



Cúram 8.2.2

Intake Configuration Guide

Note

Before using this information and the product it supports, read the information in [Notices on page 37](#)

Edition

This edition applies to Cúram 8.2.2.

© Merative US L.P. 2012, 2026

Merative and the Merative Logo are trademarks of Merative US L.P. in the United States and other countries.

Contents

Note.....	iii
Edition.....	v
1 Configuring Intake.....	9
1.1 Configuring Application Cases.....	9
Introduction.....	9
Configuring Application Case Types.....	9
Associating Application Cases with Programs.....	20
Associating Evidence Types with an Application Case.....	20
Configuring PCR for an Application Case.....	20
1.2 Configuring Programs.....	22
Introduction.....	22
Configuring a New Program.....	22
Defining Program Evidence Types.....	28
Reusing a Product Delivery Case.....	28
1.3 Configuring Internal Application Forms.....	29
Introduction.....	29
Defining a Name.....	30
Configuring an Application Script.....	30
Configuring a Submission Script.....	31
Specifying When an Application Script Can Be Submitted.....	32
Configuring Client Registration.....	32
Associating Programs with Application Forms.....	33
1.4 Configuring Person Search and Match.....	33
Introduction.....	33
Defining Person Search Match Criteria.....	34
Person Search and Match System Properties.....	34
Notices.....	37
Privacy policy.....	38
Trademarks.....	38

1 Configuring Intake

Use this information to configure Cúram Intake application triage and screening intake processes. Each of these processes can be used for capturing data about a client when they apply to an organization for benefits and services.

Related information

1.1 Configuring Application Cases

Introduction

Application cases are configured within Universal Access administration. Cúram Intake leverages Universal Access functionality in a number of areas, in particular for the administration of the programs. Cúram Intake allows organizations to define different types of application cases. There are four main aspects to configuring an application case:

- configuring an application case type
- associating programs with application case types
- adding evidence types
- defining Priority, Complexity and Risk (PCR) for an application case

Once the application case type is configured, programs can be associated. Any evidence types required to authorize the program(s) on the application case can also be associated. A Priority, Complexity and Risk (PCR) configuration can be associated with the application case to help the organization to deal appropriately with the application case.

Configuring Application Case Types

You can configure multiple types of application cases to help a client.

You must specify the name of the application case type. A reference can also be specified which allows the application case to be referenced in code.

Defining a Home Page

A home page can be specified such that the default application case home page can be overridden to use a different home page.

Configuring the Evidence Activation Strategy

The administrator can configure the evidence activation strategy for an application case. The available options are:

- **Auto Apply Evidence**
Default option. All evidence changes are automatically applied by the system.

- **Conditionally Auto Apply Evidence**

The system automatically applies evidence during Intake and Advanced Evidence Sharing (AES) only if all mandatory verifications have been satisfied or have a current waiver.

- For evidence that does not meet these conditions, the caseworker must manually apply the changes on the application case.
- See Customizing the Conditional Evidence Activation Strategy for details on how this strategy can be customized.
- See Intake Action Hook for the Conditional Activation Strategy for details on how to alert caseworkers when some evidence remains in-edit following intake.

- **Manual Apply Evidence**

All evidence changes are applied manually by caseworkers. Follows AES standard behaviour which can auto activate evidence.

When updating the evidence activation strategy for an application case type, administrators must assess the impact on existing open application cases. Specifically, if the strategy is changed from Conditionally Auto Apply Evidence or Manual Apply Evidence to Auto Apply Evidence, any evidence currently in an In Edit or Pending Deletion state will remain unchanged. In this scenario, caseworkers will not be able to manually apply updates to the affected evidence records. This restriction may result in validation errors and prevent successful authorization of the application case.

Customizing the Conditional Evidence Activation Strategy

Use the available hooks on the

`curam.commonintake.impl.ACEvidenceActivationEvaluatorHook` interface to add custom logic that prevents evidence activation when the Application Case's evidence activation strategy is set to `CONDITIONALLY` and the evidence has no outstanding mandatory verifications that have not been waived.

About this task

Customers may want to implement custom logic to leave evidence in-edit during Intake and Advanced Evidence Sharing (AES). For example, a specific evidence type may require caseworker review before activation. The custom behavior can be implemented for both Intake and AES by providing an implementation for the `ACEvidenceActivationEvaluatorHook.shouldActivateOnIntake()` and `ACEvidenceActivationEvaluatorHook.shouldValidateOnSharing()` methods.

1. Provide an implementation of the

`curam.commonintake.authorisation.impl.ACEvidenceActivationEvaluatorHook` class.

1. The following code sample shows how to implement the `curam.commonintake.authorisation.impl.shouldActivateOnSharing` class.

```
package curam.commonintake.impl;

import curam.core.struct.CaseKey;
import curam.core.sl.infrastructure.entity.struct.EvidenceDescriptorKey;
import curam.core.sl.infrastructure.struct.EvidenceActivationDetails;
import curam.util.exception.AppException;
import curam.util.exception.InformationalException;

public class CustomerACEvidenceActivationEvaluatorHook
    implements ACEvidenceActivationEvaluatorHook {

    @Override
    public EvidenceActivationDetails shouldActivateOnIntake(
        final EvidenceDescriptorKey evidenceDescriptorKey,
        final CaseKey caseKey)
        throws AppException, InformationalException {

        EvidenceActivationDetails eaDetails = new EvidenceActivationDetails();

        // Add custom code - set eaDetails.activeInd to true to activate, false
        // otherwise

        return eaDetails;
    }

    @Override
    public EvidenceActivationDetails shouldActivateOnSharing(
        final EvidenceDescriptorKey evidenceDescriptorKey,
        final CaseKey caseKey)
        throws AppException, InformationalException {

        EvidenceActivationDetails eaDetails = new EvidenceActivationDetails();

        // Add custom code - set eaDetails.activeInd to true to
        // activate, false otherwise

        return eaDetails;
    }
}
```

2. Bind the implementation in a Guice module.

1. The following code sample shows how to bind the implementation:

```
package curam.commonintake.impl;

import com.google.inject.AbstractModule;

public class Module extends AbstractModule {

    @Override
    protected void configure() {

        bind(ACEvidenceActivationEvaluatorHook.class)
            .to(CustomerACEvidenceActivationEvaluatorHook.class);
    }
}
```

Intake Action Hook for the Conditional Activation Strategy

Use the `curam.commonintake.impl.ACConditionalEvidenceInEditHook` interface to implement logic that runs when evidence remains in-edit after intake processing. This can be used, for example, to notify a caseworker to review the case. The hook point applies only when the evidence activation strategy is **CONDITIONALLY** and intake resilience mode is enabled.

