IBM Cúram Social Program Management
8.0.2

Curam Wait List Developer Guide

IBM

Note

Before using this information and the product it supports, read the information in “Notices” on page 7

Edition

This edition applies to IBM® Cúram Social Program Management 8.0.0, 8.0.1, and 8.0.2.

Contents

Tables.. iv

Chapter 1. Developing with Wait Lists.. 1

Why Customize a Wait List?.. 1

What is a Wait List?.. 1

Code Package.. 1

Customization Points.. 1

Replacing Implementations.. 1

Events Reference.. 3

Implementing a new Wait List.. 4

Integrating with Other Systems.. 5

Integrating with Cúram Provider Management.. 5

Integrating with Cúram Funded Program Management.. 5

Wait List Batch Jobs.. 5

Configuring Wait List Application Properties.. 6

Configuration of Wait List Review properties.. 6

Notices.. 7

Privacy Policy considerations.. 8

Trademarks.. 8

Tables

1. Event on Wait List Entry Creation....3

2. Event on Wait List Entry Update....3

3. Event on Wait List Entry Allocation.... 3

4. Event on Wait List Entry Removal.... 3

5. Event on Wait List Entry Expiration....4

6. Event on Wait List Entry Review....4

7. Event on Wait List Entry Deferring.... 4

8. Wait List Expiration Batch Job.... 5

Chapter 1. Developing with Wait Lists

Use this information to develop Cúram wait lists through extension points and customizable interfaces that define the default behavior. A wait list is a list of clients who are awaiting the availability of a specified resource. Cúram wait lists provide a lightweight, generic implementation that can be used to meet a wide range of requirements.

This information describes how to design, implement, customize and configure Cúram wait lists to meet these requirements.

Related concepts

Developing batch processes

Developing with the Persistence Infrastructure

Why Customize a Wait List?

Cúram wait lists can be applied to any of the industry segments in the SEM model. Social Enterprise agencies that are servicing these industry segments provide a diverse array of benefits and services to their clients. These services can have different funding sources, and some might be contracted out to third-party providers. Therefore, there are a wide range of ways in which wait lists can be used.

As a result, the purpose of the Cúram wait list functionality is to provide a lightweight, generic implementation, which can be used to meet as many of these needs as possible. Therefore, customizing the wait list to suit the various needs of SEM agencies is vital. To accommodate this, the Cúram wait list includes extension points and customizable interfaces through which the default behavior can be changed.

What is a Wait List?

A wait list is a list of clients who are awaiting the availability of a specified resource. The resources in question are usually required for the delivery of a service, such as the availability of a place in a facility. Where the resource isn't available, a client can be added to wait list, either by allocating a certain position, or adding the client to the end of the list. Once added, a client's position on the wait list can be moved up or down, depending on how pressing the client's need for the resource is.

Code Package

The Cúram wait list is implemented by using the Persistence Infrastructure approach.

This is placed under the following package:

- curam.waitlist.impl

Customization Points

You can replace the default implementation with a new custom implementation and use events to enact custom workflows.

Replacing Implementations

The Persistence Infrastructure uses Google Guice to act as a factory mechanism for interface objects. In general, it is not compliant to change an implementation, which is bound to an interface. If an implementation is replaced in a non-compliant way, difficulties can be encountered in a later upgrade to a newer version.

However, wait list does provide an interface, which allows you to replace the default implementation with a new custom implementation, if required, by creating a new Guice module class and adding a corresponding entry in the MODULE table. For more information about Guice and how to use Guice bindings that use a Module class, see Developing with the Persistence Infrastructure.

Renumbering of Wait List entries

The agency may want to renumber the wait list entries considering the priority of the entry, or on first come first served basis, or by using some other criteria.

Interface location

curam.waitlist.impl.WaitListRenumberLogic

Default implementation

The default implementation renumbers wait list entries in a wait list. The wait list entry position is incremented or decremented if there is an insertion or removal or allocation or expiry of an entry in the list.

When a new wait list entry is created, system increments the position by 1 for all the wait list entries in an 'open' state with a position higher than or equal to the position of the entry to be added to the wait list.

When a wait list entry is canceled or expired, system decrements the position by 1 for all wait list entries in an 'Open' state with a position higher than the position of the entry being removed or expired from the wait list.

The usage of this implementation in operations Create/Modify/Cancel/Expire/Allocate a Wait List Entry in a Wait List is explained here with example.

Create Wait List Entry: When a new wait list entry is created, if a wait list requires renumbering, then system calls wait list renumbering API. Wait list renumbering API increments the position by 1 for all 'Open' state wait list entries with a position higher than or equal to the position of the entry to be added to the wait list. After renumbering the existing wait list entries, system adds the client (that is, a wait list entry) to the wait list as in the specified position.

