IBM Cúram Social Program Management
8.0.2

*Funded Program Management Developer
Guide*

IBM

Note

Before using this information and the product it supports, read the information in "Notices" on page 6

Contents

Tables

Chapter 1. Developing with Funded Program Management

Use this information to customize Cúram Funded Program Management. Service layer interfaces that are designed for customization are available. A new custom implementation can be provided for any of these interfaces. Events that allow custom logic to be added to Funded Program Management processing are available.

This information describes the customization or extension points provided with the FPM component. For more information about the classes, interfaces, business event interfaces and the standard infrastructural persistence events that are included in the FPM component, see the FPM JavaDoc.

Related concepts

Developing with the Persistence Infrastructure

Cúram Express Rules Reference

Cúram Funded Program Management Guide

Using Strategy Patterns to Customize FPM

An overview of the types of customization points that are available in FPM and how agencies can use them in a supported manner.

FPM provides service layer interfaces that are specifically designed for customization. If you want to use a different strategy than the default implementation, you can provide a new custom implementation for any of these interfaces. Replace the default implementation of the following interfaces, if required, by creating a new Google Guice module class and adding a corresponding entry in the MODULE table.

For more information about Guice and how to use Guice bindings using a Module class, see Developing with the Persistence Infrastructure.

Determining a Fund Configuration

The FundConfigurationDetermination interface is used to determine a fund configuration for a program fund.

To change the way the fund configuration for a program fund is determined, a new implementation of this interface is required. For example, you can use a different fund configuration than the fund configuration that is active on the creation date of the obligation to determine the fund configuration. For example, if an agency wants to change the logic for applying the obligation to the fund configuration that is active during the service period applicable to the obligation, then the agency can implement this interface to have their own logic to determine the fund configuration.

FundConfigurationDeterminationImpl provides the default implementation and determines the fund configuration that is active on the creation date when creating an obligation.

Interface Location

curam.fundpm.util.impl.FundConfigurationDetermination

Calculating the Obligation Amount

FPM provides an interface that is used to calculate an obligation amount for each fund fiscal year, when the period for which the obligation is to be created crosses more than one fiscal year.

To change the mechanism used to calculate an obligation amount for each fund fiscal year, a new implementation of this interface is required.

Default Implementation

FundFiscalYearObligationAmtCalculationImpl provides the default implementation. It calculates the obligation amount for each fund fiscal year, when the obligation period overlaps more than one fiscal year.

To determine the obligation amount, the default implementation splits the obligation amount and the relevant amounts are applied to the respective fund fiscal years. The calculation of obligation amounts for all overlapping fund fiscal years is determined using the formula, OA = (CYD / OD) * ROA, Where: OA = The obligation amount for each fund fiscal year, CYD = The number of days in the obligation period that fall within the current fiscal year, OD = The number of days in the obligation period, ROA = The total requested obligation amount.

Interface Location

curam.fundpm.util.impl.FundFiscalYearObligationAmtCalculation

Generating the Fund Fiscal Year Line Item Reference Number

FPM provides the interface FundFiscalYearLineItemRefNum that is used to generate a reference number for a fund fiscal year line item.

A new implementation of this interface is required to change the mechanism used to generate the reference numbers.

Default Implementation

FundFiscalYearLineItemRefNumImpl provides the default implementation. The default fund fiscal year line item reference number generation mechanism creates a reference number starting from 1024 and increments it by 1.

Interface Location

curam.fundpm.util.impl.FundFiscalYearLineItemRefNum

Using Events to Add Custom Processing to FPM

Use events to add functionality before or after a specific functionality is executed.

Each service layer API of the FPM raises pre and post events. The events can be used to customize the service layer APIs. The service layer events are categorized as follows.

Standard Persistence Events (insert, modify, cancel and read)
For steps to handle these APIs, see Writing listeners for automatic persistence events.

Business Events
For steps to handle these APIs, see Create an event listener and Configure Guice.

For more information about the events that are listed, see the Javadoc for the relevant class.

Events Reference

Use events to add functionality before or after a specific functionality is executed.

Each service layer API of the FPM raises pre and post events. The events can be used to customize the service layer APIs. The service layer events are categorized as follows.

Standard Persistence Events (insert, modify, cancel and read)
For steps to handle these APIs, see Writing listeners for automatic persistence events.

Business Events
For steps to handle these APIs, see Create an event listener and Configure Guice.

For more information about the events that are listed, see the Javadoc for the relevant class.

Fund User Role

The event CheckFundOwnerEvents is raised when fund ownership is checked.