About this task

Customer may want to implement custom processing when an application case has in-edit evidence following intake where the activation strategy is **CONDITIONALLY**. For example, the caseworker may need to follow up on outstanding mandatory verifications that prevented evidence from activating. The custom behaviour can be implemented by providing an implementation for the `curam.commonintake.impl.ACConditionalEvidenceInEditHook.processInEditEvidenceNotification` method.

1. Provide an implementation of the

`curam.commonintake.impl.ACConditionalEvidenceInEditHook` class.

1. The following code sample shows how to implement the

`curam.commonintake.impl.ACConditionalEvidenceInEditHook` class.

```
package curam.commonintake.impl;
import curam.core.sl.infrastructure.struct.EIEvidenceKeyList;
import curam.util.exception.AppException;
import curam.util.exception.InformationalException;
public class CustomerACConditionalEvidenceInEditHook implements
    ACConditionalEvidenceInEditHook {
    @Override
    public void processInEditEvidenceNotification(final CaseKey caseKey)
        throws InformationalException, AppException {
        // Add custom code..
    }
}
```

2. Bind the implementation in a Guice module.

1. The following code sample shows how to bind the implementation:

```
package curam.commonintake.impl;
import com.google.inject.AbstractModule;

public class Module extends AbstractModule {
    @Override
    protected void configure() {
        bind(ACConditionalEvidenceInEditHook.class)
            .to(CustomerACConditionalEvidenceInEditHook.class);
    }
}
```

Defining Eligibility Check Processing

The administrator has the option to enable eligibility checks for an application case type. If enabled, the worker is provided with the option to check eligibility for the program(s) on an application case.

The administrator must specify a Cúram Express Rules (CER) ruleset. The ruleset outputs the eligibility decision for the program(s) checked on the application case, and, if relevant, the entitlement for that program.

If an Eligibility Check Ruleset is specified, an Eligibility Check Strategy must also be defined. This can be set to All Programs, Programs Applied For Only, or User Selection. If All Programs

is selected, then eligibility is checked for all programs configured for the application case type. If Programs Applied For Only is selected, then eligibility is checked for the programs associated with the application case only. If User Selection is configured, the worker is presented with a list of the programs configured for the application case type and has the option to select the relevant programs for which eligibility is to be checked.

The administrator may also specify a page on which the results of the eligibility check will be displayed. This page is displayed within the list dropdown panel of the Eligibility Checks list on the application case. If a page is not specified, a message stating that no further information is available is displayed in the list dropdown panel.

Related information

Assigning an Ownership Strategy

Separate ownership strategies can be configured for application cases created internally, and application cases created as a result of an online application. For example, an organization may configure that application cases created internally are assigned directly to the user who created it, whereas application cases created as a result of an online application submitted by a citizen are assigned to a work queue.

For internal and online applications, the administrator can define that the application case is assigned to a specific organization unit, position, user or work queue. For internal applications only, the administrator also has the option to specify that the application case is assigned to the current user, i.e., the user who created the application case.

A system property, Application Case Default Owner, can be set to indicate the user to be set as the owner of an application case if the ownership strategy is not configured.

Evidence Issues Ownership Strategy

An ownership strategy can be configured for evidence validation issues in application cases so that the application case will be assigned to the configured issues owner if there are evidence validation issues. The administrator can define that the application case is assigned to a specific organization unit, position, user or work queue if there are evidence validation issues with the application case. A task will be generated for the configured issues owner to work on the evidence validation issues in the application case.

This configuration is applicable for application cases created internally and application cases created as a result of an online application.

Defining a Resolution Home Page

A Resolution Home page configuration can be defined that should be used for directing the user to the application case from the task created for the issues owner to work on the evidence validation issues. For example, the configuration can be used to direct the user to the Evidences pages in the application case from the task created so that the user can work on evidence validation issues and resubmit the application case.

Use Resilience System Property

Cúram Intake provides the flexibility where in the application case will go through a resilience flow, where in the case worker can work on evidence validation and verification issues before the application case goes to 'Open' status. A system property, `curam.intake.use.resilience`, will dictate whether the application case goes through the resilience flow.

Configuring Display Options

A number of options are available to an administrator to define the functions available to workers using application case.

Evidence Dashboard

If the Evidence Dashboard indicator is selected, the Evidence Dashboard view will be available to workers on the Evidence tab of the application case. If it is not selected, this view is not available to workers. For more information on Evidence Dashboard, see the *Cúram Evidence Guide*.

Ready for Determination

Ready for Determination is an optional additional application case status, which is useful if an organization has different workers performing different tasks on a single application. For example, an organization may have one group of workers gathering and verifying the information required to authorize the case, and another group workers who review and authorize the programs. Ready for Determination is useful when this type of organization structure is in place, as it allows the organization to re-assign the application case, based on the status of Ready for Determination.

If the Ready for Determination indicator is selected, the worker must manually select to move the application case from a status of Open to a status of Ready for Determination before the programs on the application case can be authorized. A Ready for Determination workflow can also be invoked at this point, if one is configured.

If the Ready for Determination indicator is not selected, the application case transitions from a status of Open directly to a status of Closed.

Related concepts

[Ready for Determination on page 20](#)

Direct Creation

Cúram Intake provides the flexibility for organizations to define whether internal workers can create an application case via an application form, or directly without an application form, or whether both options are available. If the Direct Creation indicator is selected, the worker has the option to create the application case directly, with only the minimum information required to create the application case. If the Direct Creation indicator is not selected, the worker does not have the option to create the application case directly.

Primary Identifier System Property

A system property, `Treat SSN as Primary Alternate ID`, dictates whether the client's social security number (SSN) should be treated as the primary alternate ID during intake. If the property is not enabled, then the system will generate a primary alternate ID for the client and the client's SSN entered in the intake application will be set as a secondary alternate ID for the client.

Related Cases List System Property

Cúram Intake provides a Related Cases list on an application case which allows the worker to view the cases on which any of the clients on the application case are participants. A system property, Display Closed Related Cases, dictates whether the Related Cases list includes cases with a status of 'Closed'.

Add Client System Property

A system property, Client Selection Component Name, dictates whether the Add Client link is displayed on the Client tab within an application case and whether the Add Client action item is displayed in the application case action menu. If this property is set to Common Intake, the Add Client link and action item are displayed. If this property is set to 'HCR', the Add Client link and action item are not displayed on the application case.

Defining the Program Authorization Strategy

Cúram Intake provides a configurable authorization process for application cases. The administrator can define whether authorization takes place at the application level, for all programs applied for, or whether each program must be authorized separately. The administrator can also define an authorization ruleset and specify the ongoing case required. The following sections describe the configuration options for authorization at the application level.

Authorization Strategy

Authorization can be defined at the application level or at the program level. The administrator specifies this by selecting an authorization strategy of Application or Program.

If Application is selected, then all of the programs on the application case are authorized together when the worker selects the Authorize action on the application case tab. The following sections describe the additional configuration settings required when the authorization strategy is Application. These settings are not required when the authorization strategy is Program.

If Program is selected as the authorization strategy, then each program must be authorized separately by the worker. The additional configuration settings required for this strategy are defined at the program level.

Related concepts

[Configuring Programs on page 22](#)

Authorization Ruleset

When the authorization strategy is Application, an authorization CER ruleset must be defined. The ruleset must determine the set of programs to be authorized, and the clients that are eligible for each program. For more information on defining the authorization ruleset, see the *Cúram Intake Developer Guide*.

Integrated Case Strategy

The Integrated Case Strategy is defined in order for the system to identify whether a new or existing integrated case should be used when program authorization is successful. If a new integrated case is created, all of the application case clients are added as case participants to the integrated case. If an existing integrated case is used, any additional clients on the application case are added as case participants to the integrated case. Any evidence captured on the

application case that is also required on the integrated case is copied to the integrated case upon successful authorization. The configuration options for the integrated case strategy are as follows:

- **New**
A new integrated case of the specified type is always created when authorization is successful for one or more programs on an application
- **Existing (Exact Client Match)**
If an integrated case of the specified type exists with the same clients as those present on the application case, the existing case is automatically used. If multiple integrated cases that meet these criteria exist, the worker is presented with a list of the cases and must select one to proceed with the authorization. If no existing cases match the criteria, a new integrated case is automatically created.
- **Existing (Exact Client Match) or New**
If one or more integrated cases of the specified type exist with the same clients as those present on the application case, the user is presented with the option to select an existing case to use as the ongoing case, or to create a new integrated case. If no existing cases match the criteria, a new integrated case is automatically created.
- **Existing (Any Client Match) or New**
If one or more integrated cases of the specified type exist, where any of the clients of the application case are case participants, the user is presented with the option to select an existing case to use as the ongoing case, or to create a new integrated case. If no existing cases match the criteria, a new integrated case is automatically created.

Integrated Case Type

The administrator must specify the type of integrated case to be created and used upon successful program authorization as defined by the Integrated Case strategy above.

Note: If product delivery cases are required by the organization, these are configured at the program level.

Related information

Outcome Plan Strategy

The Outcome Plan Strategy is defined in order for the system to identify whether a new or existing outcome plan should be used when program authorization is successful. If a new outcome plan is created, all of the application case clients are added as case participants to the outcome plan. If an existing outcome plan is used, any additional clients on the application case are added as case participants to the outcome plan. Any evidence captured on the application case that is also required on the outcome plan is copied to the outcome plan upon successful authorization. The configuration options for the outcome plan strategy are as follows:

- **New**
A new outcome plan of the specified type is always created when authorization is successful for one or more programs on an application
- **Existing (Exact Client Match)**
If an outcome plan of the specified type exists with the same clients as those present on the application case, the existing case is automatically used. If multiple outcome plans that meet

these criteria exist, the worker is presented with a list of the cases and must select one to proceed with the authorization. If no existing cases match the criteria, a new outcome plan is automatically created.

- **Existing (Exact Client Match) or New**

If one or more outcome plans of the specified type exist with the same clients as those present on the application case, the user is presented with the option to select an existing case to use as the ongoing case, or to create a new outcome plan. If no existing cases match the criteria, a new outcome plan is automatically created.

- **Existing (Any Client Match) or New**

If one or more outcome plans of the specified type exist, where any of the clients of the application case are case participants, the user is presented with the option to select an existing case to use as the ongoing case, or to create a new outcome plan. If no existing cases match the criteria, a new outcome plan is automatically created.

Outcome Plan Type

The administrator must specify the type of integrated case to be created and used upon successful program authorization as defined by the Outcome Plan strategy above.

Application Level Authorization Notification System Properties

A number of system properties are provided for organizations to specify the notifications sent as part of the authorization process. The following properties are provided for application level authorization. Cúram Intake provides default configurations for these properties out of the box, however an organization can configure its own settings if required.

- **The Authorization Notifications Enabled Indicator**

Indicates whether authorization notifications will be sent. If set to false, no notifications will be sent by the authorization process.

- **The Successful Application Authorization Notification Workflow**

This property stores the name of the notification workflow that will be invoked when application level authorization succeeds for at least one program.

- **The Unsuccessful Application Authorization Notification Workflow**

This property stores the name of the notification workflow that is invoked when application level authorization is not successful for one or more programs on an application case.

Authorization may be unsuccessful if the authorization rules determine that the client is ineligible for the program, or there is insufficient evidence to authorize.

- **The Application Authorization Deferred Process Failed Notification Workflow**

This property stores the name of the notification workflow that is invoked when the application authorization deferred process fails. The deferred process may fail due to a system error, or invalid configuration of the system.

Configuring Authorization Validation

Use the

`curam.commonintake.authorisation.impl.CIAuthorisationValidationHook` interface to implement extra validations at program level authorization and at application level authorization.

About this task

Customers may want to implement custom validations at program level authorization and at application level authorization. For example, a validation that checks a specific evidence type exists or a specific set of evidences exist on the application before allowing authorization to proceed. The custom behavior can be implemented for both program authorization and application authorization by providing an implementation for the `curam.commonintake.authorisation.impl.CIAuthorisationValidationHook.validateProgram` and `curam.commonintake.authorisation.impl.CIAuthorisationValidationHook.validateApplication` methods.

1. Provide an implementation of the

`curam.commonintake.authorisation.impl.CIAuthorisationValidationHook` class.

1. The following code sample shows how to implement the

`curam.commonintake.authorisation.impl.CIAuthorisationValidationHook` class.

```
package curam.commonintake.authorisation.impl;

import curam.commonintake.authorisation.facade.struct.AuthorisationDetails;
import curam.util.exception.AppException;
import curam.util.exception.InformationalException;

public class CIAuthorisationValidationHookImpl
implements CIAuthorisationValidationHook {

    @Override
    public void validateProgramAuthorisation(final AuthorisationDetails details)
        throws InformationalException, AppException {

        // Add custom code - errors added with
        // ValidationHelper.addValidationError
        // will block authorisation and will be
        // reported to the caseworker.

    }

    @Override
    public void validateApplicationAuthorisation(final AuthorisationDetails
details)
        throws InformationalException, AppException {

        // Add custom code - errors added with
        // ValidationHelper.addValidationError
        // will block authorisation and will be
        // reported to the caseworker

    }
}
```

2. Bind the implementation in a Guice module.

1. The following code sample shows how to bind the implementation:

```
package curam.commonintake.impl;

import com.google.inject.AbstractModule;

public class Module extends AbstractModule {

    @Override
    protected void configure() {

        bind(CIAuthorisationValidationHook.class)
            .to(CIAuthorisationValidationHookImpl.class);
    }
}
```

Specifying Program Appeal Processing

The Appeal All Programs indicator allows an administrator to determine the programs that can be appealed based on the program status. If this indicator is set to Yes then programs with a status of denied, approved and pending can be appealed. If this indicator is set to No then only programs with a status of Denied can be appealed.

Specifying Workflow Processes

Cúram Intake allows organizations to specify a number of workflows which can be invoked as part of application case creation and processing. For more information on defining workflows, see the *Cúram Workflow Reference Guide*. The following sections describe each of the workflows that can be specified.

Application Case Created

The administrator can specify a workflow process to be invoked when the application case is created.

Withdrawal Request

The administrator can specify a workflow process to be invoked in response to an online request by a citizen to withdraw the application for a particular program.

Straight Through

The administrator can specify a workflow process used to drive straight through processing, i.e., from creation of the application case to creation of an ongoing case and closure of the application case. If specified, this process is invoked upon creation of the application case as a result of an online application submission or an internal application form submission. All of the clients on the application case must be successfully matched or registered before the straight through process is invoked.

Note: The straight through process is not invoked when an application case is created directly by a worker as the information captured for direct creation is insufficient.

Cúram Intake provides a system property, Integrated Case Creation Workflow Default Path, which defines the default path for the straight through workflow when the authorization strategy has been configured to allow the selection of an existing integrated case, or the creation of a

new case. The property stores a code from the CaseStrategyWflowConfig code table. The values available are Stop Workflow, Create New Integrated Case and Use Existing Integrated Case. The default setting is Create New Integrated Case. This property is only valid when one existing case exists. If multiple existing cases exist, the straight through process stops and the worker must select which option to proceed with.

Ready for Determination

The administrator can specify a workflow process to be invoked when the application case moves to a status of Ready for Determination. This can only be specified if Ready for Determination is enabled for the application case.

Associating Application Cases with Programs

As part of program configuration, the administrator can select the relevant application case to associate with that program. The administrator can view the programs that are associated with the application case when the programs are configured.

Related concepts

[Defining an Intake Processing System on page 22](#)

An intake processing system must be specified for each program. If a case processing system is not specified, the system is not able to respond to the submission of an application for that program.

Associating Evidence Types with an Application Case

Evidence can be required to authorize one or more of the programs that are associated with the application case. The administrator can associate either dynamic or non-dynamic evidence types to the application case. The evidence can be categorized, given a sort order, and indicated as preferred.

When evidence is associated with an application case, it is automatically enabled for evidence sharing using the Evidence Broker. The administrator must configure the sharing that is required between the application case and other case types, including integrated cases and product deliveries that are created as a result of program authorization. The administrator can also configure sharing of evidence from the application case to the Person/Prospect Person.

Evidence associated with an application case can also be enabled for pre-authorization sharing. This supports the sharing of evidence prior to authorization.

Related concepts

Configuring PCR for an Application Case

Priority, Complexity and Risk (PCR) assessments can be configured for an application case. PCR can be assessed manually by a worker or by a ruleset. The administrator can specify a workflow to be invoked as a result of a PCR assessment on an application case.

The administrator must enter a name for the PCR configuration, and must provide a start date and end date. PCR configurations cannot overlap for an application case.

Defining a PCR Ruleset

If the organization is using a ruleset to assess the PCR rating of an application case, a PCR ruleset must be selected for the PCR configuration. The ruleset must be defined as per the guidelines specified in the *Cúram Intake Developer Guide*. If configured, the ruleset is invoked upon creation of the application case, or in response to one of the configured ruleset execution points. The worker can review the results of the PCR assessment on the application case.

Defining a PCR Workflow

The administrator has the option to select a workflow to be invoked when PCR is assessed on an application case. Only workflows categorized as PCR can be selected for a PCR configuration. For more information on defining workflows, see the *Cúram Workflow Reference Guide*.

Specifying PCR Execution Points

Cúram Intake provides a set of events that can occur on an application case that can invoke the PCR ruleset automatically. The execution points can be optionally selected by the administrator if a PCR ruleset is specified for the PCR configuration.

Note: The PCR ruleset is always invoked automatically upon application case creation. This allows the PCR values to be used in a straight through process if required. This is not an optional execution point.

The specification of the execution points allows an organization to define points in the process where the PCR values will be recalculated. Each of these execution events is described below:

- **User Invoked**
The worker has the option to manually invoke the PCR rule set to calculate the PCR values.
- **Evidence Updated**
The PCR rule set is invoked if evidence is activated on the application case or the user edits or deletes an active evidence record.
- **New Client**
The PCR rule set is invoked when a new client is added to the application case.
- **Client Removal**
The PCR rule set is invoked when a client is removed from the application case.
- **Client Match**
The PCR rule set is invoked when a prospect client is matched to an already registered client.
- **New Program Application**
The PCR rule set is invoked when a program is added to the application case
- **Program Withdrawal**
The PCR rule set is invoked when a program is withdrawn from the application case.
- **Check Eligibility**
The PCR rule set is invoked when an eligibility check is completed for the application case.
- **Program Authorization**
The PCR rule set is invoked when a program is authorized.

Configuring PCR Range Mappings

The ranges for the PCR ratings must be mapped in order for the PCR ruleset to determine the PCR ratings for the application case. The administrator must set a numeric range for each rating. For example the High Priority range may be defined as 70-100, the Medium Priority range may be defined as 30-69 and the Low Priority range may be defined as 0-29. The ruleset then uses these values to determine whether the application case is High, Medium or Low Priority. Range mappings must also be defined for the Complexity and Risk ratings.

1.2 Configuring Programs

Introduction

UA allows agencies to define different types of programs. Once programs have been defined, they can be associated with online and internal applications, and application cases. Programs can be applied for online by citizens in Universal Access, or internally by workers using Cúram Intake. Program applications can then be processed using application cases.

There are four main aspects to configuring a program:

- Configuring programs and associated display and system processing information
- Configuring application information, such as the application case, and whether multiple applications are allowed.
- Associating program-specific evidence types

Configuring a New Program

Programs are configured on the **New Program** page. Details and specifications of the program are required to be defined when the program is created.

The sections that follow detail program configurations.

Defining a Name and Reference

A name and reference must be defined when a new program is created. The name that is defined is displayed in the internal application. The reference is used to reference the program in code.

A name and reference must be defined when creating a new program. The name that is defined is displayed to the citizen in UA and in the internal application. The reference is used to reference the program in code.

Defining an Intake Processing System

An intake processing system must be specified for each program. If a case processing system is not specified, the system is not able to respond to the submission of an application for that program.

Two options are available: **Cúram** or select from pre-configured remote systems. Programs that are processed by using Cúram Intake must have **Cúram** selected as the intake processing system.

If Cúram is specified as the intake system, an application case type must be selected. An application case of the specified type is created in response to a submission of an online application or internal application form for the program. Alternatively, when the application case is created, the worker can select to add the programs configured to use that application case type.

When an application case type is selected, the program can be added manually to that type of application case by a worker in the internal application as part of intake processing. A configuration is provided that dictates if the program is a coverage type. Coverage types are automatically evaluated by program group rules in the context of healthcare reform applications, such as insurance affordability. As such, coverage types cannot be applied for directly by a citizen or manually added to an application case by a worker and authorized. If the program is a coverage type, select the 'Yes' option. The program will be filtered out of the list of programs available to be added to online and internal applications in administration and the list of programs available to be manually added to an application case by a worker. If the program is not a coverage type, select the 'No' option. The program will be available to be manually added to online and internal applications in administration and to an application case by a worker.

An indicator is provided which dictates whether a **Reopen** action is enabled on the programs list on an application case for denied and withdrawn programs of a particular type. A workflow can be specified which is initiated when the program is reopened.

Related concepts

[Configuring Application Cases on page 9](#)

Defining Case Processing Details

A case processing system must be specified for each program.

Two options are available; **Cúram** or select from pre-configured remote systems. Select **Cúram** if the program eligibility is determined and managed by using a Cúram-based system. Select a remote system if eligibility is determined and managed by a non-Cúram-based system.

If **Cúram** is selected as the case processing system, more options are available to allow for program level authorization to be configured. Program level authorization means that if an application case contains multiple programs, each program can be authorized individually, and a separate case is used to manage the clients on an ongoing basis.

Defining the Integrated Case Strategy

The Integrated Case Strategy must be defined in order for the system to identify whether a new or existing integrated case needs to be used when program authorization is successful. The integrated case is used to host any product deliveries created as a result of the authorization.

The Integrated Case Strategy must be defined in order for the system to identify whether a new or existing integrated case should be used when program authorization is successful. The integrated case is used to host any product deliveries created as a result of the authorization. If a new integrated case is created, all of the application case clients are added as case participants to the integrated case. If an existing integrated case is used, any additional clients on the application case are added as case participants to the integrated case. Any evidence captured on the application case that is also required on the integrated case is copied to the integrated case upon successful authorization. The configuration options for the integrated case strategy are as follows:

- **New**
A new integrated case of the specified type is always created when authorization of the program is successful.
- **Existing (Exact Client Match)**
If an integrated case of the specified type exists with the same clients as those cases present on the application case, the existing case is used automatically. If multiple integrated cases that meet these criteria exist, the worker is presented with a list of the cases and must select one to proceed with the authorization. If no existing cases match the criteria, a new integrated case is created.
- **Existing (Exact Client Match) or New**
If one or more integrated cases of the specified type exist with the same clients as those cases present on the application case, the user is presented with the option to select an existing case to use as the ongoing case, or to create a new integrated case. If no existing cases match the criteria, a new integrated case is created.
- **Existing (Any Client Match) or New**
If one or more integrated cases of the specified type exist, where any of the clients of the application case are case participants, the user is presented with the option to select one of the existing cases to use as the ongoing case, or to create a new integrated case. If no existing cases match the criteria, a new integrated case is created.

Specifying the Integrated Case Type

The administrator must specify the type of integrated case to be created/used upon successful program authorization as defined by the Integrated Case strategy listed previously.

Defining the Outcome Plan Strategy

The Outcome Plan Strategy must be defined in order for the system to identify whether a new or existing outcome plan needs to be used when program authorization is successful. If a new outcome plan is created, all of the application case clients are added as case participants to the outcome plan.

If an existing outcome plan is used, any additional clients on the application case are added as case participants to the outcome plan. Any evidence that is captured on the application case that is also required on the outcome plan is copied to the outcome plan upon successful authorization. The configuration options for the outcome plan strategy are as follows:

- **New**
A new outcome plan of the specified type is always created when authorization of the program is successful.
- **Existing (Exact Client Match)**
If an outcome plan of the specified type exists with the same clients as those clients present on the application case, the existing case is automatically used. If multiple outcome plans that meet these criteria exist, the worker is presented with a list of the cases and must select one to proceed with the authorization. If no existing cases match the criteria, a new outcome plan is created.
- **Existing (Exact Client Match) or New**
If one or more outcome plans of the specified type exist with the same clients as those clients present on the application case, the user is presented with the option to select an existing case

to use as the ongoing case, or to create a new outcome plan. If no existing cases match the criteria, a new outcome plan is created.

- **Existing (Any Client Match) or New**

If one or more outcome plans of the specified type exist, where any of the clients of the application case are case participants, the user is presented with the option to select one of the existing cases to use as the ongoing case, or to create a new outcome plan. If no existing cases match the criteria, a new outcome plan is created.

Outcome Plan Type

The administrator must specify the type of integrated case to be created and used upon successful program authorization as defined by the Outcome Plan strategy described previously.

Specifying a Client Selection Strategy

The Client Selection Strategy is used to define how clients are added from the application case to the product delivery created as a result of authorization of a program. If a product delivery type is specified, a client selection strategy must be selected. The configuration options are as follows:

- **All Clients**

All of the application clients are added to the product delivery case. The application case primary client is set as the product delivery primary client. Once the primary client is set, it cannot be changed. All other clients are added to the product delivery as members of the case members group.

- **Rules**

A rule set is used to determine the clients to be added to the product delivery(if a product delivery is configured). At least one client must be determined by the rules for authorization to proceed.

- **User Selection**

The user selects the clients who should be added to the product delivery. The user must select both the primary client and any other clients to be added to the case member group on the product delivery.

Specifying a Client Selection Ruleset

A Client Selection Ruleset must be selected when the Client Selection Strategy is 'Rules'.

Specifying a Product Delivery Type

The Product Delivery Type is used to specify the product delivery used to make a payment to a client(s) in respect of a program. The drop down displays all active products configured on the system.

Note: This field applies to both program and application authorization processing i.e. Program and application authorization can result in the creation of the product delivery type specified.

Submitting a Product Delivery Automatically

The Submit Product Delivery indicator is used to dictate if the product delivery created as a result of program authorization should be submitted automatically for approval. If selected, the

product delivery created as a result of authorization of this program is submitted automatically for a supervisor for approval.

Note: This field applies to both program and application authorization processing i.e. Program and application authorization can result in the automatic submission of a product delivery.

Program Level Authorization Notification System Properties

A number of system properties are provided for organizations to specify the notifications sent as part of the authorization process. Cúram Intake provides default configurations for these properties in its initial configuration, however an organization can configure its own settings if required.

The following properties are provided for program level authorization.

- **The Authorization Notifications Enabled Indicator**
Indicates whether authorization notifications are sent. If set to false, no notifications are sent by the authorization process.
- **The Successful Program Authorization Notification Workflow**
This property stores the name of the notification workflow that is invoked when program-level authorization succeeds.
- **The Unsuccessful Program Authorization Notification Workflow**
This property stores the name of the notification workflow that is started when program-level authorization is not successful. Authorization might be unsuccessful if the client selection rules determine that the client is ineligible for the program, or insufficient evidence exists to authorize.
- **The Program Authorization Deferred Process Failed Notification Workflow**
This property stores the name of the notification workflow that is started when the program authorization deferred process fails. The deferred process might fail due to a system error, or invalid configuration of the system.

Application Level Authorization System Property

A system property is provided for organizations to specify whether the system checks for mandatory outstanding verifications associated with application case evidence on authorization, when application level authorization is configured. By default, the property is set to 'false' and the validation is not enabled, however an organization can enable the validation by updating the property to 'true':

- **Enable mandatory verification validation for application level authorization**
This property determines whether the system checks for mandatory outstanding verifications associated with an application case on application authorization. If set to true, the system prevents authorization if there are mandatory outstanding verifications. If set to false, the system allows authorization regardless of mandatory outstanding verifications.

Configuring Timers

Many organizations impose time limits within which an application for a program must be processed. Cúram provides organizations with the ability to configure application timers for each program.

For example, a government agency may have a requirement that food assistance applications must be authorized within 30 business days of the date of application.

A number of configuration options are available, including the duration of the timer, whether the timer is based on business or calendar days, a warning period, and timer extension and approval. All configuration options are outlined in the following list.

- **Duration**

The length of the timer in days. This value, along with the fields **Start Date** and **Use Business Days** (and the configured business hours for the organization) are used to calculate the expiry date for the timer. This value is used as a number of business days if **Use Business Days** is set. If **Use Business Days** is not set, this value is used as calendar days.

- **Start Date**

Specifies whether the timer needs to start on the application date or the program addition date. The options available are **Application Date** and **Program Addition Date**.

Note: In most cases, these dates are the same (the programs will be added at the same time as the application is made) but when a program is added later to the application, after initial submission, this information might not be the case.

- **Warning Days**

Allows for the specification of a number of warning days. The warning days are used to warn the user that the timer deadline is approaching. If configured, the Warning Reached workflow also is enacted when the warning date is reached and the timer is still running (for example, the program is not completed).

- **End Date Extension Allowed**

An indicator to dictate if a user can extend the timer by a number of days.

- **Extension Approval Required**

An indicator to dictate if a timer extension requires approval from a supervisor. If approval is required for the extension, the case supervisor must review and either approve or reject the extension. After the extension is approved, or if approval is not required, the timer expiry date is updated to reflect the extension.

- **Use Business Days**

An indicator to dictate that the timer should not decrement over non-working days. If this indicator is set, the system uses the **Working Pattern Hours** for the organization to determine the non-working days when it is calculating the expiry date for the timer.

- **Resume Timer**

An indicator to dictate whether the program timer needs to be resumed when the program is reopened.

- **Resume From**

If a timer is resumed, the **Resume From** field dictates the dates from which a program can be resumed. The values include the date that the program was completed (denied or withdrawn), and the date that the program was reopened.

- **Timer Start**
Allows for the specification of a workflow that is enacted when the timer starts.
- **Warning Reached**
Allows for the specification of a workflow that is enacted when the warning period is reached.
- **Deadline Not Achieved**
Allows for the specification of a workflow that is enacted if the timer deadline is not achieved; that is, the program is not being withdrawn, denied, or approved by the timer expiry date.

Configuring Multiple Applications

The Multiple Applications indicator dictates if a client can apply for this program while they have a previous application that is pending for that program.

If set to `true`, the client can have multiple pending applications for the program in question. For example, they can submit an application for this program while they already have a pending application for the program in the system. If it is set to `false`, the caseworker is not allowed to submit an internal application form for this program, or to add a program of that type to an application case while they already have a pending application in the system for that program.

Note: This setting is also applicable to online applications if configured by the organization. For more information on how this setting impacts citizens applying online, see the *Universal Access Configuration Guide*.

Defining Program Evidence Types

Evidence Types can be associated with a program. This can be used by an organization to provide support for applications for multiple programs where a particular program needs to be authorized much more quickly than other programs for which the client may have applied, for example, expedited food assistance. Using this configuration, only the evidence configured for the program is validated on program authorization. Any additional evidence recorded on the application case is not validated at this stage and is not required to be completed to authorize the program. However, if the program being authorized is the final Pending program on the application case, all evidence associated with the application case is validated. This allows benefits for the authorized program to be delivered to the client, while the worker continues to gather the evidence required for the other programs applied for.

Reusing a Product Delivery Case

When a new application is authorized, a new product delivery case is created by default. To reuse an existing product delivery case, a customization point is available that you can implement.

About this task

You can reuse an existing product delivery case by implementing the `curam.commonintake.authorisation.impl.ProductDeliveryReuseStrategy.getProductDeliveryCustomizationPoint`. For more information, see the Javadoc for the associated class.

Note: An existing product delivery case is reusable only if it is a child of the integrated case that is involved in the authorization process. Otherwise, a new product delivery case is created.

Procedure

1. Provide an implementation of the `ProductDeliveryReuseStrategy` class.

The following code shows how to implement the `ProductDeliveryReuseStrategy` class:

```
import curam.commonintake.authorisation.impl.ProductDeliveryReuseStrategy;
import curam.util.exception.AppException;
import curam.util.exception.InformationalException;

public class CustomProductDeliveryReuseStrategy
    implements ProductDeliveryReuseStrategy {

    @Override
    public ProductDelivery getProductDelivery(final long caseID)
        throws AppException, InformationalException {

        // Add custom code...

        return theProductDelivery;
    }
}
```

2. Bind the implementation in a Guice module.

The following code shows how to bind the implementation:

```
import com.google.inject.AbstractModule;

public class Module extends AbstractModule {

    @Override
    protected void configure() {

        bind(ProductDeliveryReuseStrategy.class)
            .to(CustomProductDeliveryReuseStrategy.class);

    }
}
```

1.3 Configuring Internal Application Forms

Introduction

This chapter provides an overview of the configuration settings available for defining internal applications forms in UA administration for Cúram Intake. The organization can define an internal application form for workers to create an application case using an Intelligent Evidence Gathering (IEG) script. There are four main aspects to configuring an application:

- configuring information about an application
- configuring the script and schema used to collect and store the information specified during the application process
- configuring the programs for which an application can be used to apply

This chapter outlines these aspects in more detail.

Defining a Name

The administrator must specify a name for the application form. The name is displayed to the worker if multiple application forms are configured, to allow them to select the application form they wish to proceed with.

Configuring an Application Script

An IEG script must be defined for the application which is used to collect the answers to the application questions. A script name must be specified in the Question Script field. A data store schema must be specified to store the data entered in the script. A schema name must be specified in the Schema field. On saving the application, an empty template for both the script and schema will be created by the system based on the Question Script and Schema specified. It will then be possible to update these from the Internal Application tab by selecting hyperlinks provided on the page. Clicking on the Question Script link will launch the IEG Editor which will allow the question script to be edited. Clicking on the Schema link will launch the Datastore Editor which will allow the schema to be edited. For more information, see the *Cúram Intelligent Evidence Gathering Guide*.

Specifying Finish and Quit Pages for an Internal Application Form

Two fields are available within the IEG Editor which are used to indicate the URL to direct to on exiting or completing an application. The Finish Page field indicates the URL to direct the citizen to on completing the application. The Quit Page field indicates the URL to direct the citizen to on exiting the application.

When configuring the question script for an internal application, these fields must be set to the following values:

- The Finish page field must be set to *CommonIntake_finishInternalIntakeScript*.
- The Quit page field must be set to *CommonIntake_quitInternalIntakeScript*.

Required IEG Attributes for Internal Application Forms

A number of attributes must be specified within the schema when configuring the script for an internal application form.

- An attribute, localID, must be added to the Person entity in the schema for the script. The attribute type must be IEG_INT64
- If the organization wants to capture the application date in the IEG script, a filingDate attribute must be added to the Application entity in the schema. The attribute type must be IEG_DATE.
- If the organization wants to capture the method of application in the IEG script, a methodOfApplication attribute must be added to the Application entity in the schema. The domain for the methodOfApplication attribute should be a code table domain, with the code table set to MethodOfApplication.

Prepopulating Person Information

When the internal application script is started, certain details for the person are prepopulated into the data store that is used by the IEG script. For example, first name, last name, data of birth, and

address. You can implement a customization point to prepopulate extra attributes into the data store.

About this task

You can prepopulate extra attributes into the data store by implementing the `curam.commonintake.facade.impl.ApplicationFormPrepopulationStrategy.prepopulateFrom` customization point. For more information, see the Javadoc for the associated class.

Procedure

1. Provide an implementation of the `ApplicationFormPrepopulationStrategy` class.

The following code shows how to implement the `ApplicationFormPrepopulationStrategy` class:

```
import curam.commonintake.facade.impl.ApplicationFormPrepopulationStrategy;
import curam.datastore.impl.Datastore;
import curam.datastore.impl.Entity;
import curam.util.exception.AppException;
import curam.util.exception.InformationalException;

public class CustomApplicationFormPrepopulationStrategy
    implements ApplicationFormPrepopulationStrategy {

    @Override
    public void prepopulateFromExistingConcernRole(final long concernRoleID,
        final Datastore datastore, final long rootEntityID,
        final Entity personEntity) throws AppException, InformationalException {

        // Add custom code....

    }

}
```

2. Bind the implementation in a Guice module.

The following code shows how to bind the implementation:

```
import com.google.inject.AbstractModule;

public class Module extends AbstractModule {

    @Override
    protected void configure() {

        bind(ApplicationFormPrepopulationStrategy.class)
            .to(CustomApplicationFormPrepopulationStrategy.class);

    }

}
```

Configuring a Submission Script

A submission script can be defined for an internal application form. This is used to define additional information which does not form part of the application script to be captured, for example, an application might require information regarding the citizen's ability to attend an interview. A submission script (IEG) can be specified in the Submission Script field. On saving the application, an empty template for both the submission script and schema will be created

by the system based on the Submission Script specified. It will then be possible to update this from the Application tab by selecting the hyperlinks provided on the page. Clicking on the link will launch the IEG Editor which will allow the submission script to be edited. Clicking on the Schema link will launch the Datastore Editor which will allow the schema to be edited. For more information, see the *Cúram Intelligent Evidence Gathering Guide*.

Specifying Finish and Quit Pages for Submission Scripts

Finish and quit pages must be specified for any submission scripts defined for an internal application form.

When configuring a submission script for an internal application, the Finish page and Quit page fields available in the IEG Editor must be set to the following values:

- The Finish page field must be set to `CommonIntake_finishInternalIntakeSubmissionScript`.
- The Quit page field must be set to `CommonIntake_quitIntakeSubmissionScript`.

Related concepts

[Specifying Finish and Quit Pages for an Internal Application Form on page 30](#)

Specifying When an Application Script Can Be Submitted

The administrator has the option to specify that the application script can be submitted on completion only. If this indicator is enabled, the worker will not have the option to submit the application script until the script has been completed. If this option is not enabled, the worker can select to submit the application script at any time, once the minimum mandatory information is captured.

Configuring Client Registration

The Client Registration field allows an administrator to dictate whether additional clients that are added to a type of application form are registered as prospects or persons.

As the worker creates an internal application form, and an application case directly from a person or prospect, this setting is not used when an internal application form is submitted for only one client or when the application case is created directly. To determine whether to register the client as a prospect or a person, the system checks this setting in the following two scenarios :

- If Person Search and Match is configured and no match can be found for the client
- If Person Search and Match is not configured, i.e., the clients on an application are always registered without the system automatically searching and matching them.

If the Client Registration field is not set, the system checks the system property Register as Prospect Person to identify if a client is registered as a prospect or a person.

Related concepts

[Indicating Whether Clients are Registered as Prospects or Persons on page 35](#)

The `Register as Prospect Person` system property dictates whether clients on an application are registered as prospects or persons. This property is used when additional clients

are captured on an internal application form if *Client Registration* is not defined on the online and internal applications in administration.

Associating Programs with Application Forms

The programs for which the worker can submit the application form must be associated with the configured internal application form. Any configured program can be associated with an internal application form.

When associating programs with an application form, an order can be assigned which dictates the display order of the selected program relative to other programs associated with the internal application form. The ordering is only relevant if multiple programs are associated with the application form. If multiple programs are associated with an internal application form, the worker is presented with a list of the programs in the specified order, and must select which programs to proceed with. If only one program is associated with an application form, the program is selected automatically by the system.

Related concepts

[Configuring Programs on page 22](#)

Defining Mappings for a Program

When an application for a program is processed by Cúram, the information entered in an application is mapped to the application case's evidence tables. The mappings are configured for a particular program by creating a mapping configuration of type Cúram Evidence or by using the Cúram Data Mapping Editor. A mapping configuration must be specified for a program in order for the appropriate evidence entities to be created and populated in response to an internal application form submission.

Related information

1.4 Configuring Person Search and Match

Introduction

The Configuring Person Search and Match section defines the criteria that the system uses to search for and match a client with registered persons and outlines the properties that allow an organization to configure person search and match.

The default implementation of the PersonMatch API returns an empty list. Customers can provide their own implementation of the PersonMatch API if they so wish.

If implemented, person search and match can automatically search for, and either match or register the clients as part of the application case creation process. The search process can return registered persons that meet some or all of the search criteria configured. The match process can determine whether the registered person is a conclusive match with the client, an inconclusive match with the client, or is not a match. This determination is based on the weights that are assigned to the configured search criteria and thresholds that are configured in System Administration.

If no match is found, or if person search and match is not configured, the client on the application form is registered on the system as part of the application case creation process. A system property is provided to dictate whether the client is registered as a prospect, or as a person.

Two main aspects for configuring person search and match are used:

- Configuring person search match criteria
- Configuring the system properties for person search and match

Defining Person Search Match Criteria

The person search match criteria define the data that the system uses to search for and match a client on an application with a registered person or persons.

The criteria can be defined in Participant administration. An organization must decide which search criteria to use, and the weights to assign to each criterion. The weight defines how important the criterion is in determining whether the registered person who is returned by the search matches the client on the application.

The administrator must specify a name for the search criterion. A reference also can be specified that allows the search criterion to be referenced in code. The administrator can also indicate whether the search criterion is enabled for the search and match process. The administrator must specify the weight for the search criterion. These configuration settings are described in more detail in the following sections.

Defining a Weight for a Search Criterion

The organization must decide the weight to be assigned to each search criterion defined, i.e., how important the search criterion is in determining a match for a client. For example, if the organization has defined three search criteria, First Name, Last Name and Date of Birth, First Name may be weighted at 20, Last Name may be weighted at 30, and Date of Birth may be weighted at 50. The weights are used to determine a score for a registered person returned by the search. The score determines whether the person is a conclusive match, an inconclusive match, or not a match, based on the thresholds configured in System Administration.

Person Search and Match System Properties

A number of system properties are provided to allow an organization to configure person search and match. The Person Search and Match System Properties sections describe how to enable person search and match and how to set parameters of these properties.

Setting Inconclusive Match Thresholds

Two properties are provided to define the threshold values for Person Match - `Person Match Upper Inconclusive threshold` and `Person Match Lower Inconclusive threshold`.

A score greater than the Upper Inconclusive Match threshold is considered a Conclusive Match. A score of equal to or between the Lower and Upper Inconclusive Match thresholds is considered an Inconclusive Match. A score of less than the Lower Inconclusive Match threshold is considered No Match.

Setting the Maximum Number of Search Hits

The system property, `Person Match Maximum Hits`, defines the maximum number of search results that can be returned from the person match search.

Indicating Whether Clients are Registered as Prospects or Persons

The `Register as Prospect Person` system property dictates whether clients on an application are registered as prospects or persons. This property is used when additional clients are captured on an internal application form if *Client Registration* is not defined on the online and internal applications in administration.

Note: As the caseworker creates an internal application form, and an application case directly from a person or prospect, this setting is not used when an internal application form is submitted for only one client, or when the application case is created directly.

The system checks this setting in two scenarios to determine whether to register the client as a prospect or a person:

- If Person Search and Match is configured and no match can be found for the client.
- If Person Search and Match is not configured; for example, the clients on an application are always registered without the system automatically searching and matching them.

Notices

Permissions for the use of these publications are granted subject to the following terms and conditions.

Applicability

These terms and conditions are in addition to any terms of use for the Merative website.

Personal use

You may reproduce these publications for your personal, noncommercial use provided that all proprietary notices are preserved. You may not distribute, display or make derivative work of these publications, or any portion thereof, without the express consent of Merative

Commercial use

You may reproduce, distribute and display these publications solely within your enterprise provided that all proprietary notices are preserved. You may not make derivative works of these publications, or reproduce, distribute or display these publications or any portion thereof outside your enterprise, without the express consent of Merative.

Rights

Except as expressly granted in this permission, no other permissions, licenses or rights are granted, either express or implied, to the publications or any information, data, software or other intellectual property contained therein.

Merative reserves the right to withdraw the permissions granted herein whenever, in its discretion, the use of the publications is detrimental to its interest or, as determined by Merative, the above instructions are not being properly followed.

You may not download, export or re-export this information except in full compliance with all applicable laws and regulations, including all United States export laws and regulations.

MERATIVE MAKES NO GUARANTEE ABOUT THE CONTENT OF THESE PUBLICATIONS. THE PUBLICATIONS ARE PROVIDED "AS-IS" AND WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING BUT NOT LIMITED TO IMPLIED WARRANTIES OF MERCHANTABILITY, NON-INFRINGEMENT, AND FITNESS FOR A PARTICULAR PURPOSE.

Merative or its licensors may have patents or pending patent applications covering subject matter described in this document. The furnishing of this documentation does not grant you any license to these patents.

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

Any references in this information to non-Merative 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 Merative product and use of those websites is at your own risk.

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.

The licensed program described in this document and all licensed material available for it are provided by Merative under terms of the Merative Client Agreement.

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 Merative, 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. Merative, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs. The sample programs are provided "AS IS", without warranty of any kind. Merative shall not be liable for any damages arising out of your use of the sample programs.

Privacy policy

The Merative privacy policy is available at <https://www.merative.com/privacy>.

Trademarks

Merative™ and the Merative™ logo are trademarks of Merative US L.P. in the United States and other countries.

IBM®, the IBM® logo, and ibm.com® are trademarks or registered trademarks of International Business Machines Corp., registered in many jurisdictions worldwide.

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.

Oracle and Java are registered trademarks of Oracle and/or its affiliates.

The registered trademark Linux® is used pursuant to a sublicense from the Linux Foundation, the exclusive licensee of Linus Torvalds, owner of the mark on a worldwide basis.

Microsoft™, Windows™, and the Windows™ logo are trademarks of Microsoft™ Corporation in the United States, other countries, or both.

UNIX™ is a registered trademark of The Open Group in the United States and other countries.

Other company, product, and service names may be trademarks or service marks of others.