For example, if there are 10 clients waiting for a resource with position from 1 to 10. If a new wait list entry to be created at position 5. System first trys to empty the position 5 by incrementing position by 1 for all entries, which are greater than or equal to 5 by using wait list renumbering API and then insert the new record with position 5.

Cancel/Expire/Allocate a Wait List Entry: After canceling/Expiring/Allocating a wait list entry, if a wait list requires renumbering then system calls wait list renumbering API. The wait list renumbering API decrements the position by 1 for all 'Open' state wait list entries with a position higher than the position of the removed entry from the wait list.

For example, if there are 10 clients that are waiting for a resource with position from 1 to 10. If a wait list entry (that is, a client) at position 5 is removed from the wait list, then system decrements position by 1 for all entries, which are having the position greater than 5 by using wait list renumbering API.

Update Wait List Entry: If update wait list entry, system first checks whether there is change in the position. If yes then check whether position renumbering is required. If renumbering is required, then system calls wait list renumbering API to reshuffle the wait list entries position.

For example, if there are 10 clients waiting for a resource with position from 1 to 10. When modifying the wait list entry at position 6, if user changed the position value from 6 to 8 then system first decrements the position by 1 for all 'Open' state wait list entries with a position higher than the original position (i.e.6) and then increment the position by 1 for all 'Open' state wait list entries with a position higher than or equal to the newly specified position (that is, 8). This creates an empty block so that the modified wait list entry goes and gets placed there.

For more information, on how to provide a compliant implementation for an interface, see the section 'How to provider a compliant implementation for an interface that is marked as an extension point within the framework' in Cúram Provider Management Developers Guide.

Events Reference

To allow you to write and attach custom workflow, the following events are raised by wait lists.

Event on Wait List Entry Creation

An event is raised when a wait list entry is added to a wait list.

Table 1. Event on Wait List Entry Creation

Event Detail	**Primary Event Data**	
WAITLIST.WAITLISTENTRYCREATED	waitListEntryID	curam.waitlist.impl.WaitListEntryImpl.createWaitListEntry

Event on Wait List Entry Update

An event is raised when a wait list entry is updated.

Table 2. Event on Wait List Entry Update

Event Detail	**Primary Event Data**	**Raised From**
WAITLIST.WAITLISTENTRYUPDATED	waitListEntryID	curam.waitlist.impl.WaitListEntryImpl.modifyWaitListEntry

Event on Wait List Entry Allocation

An event is raised when a wait list entry is allocated.

Table 3. Event on Wait List Entry Allocation

Event Detail	**Primary Event Data**	**Raised From**
WAITLIST.WAITLISTENTRYALLOCATED	waitListEntryID	curam.waitlist.impl.WaitListEntryImpl.allocate

Events on Wait List Entry Removal

An event is raised when a wait list entry is deleted.

Table 4. Event on Wait List Entry Removal

Event Detail	**Primary Event Data**	**Raised From**
WAITLIST.WAITLISTENTRYREMOVED	waitListEntryID	curam.waitlist.impl.WaitListEntryImpl.cancel

Event on Wait List Expiration

An event is raised when a wait list entry is expired. Wait list entry is expired when the expiry date on the wait list entry is crossed. The wait list entry can be primarily expired by using the expire wait list entry batch job.

Table 5. Event on Wait List Entry Expiration

Event Detail	**Primary Event Data**	**Raised From**
WAITLIST.WAITLISTENTRYEXPIRED	waitListEntryID	curam.waitlist.impl.WaitListEntryImpl.expire

Event on Wait List Entry Review

An event is raised when a wait list entry is selected for review. A wait list entry can be selected for review by running the WaitListReview batch job. This results in all wait list entries with a review date on or before the batch processing date being selected for review.

Table 6. Event on Wait List Entry Review

Event Detail	**Primary Event Data**	**Raised From**
WAITLIST.WAITLISTENTRYSELECTEDFORREVIEW	waitListEntryID	curam.core.sl.impl.WaitListReview.processWaitListEntriesDueForReview

Event on Wait List Entry Deferring

An event is raised when the Wait List Entry review is deferred to a future date. This event is also raised when the review date is updated to be a date in the future.

Table 7. Event on Wait List Entry Deferring

Event Detail	**Primary Event Data**	**Raised From**
WAITLIST.WAITLISTENTRYREVIEWDEFERRED	waitListEntryID	curam.waitlist.impl.WaitListEntryImpl.deferReview

Implementing a new Wait List

A customer who wants to create their own type of wait list needs to administratively add a type of wait list. This can be done by adding a custom entry in the WaitListType code table corresponding to the new wait list type or resource type.

The new resource needs to extend the interface `curam.waitlist.impl.Resource` and the corresponding implementation class needs to implement the methods that are defined in the Resource interface. This ensures that the new resource has the necessary methods implemented that are required during creation of a wait list.

The code snippet of the interface, which extends the Resource interface:

```
public interface Provider extends ProviderOrganization,
    Insertable, OptimisticLockModifiable, Resource {
    // method declaration goes here
    }
```

The code snippet of the implementation, which implements the Resource interface:

```
public class ProviderImpl extends
    ProviderOrganizationImpl<ProviderDtls< implements Provider {
      .
      .
        public long getResourceID() {
          return getDtls().providerConcernRoleID;
        }
      .
      .
```

```
        public WAITLISTTYPEEntry getResourceType() {
          return WAITLISTTYPEEntry.PROVIDER;
        }
    }
```

The APIs to add, modify, or cancel a wait list entry, to list the history of wait list entries, or to search a wait list are in the corresponding Javadoc provided with the IBM Cúram Social Program Management Platform.

Integrating with Other Systems

The Cúram wait list is a generic functionality that holds a list of clients who are waiting for a resource to become available. You can integrate this functionality with Curam Provider Management (CPM) and Cúram Funded Program Management (FundedPM).

Integrating with Cúram Provider Management

When searching for providers to deliver services, the most suitable provider may not always have availability. Where no other suitable provider is available, a client may be put on a wait list for a provider.

Caseworkers can use CPM to add a client to a wait list for a provider. A client can be wait-listed to receive both non-placement services and placement services either for the provider or for a specific provider offering.

For a placement service, the available places can be searched for within the required time period. Once it is determined that a place is available for the client, a reservation or placement can be created.

If no place has been specified for a wait-listed client, a place from a list of available places can be selected when creating the reservation or placement. There is no OOTB automated allocation of clients when a place becomes available.

Clients can be wait-listed for multiple providers, but once a reservation or placement is made wait list entries for that client for the other providers can be removed. If there are active reservations for the same or overlapping periods, the reservations will be canceled when a place is allocated to a wait-listed client.

Integrating with Cúram Funded Program Management

Cúram Funded Program Management wait lists allow clients to be placed on a wait list when funds are insufficient. For example, when a client is authorized to receive child care services and in turn, an attempt is made to obligate an amount from the Child Care fund, the client can be placed on a wait list if the child care fund does not have a sufficient amount to cover the obligation.

When funds become available, the requests to obligate funds on behalf of the client for the child care case plan items can be reassessed, and the obligation associated with the fund can be processed.

Wait List Batch Jobs

Two batch jobs provide wait list functionality.

Expire Wait List Entry

Expire Wait List Entry expires any wait list entry for which the expiration date is passed.

Table 8. Wait List Expiration Batch Job

Batch Name	**Implementation Class**
ExpireWaitListEntry	curam.core.impl.ExpireWaitListEntry

Review Wait List Entry

This batch process selects the eligible wait list entries for review and raises a workflow event to generate wait list review reminder task. All wait list entries with a review date on or before the batch processing date have a review reminder task generated. The review date is provided while a wait list entry is created.

If no review date is set for the wait list entry, it is derived by subtracting the Status Review Reminder Period from the expiry date. Status Review Reminder Period is configured in property administration.

Batch Launcher configurations for Wait List Review batch job

Since the Wait List Review batch process raises workflow events, for successful generation of review reminder tasks, you must configure the following batch launcher properties in property administration:

- curam.batchlauncher.dbtojms.enabled (is set to true)
- curam.batchlauncher.dbtojms.notification.host (have appropriate host value set)
- curam.batchlauncher.dbtojms.notification.port (have appropriate port value set)

Configuring Wait List Application Properties

You can configure wait list properties in property administration.

curam.waitlist.statusreviewreminderperiod

This property determines the reminder period for the wait list status review. It is subtracted from the expiry date to get the review date when review date is not specified. The default value is -1, in which case it is not considered.

curam.waitlist.statusreviewintervalperiod

This property determines the number of days between two wait list entry reviews. The default value is zero.

Configuration of Wait List Review properties

curam.waitlist.statusreviewreminderperiod

This property determines the reminder period for the wait list status review. It is subtracted from the expiry date to get the review date when review date is not specified. Default value is set as -1 in which case it is not considered.

curam.waitlist.statusreviewintervalperiod

This property determines the number of days between two wait list entry reviews. Default value is set as zero.

Notices

Privacy Policy considerations

Trademarks

Part Number:

(1P) P/N: