked as “No” under the mandatory column in “System Administration” -> ”Base Engine Code Base Management”, can also be present in the project package but there must be no other codebases contained in the project package which are not included in the “System Administration” -> ”Base Engine Code Base Management”.   All the mandatory codebases corresponding to BU Bridge aggregator codebase configured under “System Administration” -> ”BU Bridge Aggregator Code Base management” must be present in a Standard ICM RAFM package uploaded upon creation.   * + Any codebases corresponding to BU Bridge aggregator which are marked as “No” under the mandatory column in “System Administration” -> ”BU Bridge Aggregator Code Base Management”, can also be present in the project package but there must be no BU Bridge aggregator codebases contained in the project package which are not included in the “System Administration” -> ”BU Bridge Aggregator Code Base Management”.     **Functional Checks:**   * + System checks the API and RSG codebases between the standard project and base engine for both functional and structural differences   i.e. it checks the RSG/API codebases names are the same and the relevant code within each codebase is the same.   * + At least one LM or AR codebase must be present in uploaded Standard ICM RAFM project package. The three-letter prefix of all the LM/AR codebases available in the package should be same and there should be a BU Bridge Aggregator codebase available in the package with a name equal to this prefix.   + Please refer to RQ-2023 for more details on how the system identifies LM/AR codebases in the project package.   Note: System shall not perform any functional consistency check between the BU Bridge Aggregator codebase of Base Engine selected and BU Bridge Aggregator codebase of the uploading Standard ICM RAFM package.  If all the above structural and functional checks are passed successfully then system must create a Standard ICM RAFM Project with status “In-Review” and version 1.0 in the system.  If any of the above structural or functional checks are failed, system must throw an error message and RAFM project shall not be created in the system | | | **Status** | | **:** | | | | **Open** | | |
| **Type** | | **:** | | | | **Functional** | | |
| **Phase** | | **:** | | | | **6.1.0.0** | | |
| **From** | | **:** | | | | **Irram Sherwani** | | |
| **RQ-2016c The System shall perform the following when a user selects “Save” on the “Create a RAFM project” dialog for Merged ICM RAFM project.** | | | | | | | | | | | |
| The System must perform the following checks when user selects “Save” on the “Create RAFM Project” dialog:  **Structural Codebase Checks:**   * + API and RSG codebases of the uploaded Merged ICM RAFM project are identical to the codebases of the selected Base Engine.   + All the mandatory codebases corresponding to RSG/API codebases configured under “System Administration” -> ”Base Engine Code Base Management” must be present in a Merged ICM RAFM project package.   Any codebases corresponding to RSG/API which are marked as “No” under the mandatory column in “System Administration” -> ”Base Engine Code Base Management”, can also be present in the project package but there must be no other codebases contained in the project package which are not included in the “System Administration” ->”Base Engine Code Base Management”.   * + All the mandatory codebases corresponding to BU Bridge aggregators codebases configured under “System Administration” -> ”BU Bridge Aggregator Code Base management” must be present in the uploading Merged ICM RAFM package   Any codebases corresponding to BU Bridge aggregator which are marked as “No” under the mandatory column in “System Administration” -> ”BU Bridge Aggregator Code Base Management”, can also be present in the project package but there must be no BU Bridge aggregator codebases contained in the project package which are not included in the “System Administration” -> ”BU Bridge Aggregator Code Base Management”.  **Functional Checks:**  System checks the API and RSG codebases between the uploading merged project and the base engine for both functional and structural differences i.e. it checks the RSG/API codebases names are the same and the relevant code within each codebase is the same.   * It is required to associate uploading Merged ICM RAFM package with the existing nested assumption set. The system must identify all Standard ICM RAFM Projects which codebases are assigned to selected nested assumption set and compare them with the uploading Merged ICM RAFM Package based on the following requirements:   + LM/AR codebases of the uploaded Merged ICM RAFM package must be equal in quantity, names and relevant content to LM/AR codebases of all identified Standard ICM RAFM Projects. Uploading Merged ICM RAFM package must not include any other LM/AR codebases.   + Each Standard ICM RAFM Project has all LM/AR codebases starting with same three letter prefix and a BU Bridge Aggregator codebase with a name equal to this prefix.   + The name of one of the BU Bridge Aggregator codebases available in the ICM RAFM Merged Project package should be equal to this prefix. Only this particular BU Bridge Aggregator codebase must be used for consistency check between uploading Merged ICM RAFM package and this particular identified Standard ICM RAFM Project.   + For example: the selected Assumption Set has assigned LM/AR codebases with prefixes ‘JNL’ and ‘PCA’.   + ICM will identify sourced Standard Projects of these codebases and geographies of these Standard projects together with their BU Bridge Aggregators (For Standard Project 1 – ‘JNL’ BU Bridge Aggregator and for Standard project 2 – ‘PCA’ BU Bridge Aggregator).   + The uploading Merged ICM RAFM package must have all identified BU Bridge Aggregator codebases - ‘JNL’ and ‘PCA’, where ‘JNL’ bridge aggregator will be functionally validated against the ‘JNL’ bridge aggregator on the ‘Standard RAFM Project 1’, and equally the ‘PCA’ bridge aggregator on the ‘Merged Project’ will be functionally validated against the ‘PCA’ bridge aggregator on the ‘Standard RAFM Project 2’. * When performing the 'functional' consistency check between the BU bridge aggregators of the Merged ICM RAFM Project and the individual Standard ICM RAFM Projects, the system checks that each LM/AR codebase which is referred to or declared within the Merged ICM RAFM Project BU Bridge Aggregator is present in both the Merged ICM RAFM Project file and the corresponding geography Standard ICM RAFM Project, and that these codebases contain the same relevant content.   When comparing the merged and standard RFM project BU bridge Aggregators, the order of the references or declarations of the LM/AR codebases are ignored.  This check shall be completed by the following:  System will check the block of code which contains ‘modelNameNosub’ in it for each LM and AR codes base in the project package (see Appendix: Code Blocks BU Bridge Aggregator codebases).  In the Standard project and Merged project identical code blocks shall be available for corresponding same geography in the package.  **Example:**  The GHO codebase contained in the GHO standard project will contain identical code blocks to the GHO codebase contained in the merged project.  The JNL codebase contained in the JNL standard project will contain identical code blocks to the JNL codebase contained in the merged project.  The PCA codebase contained in the PCA standard project will contain identical code blocks to the PCA codebase contained in the merged project  The code blocks shall not be necessarily in same order in standard and merged project. The order of code blocks should not cause any system validation error as long as there are same number of code blocks in both the projects, and the respective matching code blocks contain identical relevant content.  Conversely, differences in LM and AR codebases (from the bridge aggregator codebase (Appendix: Code Blocks BU Bridge Aggregator codebases) should cause the codebases check to fail.  **Note**:- System shall not perform any **functional** consistency check between the BU Bridge Aggregator codebases of selected Base Engine and BU Bridge Aggregator codebases of the uploading Merged ICM RAFM package).  If all the above structural and functional checks are passed successfully then system shall create a Merged ICM RAFM Project with status “In-Review” and version 1.0 in the system.  If any of the above structural or functional checks are failed, system shall throw an error message and RAFM project shall not be created in the system  Note: selected assumption set is used only for consistency check upon creation of a Merged ICM RAFM Project and is not being saved as a property of the Merged ICM RAFM Project (however, references to the Standard ICM RAFM Projects which are associated with the selected assumption set are saved when the Merged ICM RAFM Project is created, and form the basis for further consistency checks when this project is modified). The created project will not be automatically assigned to the selected assumption set and must be assigned manually from the Assumption Set tab (please refer to the RQ-2077 The system shall allow the user to assign the Merged ICM RAFM Project to published nested Assumption set version | | | **Status** | | **:** | | | | **Open** | | |
| **Type** | | **:** | | | | **Functional** | | |
| **Phase** | | **:** | | | | **6.1.0.0** | | |
| **From** | | **:** | | | | **Irram Sherwani** | | |
| **RQ-2017 The system shall assign the status to the RAFM Project** | | | | | | | | | | | |
| The status assigned by the system to a new created RAFM project is "In Review". | | | **Status** | | | **:** | | **Open** | | | |
| **Type** | | | **:** | | **Functional** | | | |
| **Phase** | | | **:** | | **6.0.0.0** | | | |
| **From** | | | **:** | | **Irram Sherwani** | | | |
| **RQ-2018 The system shall assign the owner user group to the RAFM project** | | | | | | | | | | | |
| The owner user group is the user group of the user that uploads the files for the creation of the RAFM Project. | | | **Status** | | | | **:** | **Open** | | | |
| **Type** | | | | **:** | **Functional** | | | |
| **Phase** | | | | **:** | **6.0.0.0** | | | |
| **From** | | | | **:** | **Irram Sherwani** | | | |
| **RQ-2019 The system shall allow a user to upload the Base Engine RAFM Project in a zip format** | | | | | | | | | | | |
| The system shall allow a user to upload a .zip file with RAFM Project files containing:  - API codebase;  - RSG codebase;  - BU Bridge Aggregator codebase(s)  *Refer RQ-2016a/2016b/2016c for more details on validations on the Project Package uploaded.* | | **Status** | | | | **:** | | | | | **Open** |
| **Type** | | | | **:** | | | | | **Functional** |
| **Phase** | | | | **:** | | | | | **6.0.0.0** |
| **From** | | | | **:** | | | | | **Irram Sherwani** |
| **RQ-2020 The system shall allow a user to upload a Standard ICM RAFM Project in zip format** | | | | | | | | | | | |
| The system shall allow a user to upload a .zip file with RAFM Project containing:  -API codebase;  -RSG codebase;  -One or multiple Lite Model codebases;  -One or multiple Aggregation Rule codebases;  -BU Bridge Aggregator codebase(s).  Refer RQ-2016a/2016b/2016c for more details on validations on the Project Package uploaded. | | **Status** | | | | **:** | | | | | **Open** |
| **Type** | | | | **:** | | | | | **Functional** |
| **Phase** | | | | **:** | | | | | **6.1.0.0** |
| **From** | | | | **:** | | | | | **Irram Sherwani** |
| **RQ-2021 The system shall allow a user to upload a Merged ICM RAFM Project in zip format** | | | | | | | | | | | |
| The system shall allow a user to upload a .zip file with Merged ICM RAFM Project containing:  -combination of two or more Standard ICM RAFM Projects. Refer RQ-2020 for more details on Standard Project .zip file.  *Refer RQ-2016a/2016b/2016c for more details on validations on the Project Package uploaded.* | | **Status** | | | | **:** | | | | | **Open** |
| **Type** | | | | **:** | | | | | **Functional** |
| **Phase** | | | | **:** | | | | | **6.1.0.0** |
| **From** | | | | **:** | | | | | **Irram Sherwani** |
| **RQ-2022 The system shall assign the default tag to the RAFM Project** | | | | | | | | | | | |
| The default tag for a new RAFM Project is "New". | | **Status** | | | | **:** | | | | | **Open** |
| **Type** | | | | **:** | | | | | **Functional** |
| **Phase** | | | | **:** | | | | | **6.0.0.0** |
| **From** | | | | **:** | | | | | **Irram Sherwani** |
| **RQ-2023 The system shall apply the following logic to identify the LM or AR codebases in the project package of RAFM upon upload of ICM RAFM Project.** | | | | | | | | | | | |
| The naming convention for LMs and Ars shall be as follows:   * For LM and AR codebase – XXX\_YY\_Z   Where,  XXX = gho, jnl or pca depending on ownership (will be 3 letters)  YY = LM or AR marker (will be 2 letters)  Z = name of the codebase (can have varying number of letters)  The ICM Interface will identify LM and AR codebase by assessing the name of each codebase and checking the 5th and 6th characters of the string.  Where the combination of the letters are ‘lm’ or ‘ar’, these codebases can be identified as a ‘lite model’ or ‘aggregation rule’, respectively.  The check described above should not be case sensitive. | | **Status** | | | | **:** | | | | | **Open** |

## RAFM003 Share Base Engine RAFM Project with other Geographies

RAFM Projects

### Use Case

|  |  |
| --- | --- |
| **Constraints** | |
| Pre-condition | The RAFM Project summary table is populated with at least one Base Engine RAFM Project. |
| Pre-condition | User has necessary role(s) and permission(s). |
| Pre-condition | Base Engine RAFM Project has status “Validated”. |
| Pre-condition | User belongs to the owner user group. |
| Post-condition | Selected version of a Base Engine RAFM Project is shared with the selected geographies. |

|  |  |
| --- | --- |
| **Scenarios** | |
| **Basic Path** |  |
| Basic Path | 1. User selects a version of a Base Engine RAFM Project.  2. User selects the option "Share" under “Maintenance” menu.  3. The system shows the list of geographies available in the system.  4. User selects one or more geographies.  5. User selects the option "Save".  6. The system makes the selected version of Base Engine RAFM Project accessible to users of the selected geographies. |
| **Alternate 4.1** |  |
| Alternate | If none of the versions of the selected Base Engine RAFM Project are linked to a Lite Model or Aggregated Rule (via a Standard ICM RAFM Project that uses the selected Base Engine), the user can deselect the geographies that have access to the RAFM Project. The geographies deselected will no longer have access to the Base Engine RAFM Project. |
| **Alternate 5.1** |  |
| Alternate | If user clicks the button "Cancel" the use case is aborted. |

### Requirements

|  |  |  |  |
| --- | --- | --- | --- |
| **RQ-2028 The system shall allow an authorized user to share a Base Engine RAFM Project (selected versions) with other geographies.** | | | |
| The system should display the available geographies in the system (taken from the uploaded hierarchy) and allow a user to select one or more. The geography of the owner user group is always selected and cannot be deselected.  If the Base Engine RAFM Project has been used by a geography, this geography cannot be deselected. | **Status** | **:** | **Open** |
| **Type** | **:** | **Functional** |
| **Phase** | **:** | **6.0.0.0** |
| **From** | **:** | **Irram Sherwani** |
| **RQ-2029 The system shall allow a user to deselect a geography (un-share) a selected Base Engine RAFM Project in the following cases.** | | | |
| The system shall not allow a user to deselect a geography (un-share) from a selected version of the Base Engine RAFM Project if:   * a Standard ICM RAFM Project was created based on the selected Base Engine version * A Base Engine RAFM Project is being used in Scenario Set | **Status** | **:** | **Open** |
| **Type** | **:** | **Functional** |
| **Phase** | **:** | **6.0.0.0** |
| **From** | **:** | **Irram Sherwani** |

## RAFM011 Share Merged Project ICM RAFM Project with other Geographies

RAFM Projects

### Use Case

|  |  |
| --- | --- |
| **Constraints** | |
| Pre-condition | The RAFM Project summary table is populated with at least one Merged ICM RAFM Project. |
| Pre-condition | User has necessary role(s) and permission(s). |
| Pre-condition | Merged ICM RAFM Project has status “Validated”. |
| Pre-condition | User belongs to the owner user group. |
| Post-condition | Selected version of a Merged ICM RAFM Project is shared with the selected geographies. |

|  |  |
| --- | --- |
| **Scenarios** | |
| **Basic Path** |  |
| Basic Path | 1. User selects a version of a Merged ICM RAFM Project.  2. User selects the option "Share" under “Maintenance” menu.  3. The system shows the list of geographies available in the system.  4. User selects one or more geographies.  5. User selects the option "Save".  6. The system makes the selected version of Merged ICM RAFM Project accessible to users of the selected geographies. |
| **Alternate 4.1** |  |
| Alternate | If none of the versions of a Merged ICM RAFM Project are assigned to an Assumption set of the shared geographies, the user can deselect the geographies that have access to the RAFM Project. The geographies deselected will no longer have access to the Merged ICM RAFM Project. |
| **Alternate 5.1** |  |
| Alternate | If user clicks the button "Cancel" the use case is aborted. |

### Requirements

|  |  |  |  |
| --- | --- | --- | --- |
| **RQ-2081 The system shall allow an authorized user to share a Merged ICM RAFM Project (selected versions) with other geographies.** | | | |
| The system should display the available geographies in the system (taken from the uploaded hierarchy) and allow a user to select one or more. The geography of the owner user group is always selected and cannot be deselected.  If the Merged ICM RAFM Project has been used by geography, this geography cannot be deselected. | **Status** | **:** | **Open** |
| **Type** | **:** | **Functional** |
| **Phase** | **:** | **6.1.0.0** |
| **From** | **:** | **Irram Sherwani** |
| **RQ-2082 The system shall allow a user to deselect a geography (un-share) a selected Merged ICM RAFM Project in the following cases.** | | | |
| The system shall not allow a user to deselect a geography (un-share) from a selected version of the Merged ICM RAFM Project if:   * a Merged ICM RAFM Project is being used in Assumption Set of the shared geography. | **Status** | **:** | **Open** |
| **Type** | **:** | **Functional** |
| **Phase** | **:** | **6.1.0.0** |
| **From** | **:** | **Irram Sherwani** |

## RAFM004 Validate RAFM Project

RAFM Projects

### Use Case

|  |  |
| --- | --- |
| **Constraints** | |
| Pre-condition | The selected version of the RAFM Project is in status "In Review” |
| Pre-condition | User should have the necessary role(s) and permission(s) |
| Pre-condition | The logged in user is different from the "Last modified by" user. |
| Post-condition | A selected version of the RAFM Project is in status "Validated" |
| Post-condition | the RAFM Project is available for “use” by other components. |

|  |  |
| --- | --- |
| **Scenarios** | |
| **Basic Path** |  |
| Basic Path | 1. User selects a version of a RAFM Project (Base Engine or Standard ICM RAFM project or Merged ICM RAFM Project) in the RAFM Project summary table.  2. User selects the option "Validate" under “Validation” menu.  3. The system displays a “Validate RAFM Project” pop up window containing the:      - Comments: non mandatory free text format field;      - Upload file: non mandatory upload field;  4. User fills in any of the above fields or choose to keep them empty  5. User selects the "Validate" button available on the “Validate RAFM Project” dialog.  6. The system changes the status of the selected RAFM Project to "Validated" status and updates the summary table. |
| **Alternate 5.1** |  |
| Alternate | If user selects the button "Cancel" the user case is aborted. |

### Requirements

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **RQ-2024 The system shall allow a user to validate a RAFM project in status "In Review”** | | | | |
| Upon performing validation on the selected Project, system shall change the status of the RAFM project from “In Review” to “Validated” and the | | **Status** | **:** | **Open** |
| **Type** | **:** | **Functional** |
| **Phase** | **:** | **6.0.0.0** |
| **From** | **:** | **Irram Sherwani** |
| **RQ-2025 The system shall allow a user to enter a comment and/or to upload a file.** | | | | |
| Both the comment and upload file are optional fields. | | **Status** | **:** | **Open** |
| **Type** | **:** | **Functional** |
| **Phase** | **:** | **6.0.0.0** |
| **From** | **:** | **Irram Sherwani** |
| **RQ-2026 The system shall not allow the last user who has modified the version of the RAFM Project to validate it.** | | | | |
| The system shall not allow the last user who has modified the version of the RAFM Project to validate it. It should be a different user. | | **Status** | **:** | **Open** |
| **Type** | **:** | **Functional** |
| **Phase** | **:** | **6.0.0.0** |
| **From** | **:** | **Irram Sherwani** |
| RQ-2027 A user should be able to validate a RAFM Project in the system if the user has necessary Role(s)/Permission(s) | | | | |
| Please refer to Appendix:” BP009 User Manager:User Roles Permissions” for more details on relevant roles and permissions. | **Status** | | **:** | **Open** |
| **Type** | | **:** | **Functional** |
| **Phase** | | **:** | **6.1.0.0** |
| **From** | | **:** | **Irram Sherwani** |

## RAFM005 Modify RAFM Project

RAFM Projects

### Use Case

|  |  |
| --- | --- |
| **Constraints** | |
| Pre-condition | User belongs to the user group that owns the RAFM Project or to a geography to which the RAFM Project has been manually shared with. |
| Pre-condition | User should have necessary role(s) and permission(s). |
| Post-condition | The RAFM Project is updated (if it is in status "In Review") or a new version for the RAFM Project is created (if it is “Validated”). |
| Post-condition | If the RAFM Project is modified which is used in Assumption set runs:   * Assumption set run in status ‘Queued’ or ‘Requested, that uses the selected RAFM project (status In-Review) is Cancelled * Assumption set run in status ‘Completed’ that uses the selected RAFM project (status In-Review) is invalidated. * Assumption set run in status ‘Processing’ that uses the selected RAFM project (status In-Review) is invalidated after the runs are completed. |

|  |  |  |
| --- | --- | --- |
| **Scenarios** | | |
| **Basic Path** | |  |
| Basic Path | | 1. User selects a version of a RAFM Project in the summary table. 2. User selects the option “Modify” under “Maintenance” menu 3. The system shows a pop-up window “Modify RAFM Project” containing:  * Name: non-editable field; * Version: non-editable field; * RAFM Project Type: non-editable field; * Description: Optional free text editable field; * Base Engine: editable dropdown field drop-down list (applicable only if the RAFM Project selected for modify is Standard ICM RAFM project and Merged ICM RAFM Project); * Base Engine Ver.: editable drop-down list (applicable only if the RAFM Project selected for modify is Standard or Merged ICM RAFM project); * Assumption Set version: Find button which upon clicking displays pop up window displaying name and version of the published Assumption sets with nested Entity sets. (applicable only if the RAFM Project selected for modify is Merged ICM RAFM project); The selected assumption set will be used for consistency check only and not saved as a property of the Merged ICM RAFM Project. * Project package: editable upload field; * Comments: Optional free text editable field;  1. User modifies any of the editable field. 2. User selects the button “Save” on the “Modify RAFM Project” dialog. 3. System checks for the validations (Ref RQ-2031). 4. If all the validations are successfully passed, system updates the RAFM project (if it has status “In Review”) or creates a new version of the RAFM Project (if it has status “Validated”). 5. The system updates the RAFM Project summary screen /Details pane and Properties pane. 6. The system shows a confirmation message on the screen.   ***“<RAFM Project>\_<version no.> is successfully modified”.*** |
| **Alternate 2.1** | | **For ICM Standard RAFM Project and Merged RAFM Project:** |
|  | | System displays a warning message if the RAFM Project (in status “In-Review”) is used in run and the run is still in queued status or is still processing.  *Refer RQ-2230 for more details.* |
| **Alternate 5.1** | |  |
| Alternate | | If user selects the button “Cancel”, use case is aborted. |
| **Alternate 5.2** | |  |
| Alternate | | 1. If a user tries to upload a RAFM project having RAFM project version which is not configured in system (System Admin>RAFM configuration), then system shall give a warning message advising the user that the version of RAFM is not contained within the RAFM configuration page and user will be unable to run the RSG or assumption set until this has been updated. 2. The user will be given the option to continue with upload or abort the upload. |
| **Alternate 6.1** | **For Base Engine RAFM Project:** |
| Alternate | If any of the validation fails, an error message(s) is displayed by the system and use case is aborted  Please refer to RQ-2031a The System shall perform the following validations when user selects “Save” button on the “Modify a RAFM project” dialog*.* |
| **Alternate 6.2** | **For ICM Standard RAFM Project and Merged RAFM Project:** |
|  | If any of the consistency check validations fail, system shall popup an error message dialog with a link “Download Report”. Upon clicking the link system shall download a consistency check failure report (excel format) for the user. In the report, user can check which of consistency check validations are failed.  Please refer to RQ-2031a/RQ-2031b The System shall perform the following validations when user selects “Save” button on the “Modify a RAFM project” dialog*.* |

### Requirements

|  |  |  |  |
| --- | --- | --- | --- |
| **RQ-2030 The system shall allow a user to modify any version of a RAFM Project** | | | |
| The system shall allow a user to modify only the below fields for the selected version of the RAFM project.  In the case of a Base Engine RAFM project, the user can modify the following:   * Description: User can update any free text description in this field or choose to keep it empty. This field shall be optional; * Project package: User can upload any other   Base Engine RAFM Project.zip package in this field.   * Comments: User can update any free text description in this field or choose to keep it empty. This field shall be optional;   In case of Standard ICM RAFM project, user can modify the following:   * Description: User can update any free text description in this field or choose to keep it empty. This field shall be optional; * **Base Engine:** In this field, the system shall only populate the Base Engines:   + Which have status “Validated “.   *(Refer Use Case –“Validate RAFM Project” for more details on validation status of RAFM projects).*  **And**   * + Which are shared with the Geography from which Standard ICM RAFM Project is to be created.   User can select any other available Base Engine from the drop-down list. Refer RQ-2031 for more details on validations.   * Project package: User can upload a   new Standard ICM RAFM Project.zip package in this field.Refer RQ-2031 for more details on validations.   * Comments: User can update any free text description in this field or choose to keep it empty. This field shall be optional;   In case of Merged ICM RAFM project, user can modify the following:   * Description: User can update any free text description in this field or choose to keep it empty. This field shall be optional; * Base Engine: User can select any other Base Engine version in this field. * Project package: User can select any other.zip package in this field. The system remembers the original assumption set selected on creation of the merged RAFM project and completes a consistency check between the new merged project package and the original assumption set. The user is unable to view the assumption set that is associated to the merged RAFM Project (the information is held in the backend of the system). * Comments: User can update any free text description in this field or choose to keep it empty. This field shall be optional; | **Status** | : | **Closed** |
| **Type** | : | **Functional** |
| **Phase** | : | **6.1.0.0** |
| **From** | : | **Irram Sherwani** |
| **RQ-2031a The System shall perform the following validations when user selects “Save” button on the “Modify a RAFM project” dialog for Base Engine.** | | | |
| The System shall perform the following checks when user selects “Save” on the “Modify RAFM Project dialog” for base engine.  **Structural Checks:**   * The system must recognize codebases. * All the mandatory codebases corresponding to RSG/API codebases configured under “System Administration” -> ”Base Engine Code Base Management” must be present in a RAFM project package.   System shall check the codebases in the RAFM project package for this validation.  Any codebases corresponding to RSG/API which are marked as “No” under the mandatory column in “System Administration” -> ”Base Engine Code Base Management”, can also be present in the project package but there must be no other codebases in the project package which are not included in the “System Administration” -> ”Base Engine Code Base Management”.   * + All the mandatory codebases corresponding to BU Bridge aggregators codebases configured under “System Administration| -> ”BU Bridge Aggregator Code Base Management” must be present in a RAFM project package.   Any codebases corresponding to BU Bridge aggregator which are marked as “No” under the mandatory column in “System Administration” -> ”BU Bridge Aggregator Code Base Management”, can also be present in the project package but there must be no BU Bridge aggregator codebases in the project package which are not included in the “System Administration” ->” BU Bridge Aggregator Code Base Management”.   * + LM/AR codebases must not be present in RAFM project package.   If the above structural checks are successfully passed, then:   * + System shall update the selected version of RAFM project in case the RAFM Project has status “In Review”.   + System shall modify and increase the version of modified RAFM Project in case the selected RAFM project has status “Validated” (version number of the modified RAFM Project shall be incremented as per RQ- 2032)   + If the Base Engine is validated then the system shall compare API and RSG codebases from the project package zip file in the previous version and the project package zip file being saved. If the Base engine is in review then these checks will not be performed.   For version control of the codebases upon modify action, please refer to the RQ-2011 “The system shall create a new version of the codebases in the codebase metadata based on following rules”  If any of the above validation is failed, system must throw an error message and RAFM project shall not be modified. | **Status** | : | **Closed** |
| **Type** | : | **Functional** |
| **Phase** | : | **6.1.0.0** |
| **From** | : | **Irram Sherwani** |
| **RQ-2031b The System shall perform the following validations when user selects “Save” button on the “Modify a RAFM project” dialog for Standard ICM RAFM Project.** | | | |
| The System shall perform the following checks when user selects “Save” on the “Modify RAFM Project dialog” for standard RAFM Project.  Upon saving the changes of RAFM Project type “Standard ICM RAFM Project”, system shall validate the following:  **Structural Checks:**   * + API and RSG codebases of the uploading Standard ICM RAFM package are identical to the codebases of the selected Base Engine.   + All the mandatory codebases corresponding to RSG/API codebases configured under “System Administration” -> ”Base Engine Code Base Management” must be present in a Standard ICM RAFM project package.   Any codebases corresponding to RSG/API which are marked as “No” under the mandatory column in “System Administration” -> ”Base Engine Code Base Management”, can also be present in the project package but there must be no other codebases in the project package which are not included in the “System Administration” -> ”Base Engine Code Base Management”.   * + All the mandatory codebases corresponding to BU Bridge Aggregator codebases configured under “System Administration” -> ”BU Bridge Aggregator Code Base management” must be present in Standard ICM RAFM project package.   Any codebases corresponding to BU Bridge aggregator which are marked as “No” under the mandatory column in “System Administration” -> ”BU Bridge Aggregator Code Base Management”, can also be present in the project package but there must be no BU Bridge aggregator codebases in the project package which are not included in the “System Administration” -> ”BU Bridge Aggregator Code Base Management”.  **Functional Checks:**   * + System checks the API and RSG codebases between the standard project and base engine for both functional and structural differences i.e. it checks the RSG/API codebase names are the same and the relevant code within each codebase is the same.   + At least one LM or AR codebase must be present in uploaded RAFM project package. The three-letter prefix of all the LM/AR codebases available in the package should be same and there should be a BU Bridge Aggregator codebase available in the package with a name equal to this prefix.   Please refer Please refer to RQ-2035 for more details on how to identify LM/AR codebase in the project package.   * + Requirement RQ-2249   Note: Upon modification, the system shall not perform any functional consistency check between the BU Bridge Aggregator codebases of Base Engine selected and BU Bridge Aggregator codebases of the Standard ICM RAFM package uploaded.  If all the above structural and functional checks are successfully passed:   * + System updates the selected version of RAFM project in case the RAFM Project has status “In Review” without incrementing the version of the RAFM Project   + System modifies and increases the version of modified RAFM Project in case the selected RAFM project has status “Validated” (Version number of the modified RAFM Project shall be incremented as per RQ-2032).   + The codebases will be incremented in version as per RQ-2011   If any of the above structural or functional check fails, then system shall throw an error message and RAFM project shall not be modified. | **Status** | : | **Closed** |
| **Type** | : | **Functional** |
| **Phase** | : | **6.1.0.0** |
| **From** | : | **Irram Sherwani** |
| **RQ-2031c The System shall perform the following validations when user selects “Save” button on the “Modify a RAFM project” dialog for Merged RAFM Project.** | | | |
| The System shall perform the following checks when user selects “Save” on the “Modify RAFM Project dialog “for merged RAFM project.  **Structural Checks:**   * + API and RSG codebases of the uploading merged project are identical to the codebases of the selected base engine.      * All the mandatory codebases corresponding to RSG/API codebases configured under “System Administration” -> ”Base Engine Code Base Management” must be present in a RAFM Merged project package.   Any codebases corresponding to RSG/API which are marked as “No” under the mandatory column in “System Administration” -> ”Base Engine Code Base Management”, can also be present in the project package but there must be no other codebases contained in the project package which are not included in the “System Administration” -> ”Base Engine Code Base Management”.   * All the mandatory codebases corresponding to BU Bridge Aggregator codebases configured under “System Administration” ->” BU Bridge Aggregator Code Base Management” must be present in Merged ICM RAFM project package.   Any codebases corresponding to BU Bridge aggregator which are marked as “No” under the mandatory column in “System Administration -> ”BU Bridge Aggregator Code Base Management”, can also be present in the project package but there must be no BU Bridge aggregator codebases contained in the project package which are not included in the “System Administration” -> ”BU Bridge Aggregator Code Base Management”.  **Functional Checks:**  System ICM checks the API and RSG codebases between the merged project and the base engine for both functional and structural differences i.e. it checks the codebase names are the same and the relevant code within each codebase is the same.  Further consistency checks between the uploading Merged ICM RAFM Project package and Standard ICM RAFM Projects will be based on the Standard ICM RAFM Projects already associated with the selected Merged ICM RAFM Project which is being modified.  The system is required to associate the updated Merged ICM RAFM package with the nested assumption set that was assigned to the Merged RAFM Project upon creation. The system must identify all Standard ICM RAFM Projects including which codebases are assigned to the selected nested assumption set. The system will compare the standard RAFM projects and their codebases with the updated Merged ICM RAFM Package based on the following requirements:   * LM/AR codebases of the updating Merged ICM RAFM package must be equal in quantity, names and content to LM/AR codebases of all identified Standard ICM RAFM Projects. Updating Merged ICM RAFM package must not include any other LM/AR codebases. * Each Standard ICM RAFM Project has all LM/AR codebases starting with same prefix three letter prefix and a BU Bridge Aggregator codebase with a name equal to this prefix.   The name of one of the BU Bridge Aggregator codebase available in the ICM RAFM Merged Project package should be equal to this prefix. Only this particular BU Bridge Aggregator codebase must be used for consistency check between uploading Merged ICR RAFM package and this particular identified Standard ICM RAFM Project.  For example:  If the Assumption Set (selected upon creation of merged project) has assigned LM/AR codebases with prefixes ‘PCA and ‘JNL.  ICM will identify source Standard Projects of these codebases and geographies of these Standard projects together with their BU Bridge Aggregators (For Standard Project 1 – ‘PCA’ BU Bridge Aggregator and for Standard project 2 – ‘JNL’ BU Bridge Aggregator).  The uploading Merged ICM RAFM package must have all identified BU Bridge Aggregator codebases - ‘PCA’ and ‘JNL’, where ‘PCA’ bridge aggregator will be functionally validated against the ‘PCA’ bridge aggregator on the ‘Standard RAFM Project 1’, and equally the ‘JNL’ bridge aggregator on the ‘Merged Project’ will be functionally validated against the ‘JNL’ bridge aggregator on the ‘Standard RAFM Project 2’.  When performing the 'functional' consistency check between the BU bridge aggregators of the Merged ICM RAFM Project and the individual Standard ICM RAFM Projects, the system checks that each LM/AR codebase which is referred to or declared within the Merged ICM RAFM Project BU Bridge Aggregator is present in both the Merged ICM RAFM Project file and the corresponding geography Standard ICM RAFM Project, and that these codebases contain the same relevant content.  When comparing the merged and standard RFM project BU bridge Aggregators, the order of the references or declarations of the LM/AR codebases are ignored.    This check shall be completed by the following:  System will check the block of code which contains ‘modelNameNosub’ in it for each LM and AR codes base in the project package (see Appendix: Code Blocks BU Bridge Aggregator codebases).  In the Standard project and Merged project identical code blocks shall be available for the corresponding same geography.  Example:  The GHO codebase contained in the GHO standard project will contain identical code blocks to the GHO codebase contained in the merged project.  The JNL codebase contained in the JNL standard project will contain identical codeblocks to the JNL codebase contained in the merged project.  The PCA codebase contained in the PCA standard project will contain identical code blocks to the PCA codebase contained in the merged project  The code blocks shall not be necessarily in same order in standard and merged project. The order of code blocks should not cause any system validation error as long as there are same number of identical code blocks in both the projects.  However differences in LM and AR codebases (from the bridge aggregator codebases (Appendix: Code Blocks BU Bridge Aggregator codebase) should cause the codebases check to fail.  **Note:** Upon modification, the system shall not perform any functional consistency check between the BU Bridge Aggregator codebases of Base Engine selected and BU Bridge Aggregator codebases of the Merged ICM RAFM Project uploaded.    Please refer Please refer to RQ-2035 for more details on how to identify LM/AR codebase in the project package.  If the above validation is successfully passed, then:   * + System shall update the selected version of RAFM project in case the RAFM Project has status “In Review”.   + System shall modify and increase the version of modified RAFM Project in case the selected RAFM project has status “Validated” (version number of the modified RAFM Project shall be incremented as per RQ- 2032).   For version control of the codebases upon modify action, please refer to the RQ-2011 “The system shall create a new version of the merged RAFM Project”.  If any of the above structural or functional check is failed, system shall throw an error message and RAFM project shall not be modified in the system. | **Status** | : | **Closed** |
| **Type** | : | **Functional** |
| **Phase** | : | **6.1.0.0** |
| **From** | : | **Irram Sherwani** |
| **RQ-2032 The system shall create a new version of the RAFM Project if the modified version has status “Validated"** | | | |
| Upon modify action when the user modifies the validated RAFM project, system shall create a new version of the modified RAFM Project in the summary table. The system creates a new version and increases the version number according to the following rules.   1. Every new code file modification (via uploading a new Project package) of the current validated RAFM project will increment the major version of the RAFM project by 1.x (dropping minor version to x.0)   Example: RAFM Project ver. 1.0 has been modified and a new version 2.0 has been created. Another code files modification of the RAFM Project ver. 1.0 will increment the version to 3.0, because version 2.0 is already occupied by another version of the RAFM Project   1. Every new metadata modification (via changing the description, for example) of the current validated RAFM project will increment the minor version of the RAFM project by x.1, major version will be the same.   Example: RAFM Project ver. 1.0 has been modified and a new version 2.0 has been created. Another metadata modification of the RAFM Project ver. 1.0 will increment the version to 1.1, even though higher version - 2.0 is already created and available. | **Status** | **:** | **Open** |
| **Type** | **:** | **Functional** |
| **Phase** | **:** | **6.0.0.0** |
| **From** | **:** | **Irram Sherwani** |
| **RQ-2033 The system shall update the existing version of the RAFM Project if the modified version has status “In Review"** | | | |
| Upon modify action when user modifies the RAFM project in status “In Review”, system shall update the changes in the same version of the RAFM Project. In this case system shall not create a new version of the RAFM Project in the summary table. | **Status** | **:** | **Open** |
| **Type** | **:** | **Functional** |
| **Phase** | **:** | **6.0.0.0** |
| **From** | **:** | **Irram Sherwani** |
| RQ-2034 User should be able to modify the RAFM Project in the system if user has necessary Role(s)/Permission(s) | | | |
| Please refer to Appendix:” BP009 User Manager: User Roles Permissions” for more details on relevant roles and permissions. | **Status** | **:** | **Open** |
| **Type** | **:** | **Functional** |
| **Phase** | **:** | **6.1.0.0** |
| **From** | **:** | **Irram Sherwani** |
| **RQ-2035 The system shall apply the following logic to identify the LM or AR codebases in the project package of RAFM upon re-upload of ICM RAFM Project.** | | | |
| The naming convention for LMs and ARs shall be as follows:   * For LM and AR codebase – XXX\_YY\_Z   Where,  XXX = gho, jnl or pca depending on ownership (will be 3 characters)  YY = LM or AR marker (will be 2 characters)  Z = name of the codebase (can have varying number of characters)  The ICM Interface will identify LM and AR codebases by assessing the name of each codebase and checking the 5th and 6th characters of the string.  Where the combination of the letters are ‘lm’ or ‘ar’, these codebases can be identified as a ‘lite model’ or ‘aggregation rule’, respectively.  The check described above should not be case sensitive. | **Status** | **:** | **Open** |
| **Type** | **:** | **Functional** |
| **Phase** | **:** | **6.1.0.0** |
| **From** | **:** | **Irram Sherwani** |
| RQ-2230 System should invalidate the Assumption Set Run with ‘Queued’, ‘Processing’ or ‘Completed’ status if the RAFM project version included in the Assumption set run is modified. | | | |
| If the user tries to modify the RAFM project version(in status In-Review) which is included in an assumption set runs (in status of ‘Queued’, ‘Processing’ or ‘Completed’), the system shall display a warning message informing the user that if the user will proceed with the modification action, the system will invalidate/cancel the linked assumption set runs.  If the user confirms to the modification action, then:   * Assumption set run in status ‘Queued’ or ‘Requested, that uses the selected RAFM project (status In-Review) is Cancelled * Assumption set run in status ‘Completed’ that uses the selected RAFM project (status In-Review) is invalidated. * Assumption set run in status ‘Processing’ that uses the selected RAFM project (status In-Review) is invalidated after the runs are completed..   Following modifications shall be in scope for invalidating the assumption set runs:   * + - Uploading a new ‘Project Package’ to replace the current project package assigned.     - Updating the ‘Base Engine’ version that is embedded within the Standard RAFM project. | **Status** | **:** | **Open** |
| **Type** | **:** | **Functional** |
| **Phase** | **:** | **6.6.0.0** |
| **From** | **:** | **Irram Sherwani** |
| RQ-2249 System should not allow the user to modify In-Review Standard RAFM project version if the modified project no longer contains codebase which has already been assigned to a LM/AR. | | | |
| When a standard project is In-Review, if a new zip file is uploaded (upon modify action), which no longer contains a codebase which has already been assigned to a Lite Model or Aggregation Rules, then the system does not allow the user to make this modification. System will display an error message and  modify action shall not be successful. | **Status** | **:** | **Open** |
| **Type** | **:** | **Functional** |
| **Phase** | **:** | **6.6.0.0** |
| **From** | **:** | **Irram Sherwani** |
| RQ-2250 System shall reassess ‘LM/AR Validation’ and ‘RSG Validation’ indicators upon modification of RAFM Project in status “In-Review”. | | | |
| Upon modification of the Standard Project in status ”In-Review”:  the ‘LM/AR Validation ‘indicator on the ‘Entity set’ screen  the ‘LM/AR Validation’ /‘RSG Validation’ indicator assumption set screen  will be required to be refreshed where the modified ‘Standard’ RAFM project requires to be assessed.  Upon modification of the Merged Project in status ”In-Review”:   * The ‘LM/AR Validation’ and ‘RSG Validation’ indicator on the assumption set screen will be refreshed where the modified Merged RAFM project requires to be assessed.   *Refer RQ-2072 Use case ASM001 “View Assumption Set” for more details*. | **Status** | **:** | **Open** |
| **Type** | **:** | **Functional** |
| **Phase** | **:** | **6.8.0.0** |
| **From** | **:** | **Irram Sherwani** |

## RAFM006 Delete RAFM Project

RAFM Projects

### Use Case

|  |  |
| --- | --- |
| **Constraints** | |
| Pre-condition | User should have necessary role(s) and permission(s) |
| Pre-condition | Base Engine RAFM Project has status “In-Review”. |
| Post-condition | The selected RAFM Project (all its versions) is deleted |

|  |  |
| --- | --- |
| **Scenarios** | |
| **Basic Path** |  |
| Basic Path | 1. User selects a version of a RAFM Project in the RAFM Project summary table.  2. User selects the option “Delete” under “Maintenance” menu.  3. The system displays confirmation message asking user to confirm.  4. User selects the button “OK”.  5. The system removes all versions of the selected RAFM Project from the data base and updates the RAFM Project summary and events tables. |
| **Alternate 4.1** |  |
| Alternate | If user selects the button “Cancel”, the use case is aborted. |

### Requirements

|  |  |  |  |
| --- | --- | --- | --- |
| **RQ-2090 The system shall allow a user to delete a RAFM Project** | | | |
| The system shall allow a user to delete a RAFM Project if:    -In case of Base Engine RAFM project:  ‘Base Engine’ is not assigned to scenario set with any status;  ‘Base Engine’ is not embedded within a ‘Standard RAFM Project’ or a ‘Merged RAFM Project’;  - In case of Standard ICM RAFM project:   1. codebases of Standard ICM RAFM project are not used in any of the Lite Models or Aggregation rules. 2. Standard ICM RAFM Project is not included in a run on an assumption set regardless of its run status of “validated” or “invalidated”.   This validation should only be checked if the first one (point no.1) is passed successfully.  -In case of Merged Projects:   * it is not assigned to Assumption Set. | **Status** | **:** | **Open** |
| **Type** | **:** | **Functional** |
| **Phase** | **:** | **6.0.0.0** |
| **From** | **:** | **Irram Sherwani** |
| RQ-2091 User should be able to delete RAFM Project in the system if user has necessary Role(s)/Permission(s) | | | |
| Please refer to [Appendix: BP009 User manager /User Roles and permissions](https://secondfloorr.sharepoint.com/sites/Prudential/Shared%20Documents/eFrame-projects/ICM-projects/1.%20GHO/05%20-%20Design/Release%206.0/Archive%20-%20Old%20Versions/ICM_Replacement_Phase_1_Release_1v1.0.docx) for more details | **Status** | **:** | **Open** |
| **Type** | **:** | **Functional** |
| **Phase** | **:** | **6.0.0.0** |
| **From** | **:** | **Irram Sherwani** |

## RAFM010 Copy RAFM Projects

### Use Case

|  |  |
| --- | --- |
| **Constraints** | |
| Pre-condition | User belongs to the user group that owns the RAFM Project or to a geography to which the RAFM Project has been manually shared with. |
| Pre-condition | User should have necessary role(s) and permission(s). |
| Pre-condition | The status of RAFM Project version to be copied is ‘Validated’. |
| Post-condition | The RAFM Project is copied or and is available in the summary table |

|  |  |  |
| --- | --- | --- |
| **Scenarios** | | |
| **Basic Path** | |  |
| Basic Path | | 1. User selects a version of a RAFM Project in the summary table. 2. User selects the option “Copy” under “Maintenance” menu 3. The system shows a pop up window “Copy RAFM Project” containing field:  * Name: mandatory text box field;  1. User fills in the requested field. 2. User selects the button “Save” on the “Copy RAFM Project” dialog. 3. System creates a copy of RAFM Project version with status “In-Review” and version number 1.0. 4. The system updates the RAFM Project summary screen. 5. The system shows a confirmation message on the screen.   ***“<RAFM Project>\_<version no.> is successfully copied”.*** |
| **Alternate 5.1** | |  |
| Alternate | | If user selects the button “Cancel”, use case is aborted. |
| **Alternate 5.2** |  |
| Alternate | If the name entered by the user is not unique (i.e. already existing in system), system throws an error message and use case is aborted |

### Requirements

|  |  |  |  |
| --- | --- | --- | --- |
| **RQ-2049 The system shall allow a user to copy any version of a Validated status RAFM Project** | | | |
|  | **Status** | : | **Closed** |
| **Type** | : | **Functional** |
| **Phase** | : | **6.1.0.0** |
| **From** | : | **Irram Sherwani** |

## RAFM007 Download RAFM Project

RAFM Projects

### Use Case

|  |  |
| --- | --- |
| **Constraints** | |
| Pre-condition | User should have necessary role(s) and permission(s) |
| Post-condition | The project package of RAFM project in .zip format is available for user. |

|  |  |
| --- | --- |
| **Scenarios** | |
| **Basic Path** |  |
| Basic Path | 1. User selects a version of a RAFM Project in the RAFM Project summary table.  2. User selects the “Download” menu button.  3. The system displays “Download Project File” option under download menu.  4. User select the Option “Download Project File”  5. The RAFM Project .zip file is downloaded and is available to user |
| **Alternate 4.1** |  |
| Alternate | If user selects the button “Cancel” the use case is aborted. |

### Requirements

|  |  |  |  |
| --- | --- | --- | --- |
| **RQ-2038 The system shall allow a user to download the RAFM Project** | | | |
| The system shall provide user with the .zip file in the same structure it has been uploaded.  File name of the downloaded archive shall be named using the given name on the RAFM Project with the version number that has been applied by the ICM | **Status** | : | **Open** |
| **Priority** | : | **Functional** |
| **Phase** | : | **6.0.0.0** |
| **From** | : | **Irram Sherwani** |
| **RQ-2039 User should be able to download the RAFM Project in the system if user has necessary Role(s)/Permission(s)** | | | |
| Please refer to Appendix:” BP009 User Manager: User Roles Permissions” for more details on relevant roles and permissions | **Status** | **:** | **Open** |
| **Type** | **:** | **Functional** |
| **Phase** | **:** | **6.1.0.0** |
| **From** | **:** | **Irram Sherwani** |

## RAFM008 Filter RAFM Project

RAFM Projects

### Use Case

|  |  |
| --- | --- |
| **Constraints** | |
| Pre-condition | The RAFM Project's summary table is at least populated with one RAFM Project. |
| Pre-condition | Filters are enabled (column filters). |
| Post-condition | Only the versions of RAFM Project that match the filters applied are displayed in the RAFM Project summary table. |

|  |  |
| --- | --- |
| **Scenarios** | |
| **Basic Path** |  |
| Basic Path | 1. The system displays the RAFM Project summary table.  2. User selects the option “filter”.  3. The system displays a pop up window containing the applicable filters.  4. User enters the requested filters.  5. User selects the button “Save”.  6. The system applies all the selected filters to the RAFM Project in the summary table and displays only the versions of the RAFM Project that match the selected filters. |
| **Alternate 5.1** |  |
| Alternate | If user selects the button “Clear” the selected filters are reset and the system shows the complete sample of RAFM Project available for user. |
| **Alternate 5.2** |  |
| Alternate | If user selects the button “Cancel” the use case is aborted. |

### Requirements

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| RQ-2040 The system shall allow a user to filter the RAFM Project by "Name". | | | | |
| The system shall allow the following filters:  - Includes: free text. The system will search for the exact string to be included in the RAFM Project name.  - Excludes: free text box. | | **Status** | **:** | **Open** |
| **Type** | **:** | **Functional** |
| **Phase** | **:** | **6.0.0.0** |
| **From** | **:** | **Irram Sherwani** |
| **RQ-2041 The system shall allow a user to filter the RAFM Project by Description** | | | | |
| The system shall allow the following filters:  - Includes: free text. The system will search for the exact string to be included in the Description.  - Excludes: free text box. | | **Status** | **:** | **Open** |
| **Type** | **:** | **Functional** |
| **Phase** | **:** | **6.0.0.0** |
| **From** | **:** | **Irram Sherwani** |
| **RQ-2042 The system shall allow a user to filter the RAFM Project by Last modification date** | | | | |
| The system shall show the following:  - Start date.  - End Date. | | **Status** | **:** | **Open** |
| **Type** | **:** | **Functional** |
| **Phase** | **:** | **6.0.0.0** |
| **From** | **:** | **Irram Sherwani** |
| **RQ-2043 The system shall allow a user to filter the RAFM Projects by Last modified user** | | | | |
| The system shall show the list of users grouped per user groups.  User will be able to select a complete user group or some users within a user group. | | **Status** | **:** | **Open** |
| **Priority** | **:** | **Medium** |
| **Phase** | **:** | **6.0.0.0** |
| **From** | **:** | **Irram Sherwani** |
| **RQ-2045 The system shall allow a user to filter the RAFM Project by version number** | | | | |
| The system shall allow a user to enter a version number, to select the "Highest version" or to select the "most recently created" version.  The highest version of a RAFM Project is considered the version with the highest version number.  The most recently created version is the version with the latest date created. | | **Status** | **:** | **Open** |
| **Priority** | **:** | **Medium** |
| **Phase** | **:** | **6.0.0.0** |
| **From** | **:** | **Irram Sherwani** |
| **RQ-2046 The system shall allow a user to filter the RAFM Project by RAFM Project Type** | | | | |
| The system shall allow a user to select among the types available in the system. | **Status** | | **:** | **Open** |
| **Priority** | | **:** | **Medium** |
| **Phase** | | **:** | **6.0.0.0** |
| **From** | | **:** | **Irram Sherwani** |
| **RQ-2047 The system shall allow a user to filter the RAFM Project by status** | | | | |
| The system shall allow a user to select among the list of available statuses in the system. | **Status** | | **:** | **Open** |
| **Priority** | | **:** | **Medium** |
| **Phase** | | **:** | **6.0.0.0** |
| **From** | | **:** | **Irram Sherwani** |

## RAFM009 Filter by Tags

RAFM Projects

### Use Case

|  |  |
| --- | --- |
| **Constraints** | |
| Pre-condition | The RAFM Project summary table is at least populated with one RAFM Project. |
| Pre-condition | Filters are enabled. |
| Post-condition | Only the versions of RAFM Project that match the filters applied are displayed in the RAFM Project summary table. |

|  |  |
| --- | --- |
| **Scenarios** | |
| **Basic Path** |  |
| Basic Path | 1. The system displays the “Selected Tags” summary table.  2. User selects the option “Filter by Tags”.  3. The system displays a pop up window containing the applicable filters for “Default Tags”, “Descriptive Tags” and “Prescriptive Tags”.  4. User selects/deselects tags to be included /excluded from the summary table view. *(de-select applicable if tags are already selected).*  5. User selects the button “OK”  6. The system applies all the selected filters to the RAFM Project in the library and displays only the versions of the RAFM Projects that match the selected filters. |
| **Alternate 5.1** |  |
| Alternate | If user selects the button “Deselect All” the selected filters are deleted and the system shows the complete sample of RAFM Project available for user.  The “Deselect All” button label changes to “Select All” and user can use this button for selecting all the tags for filters. |
| **Alternate 5.2** |  |
| Alternate | If user selects the button “Cancel” the use case is aborted. |

### Requirements

|  |  |  |  |
| --- | --- | --- | --- |
| **RQ-2048 the system shall display only those RAFM Project versions in the summary table that match the selected tag filters.** | | | |
| The system shall allow a user to filter the RAFM Projects in the summary table of RAFM Projects by "Selected tags".  The system applies all the selected filters to the RAFM Project in the library and displays only the versions of the RAFM Projects that match the selected filters. | **Status** | **:** | **Open** |
| **Type** | **:** | **Functional** |
| **Phase** | **:** | **6.0.0.0** |
| **From** | **:** | **Irram Sherwani** |

## RAFM010 Copy RAFM Projects

RAFM Projects

### Use Case

|  |  |
| --- | --- |
| **Constrains** | |
| Pre-condition | User belongs to the user group that owns the RAFM Project or to a geography to which the RAFM Project has been manually shared with. |
| Pre-condition | User should have necessary role(s) and permission(s). |
| Post-condition | The RAFM Project is copied or and is available in the summary table |

|  |  |  |
| --- | --- | --- |
| **Scenarios** | | |
| **Basic Path** | |  |
| Basic Path | | 1. User selects a version of a RAFM Project in the summary table. 2. User selects the option “Copy” under “Maintenance” menu 3. The system shows a pop up window “Copy RAFM Project” containing field:  * Name: mandatory text box field;  1. User fills in the requested field. 2. User selects the button “Save” on the “Copy RAFM Project” dialog. 3. System creates a copy of RAFM Project version with status “In-Review” and version number 1.0. 4. The system updates the RAFM Project summary screen. 5. The system shows a confirmation message on the screen.   ***“<RAFM Project>\_<version no.> is successfully copied”.*** |
| **Alternate 5.1** | |  |
| Alternate | | If user selects the button “Cancel”, use case is aborted. |
| **Alternate 5.2** |  |
| Alternate | If the name entered by the user is not unique (i.e. already existing in system), system throws an error message and use case is aborted |

### Requirements

|  |  |  |  |
| --- | --- | --- | --- |
| **RQ-2049 The system shall allow a user to copy any version of a RAFM Project** | | | |
|  | **Status** | : | **Closed** |
| **Type** | : | **Functional** |
| **Phase** | : | **6.1.0.0** |
| **From** | : | **Irram Sherwani** |

## RAFM012 Rename RAFM Projects

RAFM Projects

### Use Case

|  |  |
| --- | --- |
| **Constraints** | |
| Pre-condition | User has necessary permissions |
| Pre-condition | The user belongs to the owner user group. |
| Pre-condition | Status of all versions of the RAFM project are in “In-Review”. |
| Post-condition | The selected RAFM Project (all its versions) is renamed |

|  |  |
| --- | --- |
| **Scenarios** | |
| **Basic Path** |  |
| Basic Path | 1. The user selects a version of a RAFM Project in the RAFM Projects summary table.  2. The user selects the option "rename"  3. The system shows a pop up window containing:      - name: text field to enter the name for the RAFM Projects.  4. The user selects the "rename" button.  5. The system updates the name of RAFM project in the summary table. |
| **Alternate 4.1** |  |
| Alternate | If the user selects the button "cancel" the user case is aborted. |
| **Alternate 5.1** |  |
| Alternate | The system shows an error message if the name assigned to the RAFM Project is not unique. |

**Requirements**

|  |  |  |  |
| --- | --- | --- | --- |
| ****RQ-2130 The system shall allow the user to rename a RAFM Project.**** | | | |
| The name is a mandatory field and must be unique. | Status | : | Closed |
| Type | : | Functional |
| Phase | : | 6.1.0.0 |
| From | : | Irram Sherwani |

Appendix

1. Code Blocks BU Bridge Aggregator codebase

When comparing a Merged and a Standard project the order of the LM/AR codebases is not relevant. For RAFM Projects that contain codebases as separate \*.cpp files, to ensure the order of LM/AR codebases are ignored the system use a specific block of code, referred to as modelNameNoSub’:

if (modelNameNosub() == "icm|aggregate|gho" && sm\_gho\_lm\_9.ms\_ReadArray()) {

if (iterationLoopRun) // don't read array next time

sm\_gho\_lm\_9.ms\_ReadArray(false);

sm\_gho\_lm\_9.setModelPointFileId(ms\_gho\_lm\_9\_readfile, "keyfield");

sm\_gho\_lm\_9.setSearchFields(sm\_gho\_lm\_9.getPersistentObject()->SearchFields);

sm\_gho\_lm\_9.resize(0);

if(sm\_gho\_lm\_9.gotoRec(1)){

sm\_gho\_lm\_9.seekFile(xstring(gho->lm\_lookup\_lm\_converted9).strip(-1));

sm\_gho\_lm\_9.readFile(xstring(gho->lm\_lookup\_lm\_converted9).strip(-1));

}

For RAFM Projects that contain codebases in a codemetadata.json file, ‘modelNameNosub’ block of code is no longer required.

1. WDF Versions of the RAFM Project

Over time, RAFM projects evolve and provide new features and options. To support backward compatibility, the system supports all WDF versions of RAFM projects. This allows new RAFM projects with new features as well as old RAFM projects (with limited new features) to be used in parallel. Below you will find all currently available RAFM project WDF versions with a list of their available features.

RAFM projects uploaded into ICM has property - WDF Version. This property specifies the WDF format of the RAFM project that needs to be used while generating WDF files for RAFM runs.

The value of this property is extracted from the (base engine contained in the) RAFM project being uploaded by a special keyword present in ICM codebase, using the following logic:

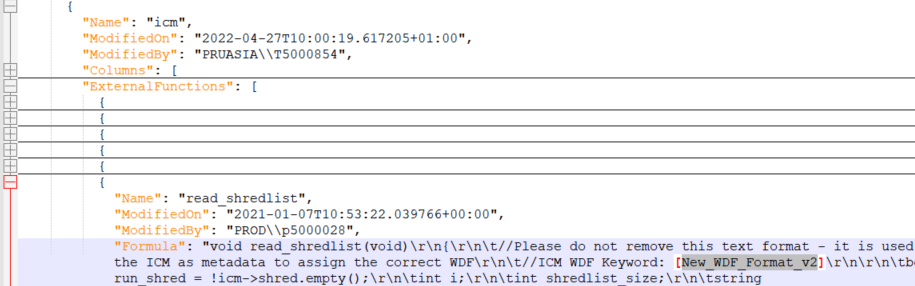
* If ICM codebase contains a keyword with the following structure: [**New\_WDF\_Format\_vXXX**] then the WDF version of the RAFM project will be **XXX**.
* If there is no such keyword that means that the project is using the old format so it will get a WDF version value **0**.

|  |  |
| --- | --- |
| WDF Version 0 | The version of the RAFM project that supports the basic set of functions described in BPs. Its functionality is taken as the basis. |
| WDF Version 1 | RAFM projects WDF v1 support all features of the RAFM projects WDF v0, and next new features: -support runs with multiple shreds |
| WDF Version 2 | RAFM projects WDF v2 support all features of the RAFM projects WDF v1, and next new features: -new RAFM manifest file  -multiple confidence intervals support -shred support for CS batch runs |

Graphical user interface, text, application, email

Description automatically generated

Example of icm model (part of icm.cpp)



Example of icm model (part of codemetadata.json)

1. Identification of codebases and consistency check based on codemetadata.json

For RAFM Projects that contain codebases in a codemetadata.json file, the system shall identificate each codebase, create piece of text for each codebase and use this text as a basis for codebases consistency check and Code Pannel.

The system shall try to find codemetadata.json file in the RAFM project folder (\Obj\x64\Release\codemetadata.json). lf such file exists, the system shall use it for identification of all codebases. (if the file is missing, the system shall identify codebases from the relevant \*.cpp files).

Required information shall be taken from the CodeManager -> ModelClasses in the codemetadata.json file.

The strings contained in the “Names” of the items in the CodeManager->ModelClasses correspond to the name of a codebase, i.e. a codebase will be created for each entry in the ModelClasses section.

The following text shall be created for each codebase entry that corresponds to lite models and aggregation rules, this is held under “CodeManager” element, with sub element “ModelClasses”:

* For each entry in the “ExternalFunctions” (if any exist) concatenate the “Name” entry followed by the entry from the “Formula” attribute
* For each entry in the “Scalars” (if any exist) concatenate the “Name” entry and the value from the “Formula” attribute.
* For each entry in the “TempTables” (if any exist) concatenate the “Name” entry and the value from the “Formula” attribute.
* For each entry in the “Variables” (if any exist) concatenate the “Name” entries.
* For each entry in the “Columns” element (if any exist) concatenate “Name” entry and the value from the “Formula” attribute.

Between each section the system shall insert a heading to indicate which section is concatenated, e.g. “Columns section”

Entries within the .json file are arranged alphabetically, firstly by modelclass name and then by Column, ExternalFunction, Scalar, TempTables and Variable names. The system shall preserve the alphabetical order within each modelclass codebase when creating the concatenated text.

The combined text for each codebase then is a basis for codebase consistency check and content for displaying in the Code Panel.

1. Identification of codebases and consistency check based on cpp files

For RAFM Projects that contain codebases in cpp files, the system shall ignore the specific lines in the cpp files to apply the functional consistency check:

* ignore all the lines which start with string //MODELCLASS;
* ignore all the lines which start with string #include “ModelClass\;
* ignore all the lines which exists in the constructor of the submodel class – it will start with //constructor begincolumn and ends with //constructor End@2;
* ignore copy\_names() function. This function always appears after the constructor and ends with //copy\_names END@2.

The filtered codebase data then is a basis for codebase consistency check.

Note: the system shall not filter cpp codebase content to display it on the Code Panel.