Interface Location

curam.fundpm.funduserrole.impl.CheckFundOwnerEvents

Event Interfaces

Table 1. Fund User Role Event Details		
Name	**Pre Event Data**	**Post Event Data**
CheckFundOwnerEvents	Fund user role accessor reference, program fund, organization object reference, type, and username.	Fund user role accessor reference, program fund, organization object reference, type, username, and fund owner indicator.

Fund Fiscal Year

The event FundFiscalYearCanObligateEvents is raised to check whether obligation can occur before it is performed.

Interface Location

curam.fundpm.impl.FundFiscalYearCanObligateEvents

Event Interfaces

Table 2. Fund Fiscal Year Event Details		
Name	**Pre Event Data**	**Post Event Data**
FundFiscalYearCanObligateEvents	Fund fiscal year accessor reference and amount to be obligated.	Fund fiscal year accessor reference, amount to be obligated and the obligation result indicator.

Fund Fiscal Year

The event FundFiscalYearObligateAmountEvents is raised when an obligation is performed.

Interface Location

curam.fundpm.impl.FundFiscalYearObligateAmountEvents

Event Interfaces

Table 3. Fund Fiscal Year Event Details		
Name	**Pre Event Data**	**Post Event Data**
FundFiscalYearObligateAmountEvents	Fund fiscal year accessor reference, amount to be obligated, obligation transaction type, funded item reference and type.	Fund fiscal year accessor reference, amount to be obligated, obligation transaction type, funded item reference and type.

Process Obligation

The event ObligationProcessObligationEvents is raised when an obligation request is processed.

Interface Location

curam.obligation.impl.ObligationProcessObligationEvents

Event Interfaces

Table 4. Obligation Event Details		
Name	**Pre Event Data**	**Post Event Data**
ObligationProcessObligationEvents	Obligation accessor reference, fundRelatedID, fundRelationType, requested obligation amount, obligation period, obligation related ID, type, case participant role ID and wait list expiry days.	Obligation accessor reference, fundRelatedID, fundRelationType, requested obligation amount, obligation period, obligation related ID, type, case participant role ID, wait list expiry days and Obligation result.

Create Obligation

The event ObligationCreateObligationEvents is raised when an obligation is created.

Interface Location

curam.obligation.impl.ObligationCreateObligationEvents

Event Interfaces

Table 5. Obligation Event Details		
Name	**Pre Event Data**	**Post Event Data**
ObligationCreateObligationEvents	Obligation accessor reference, program fund, obligation amount, obligation period, obligation related details, caseparticipant role details, and wait list expiry days.	Obligation accessor reference, program fund, obligation amount, obligation period, obligation related details, caseparticipant role details, wait list expiry days, and the result after obligation creation.

Update Obligation

The event ObligationUpdateObligationEvents is raised when an obligation is updated.

Interface Location

curam.obligation.impl.ObligationUpdateObligationEvents

Event Interfaces

Table 6. Obligation Event Details		
Name	**Pre Event Data**	**Post Event Data**
ObligationUpdateObligationEvents	Obligation accessor reference, obligation amount, obligation period and obligation related details.	Obligation Accessor reference, obligation amount, obligation period, obligation related details and the result after obligation details are updated.

Using the Funding Rule Set to Retrieve Funds

During the obligation process, the first step is to retrieve funds for obligation. A funding rule set is provided with FPM and is used during the retrieve funds for obligation process to get a list of funds based on priority. The rule set can be a classic rule set or a creole rule set.

Funding Rule Details

Table 7. Funding Rule Set Details		
Rule Set	**Rule Set Type**	**Rule Set Source Location**
Funding rule set	Classic Rule Set	EJBServer\components\FundPM\rulesets\SELECTPRODFUNDS.xml
Funding rule set	CER Rule Set	EJBServer\components\FundPM\CREOLE_Rule_Sets\SELECTPRODFUNDS.xml

Maintaining the Fund Wait List

A fund wait list is maintained as part of Funded Program Management.

When the *ObligateFund API* is called for processing an allocation request, there is a possibility that none of the eligible funds have sufficient budgets available for allocation. In such a case, the Funded Program Management system creates wait list entries in the *WaitlistEntry* entity.

First, the system checks if each of the funds in the list have entries in the *Waitlist* entity. If they are not there, it creates an entry. Then the system creates the entries in *WaitlistEntry*. Priority and Position are considered when creating the entries. The environment variable ENV_FUNDPM_WAIT_LIST_EXPIRY_DAYS can be used to configure the number of wait list expiry days for the wait list to create during obligation creation.

Configuring Batch Processes

The following batch process provides Funded Program Management functionality and can be configured in the system administration application.

ModifyFundFiscalYearStatus

This batch process modifies the status of all the fund fiscal years to INACTIVE for the fiscal years whose end date is before the batch processing date. If the processing date is not entered by the user, the current system date is used as the processing date.

Run this batch process daily to make sure that the fund fiscal year status is accurate.

Notices

This information was developed for products and services offered in the United States.

IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not grant you any license to these patents. You can send license inquiries, in writing, to:

IBM Director of Licensing IBM Corporation North Castle Drive, MD-NC119 Armonk, NY 10504-1785 US

For license inquiries regarding double-byte character set (DBCS) information, contact the IBM Intellectual Property Department in your country or send inquiries, in writing, to:

Intellectual Property Licensing Legal and Intellectual Property Law IBM Japan Ltd. 19-21, Nihonbashi-Hakozakicho, Chuo-ku Tokyo 103-8510, Japan

INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some jurisdictions do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-IBM websites are provided for convenience only and do not in any manner serve as an endorsement of those websites. The materials at those websites are not part of the materials for this IBM product and use of those websites is at your own risk.

IBM may use or distribute any of the information you provide in any way it believes appropriate without incurring any obligation to you.

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact:

IBM Director of Licensing IBM Corporation North Castle Drive, MD-NC119 Armonk, NY 10504-1785 US

Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

The licensed program described in this document and all licensed material available for it are provided by IBM under terms of the IBM Customer Agreement, IBM International Program License Agreement or any equivalent agreement between us.

The performance data and client examples cited are presented for illustrative purposes only. Actual performance results may vary depending on specific configurations and operating conditions.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBMproducts. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

Statements regarding IBM's future direction or intent are subject to change or withdrawal without notice, and represent goals and objectives only.

All IBM prices shown are IBM's suggested retail prices, are current and are subject to change without notice. Dealer prices may vary.

This information is for planning purposes only. The information herein is subject to change before the products described become available.

This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to actual people or business enterprises is entirely coincidental.

COPYRIGHT LICENSE:

This information contains sample application programs in source language, which illustrate programming techniques on various operating platforms. You may copy, modify, and distribute these sample programs in any form without payment to IBM, for the purposes of developing, using, marketing or distributing application programs conforming to the application programming interface for the operating platform for which the sample programs are written. These examples have not been thoroughly tested under all conditions. IBM, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs. The sample programs are provided "AS IS", without warranty of any kind. IBM shall not be liable for any damages arising out of your use of the sample programs.

Privacy Policy considerations

IBM Software products, including software as a service solutions, ("Software Offerings") may use cookies or other technologies to collect product usage information, to help improve the end user experience, to tailor interactions with the end user or for other purposes. In many cases no personally identifiable information is collected by the Software Offerings. Some of our Software Offerings can help enable you to collect personally identifiable information. If this Software Offering uses cookies to collect personally identifiable information, specific information about this offering's use of cookies is set forth below.

Depending upon the configurations deployed, this Software Offering may use session cookies or other similar technologies that collect each user's name, user name, password, and/or other personally identifiable information for purposes of session management, authentication, enhanced user usability, single sign-on configuration and/or other usage tracking and/or functional purposes. These cookies or other similar technologies cannot be disabled.

If the configurations deployed for this Software Offering provide you as customer the ability to collect personally identifiable information from end users via cookies and other technologies, you should seek your own legal advice about any laws applicable to such data collection, including any requirements for notice and consent.

For more information about the use of various technologies, including cookies, for these purposes, see IBM's Privacy Policy at http://www.ibm.com/privacy and IBM's Online Privacy Statement at http://www.ibm.com/privacy/details the section entitled "Cookies, Web Beacons and Other Technologies" and the "IBM Software Products and Software-as-a-Service Privacy Statement" at http://www.ibm.com/software/info/product-privacy.

Trademarks

IBM, the IBM logo, and ibm.com are trademarks or registered trademarks of International Business Machines Corp., registered in many jurisdictions worldwide. Other product and service names might be trademarks of IBM or other companies. A current list of IBM trademarks is available on the Web at " Copyright and trademark information " at http://www.ibm.com/legal/copytrade.shtml.

Adobe, the Adobe logo, PostScript, and the PostScript logo are either registered trademarks or trademarks of Adobe Systems Incorporated in the United States, and/or other countries.

IBM ®

Part Number: