IBM Cúram Social Program Management
Version 7.0.5

Inside Cúram Decision Assist

IBM

Note

Before using this information and the product it supports, read the information in "Notices" on page 31

Contents

List of Figures

List of Tables

Chapter 1. Developing with Decision Assist

Use this information to develop custom classes and widgets for Cúram Decision Assist. Determination-based assessments support the capture of answers to questionnaires from relevant parties, the consolidation of those answers if necessary, and the passing of the answers through decision matrices to reach outcomes and aid decisions making.

Overview

This information provides details of the functionality in the Cúram Decision Assist application module. It describes the classes and widgets within the module and how the module integrates with IBM Cúram Social Program Management Platform. This information is targeted at a technical audience intending to implement Cúram Decision Assist.

Prerequisites

The `Cúram Decision Assist Guide` describes the business concepts of Cúram Decision Assist and thus is an important prerequisite.

Business Examples Used in this Guide

This document uses the following business examples to demonstrate Cúram Decision Assist functionality:

Sample Family Risk Assessment
The sample family risk assessment gathers information from a family where there is a current complaint for either abuse or neglect and ascertains the risk level for the family. For this business example, there are two decision matrices: one produces a single outcome based on a cumulative score (where the outcome is the level of risk) and the other produces a cumulative score only (the higher the score, the greater the risk).

Sample Division of Youth Corrections Screening
The sample division of youth corrections screening is used to determine the placement level for an arrested youth. There is one decision matrix which produces a single outcome, i.e. the level of placement, from the combination of answers passed into the matrix.

Sample Abuse Assessment
The sample abuse assessment is used to assess the level of abuse suffered by a person. In this assessment, the answers to the abuse questions are assigned a score; there is one decision matrix which produces a total score which measures the level of abuse.

Sample Family Services Assessment Form
The sample family services assessment form gathers information about a family's circumstances to ascertain which services may help improve the family's current situation. Each question in the assessment has the potential to lead to an outcome. There is one decision matrix for the assessment which is configured to produce multiple outcomes. This is an example where outcomes map to service plans and plan items map to decisions.

Sample Partner Maltreatment Assessment
The sample partner maltreatment abuse assessment gathers information to ascertain whether a person is being abused by their partner and the severity of that abuse. The combination of answers given to the questions asked of the partner will produce a total score which determines the partner's level of abuse. There is one decision matrix for the assessment which is configured to a single outcome and a total score. The outcomes are obtained from the combination of answers given.

Sample Asthma Medical Assessment
The sample asthma medical assessment gathers information to ascertain the severity of a person's asthma. The answers to questions vary for male and females so there are two determination packages: one for each gender. Multiple sources may be asked the questions and the latest answer for

each question is used to make decisions. There are two determination packages and thus two decision matrices for each gender.

Decision Assist Administration Class and Widget Overview

This information describes the Cúram Decision Assist administration classes and widgets. It does so by following the logic an administrator should follow to configure decision assist assessments. The first task is the configuration of determinations. Determination configurations integrate with Cúram assessments.

Each determination configuration can include one or more determination packages and each determination package has associated questionnaires and decision matrices. The questionnaires retrieve answers, the decision matrices contain simple rules, and the answers are run against the simple rules to reach a decision. The outcomes class is used to store these decisions.

Important: Decision Assist administration classes are located in both the facade and service layers. This makes is possible for the user interface to call them, and also for other process classes to call them in the service layer.

DeterminationConfig Class

The class links Cúram Decision Assist to the assessment functionality in IBM Cúram Social Program Management Platform. To understand the business processes for this class, it is helpful to have a high-level summary of the following attributes in the DeterminationConfig entity:

- Determination configuration type: This type is either medical, screening, or needs and is informational only.
- Multi-source indicator: Indicates whether there can be third-party answers for questionnaires. For example, the Asthma Medical assessment gathers information to ascertain the severity of a person's asthma. To complete this assessment, multiple sources can provide answers including the person whose asthma is being assessed and also their doctor.
- Stand-alone indicator: Indicates whether the determination configuration is stand-alone. The impact of setting the indicator to true is that only one association can be made between the determination configuration and an assessment. In other words, there is a one-to-one relationship between the assessment and determination configuration. When a user creates a determination-based assessment for the decision assist assessment, the stand-alone determination configuration is automatically used. Alternatively, if this indicator is set to false, then multiple determination configurations can be associated to the same decision assist assessment. When creating a determination-based assessment of this type, the user can select from the list of associated determination configurations.
- SupportCOCProcessing indicator: Allows for change of circumstance processing which has to do with making new decisions over time. The system reads this indicator to determine which view page to display when a user clicks the View link on the Assessments list page within a case. When set to true, all determinations and the latest decisions for each decision matrix are displayed on the View Assessment Details page. When set to false, there can be only one determination for the assessment. Clicking the View link brings a user directly to the View Determination Details page for the one determination.
- Automatic consolidation indicator: Indicates whether automatic consolidation is used to consolidate answers from multiple sources. The multi-source indicator must be set to true for automatic consolidation to be relevant. If answers from multiple sources are supported and if this is set to true, the latest answers by default are selected as the answers for the decision making.
- Start and end date set parameters for determination config: This sets the start and end date parameters for the determination configuration.

In addition to the above attributes, name translations can be added for a determination configuration record. The locale of a user who logs in is read and matched with the appropriate translation and the determination config name is displayed in language appropriate to user.

The following table describes in detail the business processes that make up the DeterminationConfig class:

Table 1: Description of DeterminationConfig Business Processes	
Business Process	**Description**
getDeterminationConfigIndicatorDetails	This method is used to read all active assessment configurations of type decision assist. Typically it is intended to be used by users responsible for administrating determination configuration to view determination configuration indicator details.
listActiveDAAssessmentConfig	This method is used to read all active assessment configurations of type decision assist.
listDeterminationConfig	This method lists all the determination configuration records.
readDeterminationConfig	This method returns a determination configuration record.
readDeterminationConfigIndicatorDetails	This method returns determination configuration indicators for a particular determination configuration.
readDeterminationConfigNameTranslation	This method returns a determination configuration name translation record.
listActiveDeterminationConfig	This method lists all active determination configurations excluding those which are already assigned to the assessment.
readAssessment	This method returns an assessment record.
readDAAssessment	This method is used to read assessment configuration details of type decision assist.
readDatesIndicatorDetails	This method is used to know whether the determination configuration's start and end dates are in future or not.
listDAAssessmentConfiguration	This method is intended to be used by users responsible for managing assessment configurations to list all the assessment configurations on the system, which are of type decision assist.

DeterminationPackage Class

This class packages up a set of questionnaires and a set of decision matrices as part of a determination configuration. It sets up the questions to be asked of specific persons and the rules which will be run against their answers in order to reach a decision. The attributes for the DeterminationPackage entity include the date and time the determination package was created on the system and the name of the system user who created it. The entity also includes product, age, and gender.

These attributes are used to match the appropriate determination package to a client being assessed (see "Process of Matching a Determination Package to a Client" on page 28).

The following table describes the business processes that make up the DeterminationPackage class:

Table 2: Description of DeterminationPackage Business Processes	
Business Process	**Description**
readDeterminationPackage	This method reads a determination package record. The difference between the read and view methods is that this method only returns the determination package details, not the questionnaire and decision matrix details.
viewDeterminationPackageDetails	This method returns all determination package details, including the decision matrix and questionnaire details.
readAllDPsForDeterminationConfig	This method retrieves all the determination packages associated with a determination configuration. Only the latest version of the determination package is displayed.
listQuestionnairesAssociatedToDP	This method returns the list of questionnaire records for a determination package record.

Questionnaire Class

Use the class to maintain questionnaires that can be associated with one or more determination packages. These questionnaires are used to retrieve the answers that are passed into the decision matrix. The attributes for the Questionnaire entity include the date and time the questionnaire was created on the system and the name of the system user who created it.

Questionnaires are created and maintained by using Cúram Intelligent Evidence Gathering (IEG). For information on integration with Cúram IEG, see "Questionnaire Integration with IEG" on page 27.

The following table describes the business processes that make up the Questionnaire class:

Table 3: Description of Questionnaire Business Processes	
Business Process	**Description**
listQuestionnaires	This method returns the list of questionnaire records on the system.
readIEGScriptKey	This method is used to return the IEGScript key information for editing the questions of the questionnaire.
readLatestQuestionnaireVersion	This method returns the most current questionnaire record on the system regardless of status. One questionnaire can have multiple versions, and the most current questionnaire record and its most current version are returned.
listTPsAssociatedToQuestionnaire	This method lists all the typical picture records that are associated with a questionnaire record.
viewQuestionnaireDetails	This method returns a questionnaire record with questionnaire details, questionnaire version information, and the list of questions in it.
readQuestionnaire	This method returns the questionnaire header details only: questionnaire name, version number, and type.

Table 3: Description of Questionnaire Business Processes (continued)	
Business Process	**Description**
`readLatestQuestionnaireVersionDescription`	This method returns some details of most current questionnaire record on the system regardless of status.
`readTypicalPictureHeaderDetails`	This method returns a questionnaire record based on the version number provided.
`validateQuestionScript`	This method is used to validate the question script that is associated with the questionnaire version.

DecisionMatrix Class

The Decision Matrix class is used to maintain information regarding the decision matrix. The following is a summary of some of the decision matrix header details which are maintained:

- Name: The name of the decision matrix should be unique.
- Type: The options for the decision matrix type are the same options as for the determination configuration and questionnaire (medical, screening, or needs). Note that the decision matrix type does not have to match the type of questionnaire or the type of determination configuration that it is associated with; it is informational only and thus there is no validation.
- Cumulative score: When selected, this means that a total score will be calculated when the all answers given are passed through the decision matrix. For example, one of the sample family risk assessment decision matrices produces a cumulative score - the higher the score, the greater the risk.
- Multiple outcomes: When selected, this means that there can be more than one outcome rendered from the decision matrix. For example, the sample family services assessment form gathers information about a family's circumstances to ascertain which services may help improve the family's current situation. Each question in the assessment has the potential to lead to an outcome (the outcomes being matched with appropriate services).
- Scoring by outcomes: When selected, rather than returning a cumulative score for the matrix as a whole, a score is calculated for each outcome generated from the answers used to generate that outcome. For example, the sample family services assessment form gathers information about a family's circumstances to ascertain which services may help improve the family's current situation. Each question in the assessment has the potential to lead to an outcome (the outcomes being matched with the appropriate services), if the answer is given a high enough score. If the outcome is generated, both the outcome and score are returned.
- Decision matrix version number: This is used to maintain different versions of the same decision matrix. It is also used to clone a decision matrix. More information about decision matrix versions is provided in the descriptions of the decision matrix business processes.

The following table describes the business processes that make up the DecisionMatrix class:

Table 4: Description of DecisionMatrix Business Processes	
Business Process	**Description**
readAllLatestVersionDecisionMatrices	This method retrieves the latest decision matrices in full, i.e. the decision matrix header details and data.
readDecisionMatrixVersion	This method returns the details of a decision matrix record for the specified version key. This key is used to distinguish the different versions of the same decision matrix.

Table 4: Description of DecisionMatrix Business Processes (continued)	
Business Process	**Description**
readAllDPsForDecisionMatrices	This method retrieves all the determination packages associated with the decision matrix. Only the latest version of the determination package is displayed.
readDecisionMatrixHeaderDetails	This method returns the name and ID of all of the latest decision matrix records. This is used, for example, to populate a drop-down list of the latest decision matrices.
readDecisionMatrixData	This method returns all decision matrix data (row, column, and cell data).
listOutcomesForMatrix	This method returns the list of outcome records associated with a decision matrix.
listDecisionMatrixVersionDetails	This method is intended to be used by users responsible for managing decision matrix to list all the decision matrix records of the specified version key on the system.
getQuestionnaireForMatrix	This method returns the questionnaire record selected for the decision matrix.
getQuestionsForMatrix	This method returns the questions selected from their questionnaire for the decision matrix. This method is called after the GetQuestionnaireForMatrix method.
readOutcomeScoreAssociation	This method returns the association between an outcome and a decision matrix.

Decision Matrix Widget

The Decision Matrix widget opens when a user edits a decision matrix. Decision Matrix is effectively a table which has rows and subrows, columns and subcolumns. Each row in the matrix (excluding header information) represents a single question; each column represents a potential outcome.

Each question can have multiple subrows to represent potential answers to the question; each outcome can have multiple subcolumns which contain a checkbox for each answer; the checkboxes are used to specify which combinations of answers lead to a given outcome. The answer type values supported in the Decision Matrix are booleans (true/false), code tables, numerals, and strings.

Contradictions can also be added to a Decision Matrix which indicate answers in contradiction to each other. For example, the question, "Can you walk" is parallel to a question, "Can you run". If a person answers no to the first question, then it should not be possible to answer yes to the second question; it would be a contradiction for a person to state that he or she can run but cannot walk.

In order to display the initial view of a Decision Matrix, an XML string describing its contents will be returned from the server code. A user can edit the Decision Matrix (add questions, answers, outcomes, etc.), and save all of these modifications at the same time. A string with the same XML structure will be passed into the server code, thus storing the updates. The following is a sample XML string which can be passed into the server code (including support for multiple locales):

```
<DECISION_MATRIX_XML ID="M1">
    <QUESTIONS>
        <QUESTION QUESTION_ID="Q1" IS_NEW="false">
            <ANSWER ANSWER_ID="1" VALUE="C1" PRIORITY="1" />
            <ANSWER ANSWER_ID="2" VALUE="C2" PRIORITY="2" />
            <ANSWER ANSWER_ID="3" VALUE="C3" PRIORITY="3" />
        </QUESTION>
        <QUESTION QUESTION_ID="Q3" IS_NEW="false">
```

```xml
            <ANSWER ANSWER_ID="1" VALUE="C1" PRIORITY="1" />
            <ANSWER ANSWER_ID="2" VALUE="C2" PRIORITY="2" />
            <ANSWER ANSWER_ID="3" VALUE="C3" PRIORITY="3" />
        </QUESTION>
        <QUESTION QUESTION_ID="Q5" IS_NEW="false">
            <ANSWER ANSWER_ID="1" VALUE="A" PRIORITY="1" />
            <ANSWER ANSWER_ID="2" VALUE="B" PRIORITY="2" />
            <ANSWER ANSWER_ID="3" VALUE="C" PRIORITY="3" />
        </QUESTION>
        <QUESTION QUESTION_ID="Q6" IS_NEW="true">
            <ANSWER ANSWER_ID="1" VALUE="Yes" PRIORITY="1" />
            <ANSWER ANSWER_ID="2" VALUE="No" PRIORITY="2" />
        </QUESTION>
        <QUESTION QUESTION_ID="Q7" IS_NEW="true">
            <ANSWER ANSWER_ID="1" PRIORITY="1" MIN="1" MAX="9" />
            <ANSWER ANSWER_ID="2" VALUE="10" PRIORITY="2" />
            <ANSWER ANSWER_ID="3" PRIORITY="3" MIN="11" />
        </QUESTION>
    </QUESTIONS>
    <CONTRADICTIONS>
        <COMBINATION COMBINATION_ID="C1">
            <MESSAGE LOCALE="en_US" TEXT="A person undergoing
              third level treatment for arthritis cannot
              walk for 3 KMs"/>
            <MESSAGE LOCALE="fr" TEXT="Une personne subissant
             le troisième traitement égal pour l'arthrite ne
             peut pas marcher pour 3 KM"/>
            <SELECTION QUESTION_ID="Q1" ANSWER_ID="1" />
            <SELECTION QUESTION_ID="Q7" ANSWER_ID="1" />
        </COMBINATION>
        <COMBINATION COMBINATION_ID="C2">
            <MESSAGE LOCALE="en_US" TEXT="A person undergoing
              third level treatment for arthritis cannot walk
              without support"/>
            <MESSAGE LOCALE="fr" TEXT="Une personne subissant
             le troisième traitement égal pour l'arthrite ne
             peut pas marcher sans le soutien"/>
            <SELECTION QUESTION_ID="Q1" ANSWER_ID="1" />
            <SELECTION QUESTION_ID="Q6" ANSWER_ID="2" />
            <SELECTION QUESTION_ID="Q7" ANSWER_ID="1" />
        </COMBINATION>
    </CONTRADICTIONS>
    <OUTCOMES>
        <OUTCOME OUTCOME_ID="O1" IS_NEW="false">
            <COMBINATION COMBINATION_ID="C3">
                <SELECTION QUESTION_ID="Q1" ANSWER_ID="3" />
                <SELECTION QUESTION_ID="Q3" ANSWER_ID="2" />
                <SELECTION QUESTION_ID="Q6" ANSWER_ID="1" />
            </COMBINATION>
        </OUTCOME>
        <OUTCOME OUTCOME_ID="O2" IS_NEW="true">
            <COMBINATION COMBINATION_ID="">
                <SELECTION QUESTION_ID="Q3" ANSWER_ID="3" />
                <SELECTION QUESTION_ID="Q5" ANSWER_ID="2" />
                <SELECTION QUESTION_ID="Q6" ANSWER_ID="2" />
                <SELECTION QUESTION_ID="Q7" ANSWER_ID="2" />
            </COMBINATION>
        </OUTCOME>
    </OUTCOMES>
    <DELETED_ITEMS>
        <QUESTION QUESTION_ID="Q2"/>
        <QUESTION QUESTION_ID="Q4"/>
        <OUTCOME OUTCOME_ID="O3"/>
    </DELETED_ITEMS>
</DECISION_MATRIX_XML>
```

Outcomes Class

The Outcomes class is used to store the outcomes of the decisions rendered by the decision matrix. A determination package can result in multiple decisions of different types with multiple outcomes if it has more than one decision matrix associated with it.

The start and end date attributes on the Outcome entity are used to filter which active outcomes are available for selection when editing decision matrix data. Once an outcome is added to a decision matrix, it stays in place regardless of its end date.

As with the DeterminationConfig class, name translations can be added for an outcome record. The locale of a user who logs in is read and matched with the appropriate translation and the outcome name will be displayed in the language appropriate to the user.

The following table describes the business processes that make up the Outcomes class:

Table 5: Description of Outcomes Business Processes	
Business Process	**Description**
listActiveOutcomes	This method returns the list of active only outcome records on the system.
readOutcome	This method reads an outcome record. The difference between the read and view methods is that this method only returns the outcome details, not the name translation details.
readOutcomeNameTranslation	This method is to read an outcome name translation record.
viewOutcome	This method reads an outcome record, including outcome details and name translation details.
listOutcomeDetails	This method is intended to be used by users responsible for managing outcomes to list all the outcome records on the system irregardless of status.

QuestionnaireDecisionOutcomeMetadata Class

This class is used to store presentation data for questionnaires. Presentation data indicates the text that is displayed for a questionnaire.

For example, if the presentation data is set to "mandatory" for a questionnaire, the text "mandatory" is displayed beside the questionnaire on the View Assessment Details and View Determination Details pages. Presentation data can also label a questionnaire as recommended. There is no validation for presentation data; it is for information purposes only.

Presentation data will only be read for questionnaires associated with a determination package and which link to an outcome and a decision matrix.

The following table describes the business processes that make up the QuestionnaireDecisionOutcomeMetadata class:

Table 6: QuestionnaireDecisionOutcomeMetadata Business Processes	
Business Process	**Description**
readQuestionnairePresentationData	Reads the presentation data record for a questionnaire based on a key. This key distinguishes different versions of the same presentation data record.
listPresentationData	This method returns a list of all presentation data records for a questionnaire.
readAllActiveOutcomes	This method returns the list of all active outcomes on the system. It is used to link presentation data to an active outcome.

Table 6: QuestionnaireDecisionOutcomeMetadata Business Processes (continued)	
Business Process	**Description**
readPresentationDataHeaderDetails	This method returns the questionnaire header details: questionnaire name, version number, type, and status.

TypicalPicture Class

The Typical Picture class is used to compare the answers recorded for a questionnaire to a set of typical answers for that same questionnaire.

It is used primarily for medical conditions. For example, a person may answer a series of questions relating to a medical condition. The answers the person gives are aligned with typical answers to the same set of questions and the differentials between the real and typical answers can be observed.

The following table describes the business processes that make up the TypicalPicture class:

Table 7: Description of TypicalPicture Business Processes	
Business Process	**Description**
readTypicalPicture	This method reads a typical picture record.
readAllActiveOutcomes	This method lists all active outcomes on the system.
readAllDecisionOutcomes	This method reads the combination of all decision matrices and outcomes created for a typical picture record.
readDecisionOutcome	This method reads the presentation data record for a typical picture based on a key. This key distinguishes different versions of the same presentation data record.
readTypicalAnswers	This method reads a single combination of decision matrix and outcome association for a typical picture record.
readTypicalAnswerForQuestion	This method returns the typical answers for a questionnaire.

Condition Class

This class is used to associate Conditions to Determination Configurations for use in medical assessments.

Name translations can be added for a condition record. The locale of a user who logs in is read and matched with the appropriate translation and the condition name will be displayed in the language appropriate to the user.

The following table describes the business processes that make up the Condition class:

Table 8: Description of Condition Business Processes	
Business Process	**Description**
listConditionsAssociatedToDC	This method returns the list of condition records associated with a determination configuration record.

Table 8: Description of Condition Business Processes (continued)

Business Process	Description
listDCsAssociatedToCondition	This method lists all the determination configuration records with which a condition record is associated.
listICDCodesAssociatedToCondition	This method lists all the ICD code records for a condition record.
viewICDCodeConditionAssociation	This method reads an ICD code association record.
viewCondition	This method is to view all the details of the condition.
searchConditionsForICDDetails	This method is to search the condition details for the given ICD code details.
readConditionNameTranslation	This method reads a condition name translation record.
listConditionDetails	This method is intended to be used by users responsible for managing conditions to list all the conditions records on the system.

ICDCode Class

This class is used to associate International Classification of Diseases (ICD) Codes with Conditions. ICD Codes are the World Health Organization International Classification of Diseases.

Name translations can be added for an ICD code record. The locale of a user who logs in is read and matched with the appropriate translation and the ICD code name will be displayed in the language appropriate to the user.

The following table describes the business processes that make up the ICDCode class:

Table 9: Description of ICDCode Business Processes

Business Process	Description
searchICDCode	This method searches for a particular ICD code.
searchICDCodeDetails	This method returns a list of ICD code details based on the search criteria.
listICDCodeVersionDetails	This method lists out all the ICD code version details.
readReleasedVersionNumbers	This method reads only released versions of ICD codes.
viewICDCodeDetails	This method returns an ICD code record.
readICDTextTranslation	This method is to get the ICD code text translation details.
readICDTextTranslations	This method reads an ICD code name translation record.
readActiveVersionNumbers	This method is to get the list of active ICD code version.
readICDTextTranslations	This method reads an ICD code name translation record.

Table 9: Description of ICDCode Business Processes (continued)	
Business Process	**Description**
viewICDCodeMaxVersionNumber	To method is to get the details of the latest ICD code verson.

Determination Based Assessment Class and Widget Overview

This information describes the classes and widgets that make up determination based assessments. The goal of determination based assessments is to make decisions on behalf of a client and for either a product delivery case or an integrated case.

It is important to make the connection between decision assist administration and determination based assessments. When a determination based assessment is created on a case, it inherits the pre-configured information set up as part of decision assist administration.

The functionality for determination based assessments supports the capture of answers to questionnaires from relevant parties (including third-parties), the consolidation of those answers if necessary, and the passing of the answers through decision matrices in order to reach outcomes and make decisions.

DAAssessment Class

The business processes in this class are fundamental to the determination based assessment lifecycle. There are three statuses in this lifecycle: in progress, canceled, or completed. When first created, a determination based assessment is in progress. It can be canceled at any point in its lifecycle.

In order to complete a determination based assessment, a decision must be made. Based on a decision, an outcome can be made for the determination based assessment.

Determinations can be marked as complete. If change in circumstances are supported, once a determination is complete, no decisions can be created for it; however, new determinations can be made for the same determination based assessment. If change in circumstances are not supported, once a determination is complete, no more determinations (or decisions) can be made for the determination based assessment.

Assessments can be cloned which results in the copying of the assessment and all its components into a new assessment (questionnaires, decision matrices, etc.).

The following table describes the business processes that make up the DA

Assessment Class:

Table 10: Description of DAAssessment Business Processes	
Business Process	**Description**
viewAssessment	Reads the details of an assessment. The assessment details include the determination delivery key, determination configuration flags, a list of questionnaires, a list of third party requests, if the determination supports multiple source of answers and a list of decisions made for the assessment.

Table 10: Description of DAAssessment Business Processes (continued)	
Business Process	**Description**
completeAssessment	Completes an assessment, modifying the assessment status to 'Completed'. All the associated determinations are also marked as completed. Prior to completing the assessment, the system verifies whether the user has indicated that a decision should be made as part of the completion of the assessment, and if so a decision is also created for the assessment. The status of all the existing decisions is updated to 'Superseded'.
cancelAssessment	Cancels an assessment by modifying the assessment status to 'Cancelled'. All the associated determinations and decisions of the assessment are also cancelled.
areDecisionsMadeForAssessment	Checks whether any decisions are made for the assessment or not.
areNewAnswersProvided	Checks whether or not the assessment answers have been modified since the last time they were consolidated.
makeDecision	Makes decisions for an assessment. This applies the answers provided to the set of decision matrices associated with a determination. A decision is created for each decision matrix that is defined for the determinations of the assessment. Each decision can have zero to many outcomes within it. A decision may also contain a score per outcome or an overall score.
readAllAssessmentWithUserFullNameStatusHistory	Returns the status history for an assessment.
listCaseParticipants	Returns a list of case participants associated with an assessment for a determination.
viewAssessmentWithCOC	Reads the assessment details for an assessment which supports change of circumstance processing. The information returned includes the assessment details and a list of determinations associated with the assessment.
viewAssessmentDetailsWithCOC	Reads the details of an assessment which supports changes of circumstance. The details include the list of decisions made for the assessment.
cloneAssessment	Clones an assessment. All the assessment information, including decisions, questions and answers, third party requests, primary client details and determinations are also copied to the cloned assessment. For more information on this, see "Determination Package Matching Rules" on page 29.
viewCaseMemberDetails	Reads the case member details for a case participant.

Table 10: Description of DAAssessment Business Processes (continued)	
Business Process	**Description**
readAssessmentDeliveryDetails	Reads the details of an assessment delivery for a case.

Decisions Class

The Decisions class is used to view decision information and to compare decisions.

The following table describes the business processes that make up the Decisions Class:

Table 11: Description of Decisions Business Processes	
Business Process	**Description**
viewDecisionDetails	Reads the details of a decision. The details of outcomes associated with the decision are also retrieved.
viewDecisionAnswerDetails	Reads the details of questions and answers associated with a determination decision in an xml format.
listDecisionsDetailsForComparison	Returns a list of decisions associated with the determination for comparison. All the decisions except the source decision whose key is passed as a parameter to this method and their outcome details are retrieved.
compareDecisionsDetails	Reads the details of two determination decisions for comparison. The details include the decision details, outcome details and the questions and answers received for the two determination decisions in an xml format. For information on the evidence comparison widget, see "Evidence Comparison Widget" on page 13.
readDeterminationDecisionHistory	Returns the decision history for a determination. A list of decision outcome details is also retrieved.
readAssessmentDecisionHistory	Returns the decision history for an assessment. A list of decision outcome details is also retrieved.

Evidence Comparison Widget

The Evidence Comparison widget is used to compare the answers to the same questionnaires entered at different times in order to see the differences between them. Notice that for consolidated answers, the widget provides a view link which enables a user to view the answers from the different sources.

The obvious situation in which there are differences between the two baselines for a given question is where the answers are actually different, for example, when circumstances have changed. It is important to note that the questions asked can also change over time, which means that it is possible to have a question present in one baseline but not in the other.

The UIM FIELD should have a source connection to a "display phase" server interface based on the domain EVIDENCE_COMPARISON_XML.

Evidence Comparison XML Format

The Evidence Comparison Widget requires XML data that conforms to the evidence comparison schema. This schema is defined in the `evidence-comparison.xsd` file located in the `lib/curam/xml/schema` folder of your CDEJ installation folder.

Below is an example of the kind of XML necessary:

```
<EVIDENCE_COMPARISON DETERMINATION_DELIVERY_ID="45556363"
    BASELINE1_LABEL="label 1" BASELINE2_LABEL="label 2">
  <QUESTIONNAIRE QUESTIONNAIRE_ID="QE1"
              LABEL="Questionnaire 1">
    <QUESTION QUESTION_ID="Q1" HIGHLIGHT="false"
            TYPE="NORMAL" SUB_TYPE="INT32"
            LABEL="What is your gender?">
      <BASELINE1 ANSWER="Male" SHOW_LINK="false"
              DISPLAY_TEXT="Question Not Asked"/>
      <BASELINE2 ANSWER="Male" SHOW_LINK="false"/>
    </QUESTION>
    <QUESTION QUESTION_ID="Q2" LABEL="What is your age?"
            HIGHLIGHT="false" TYPE="NORMAL" SUB_TYPE="INT32">
      <BASELINE1 ANSWER="35" SHOW_LINK="false"/>
      <BASELINE2 ANSWER="35" SHOW_LINK="false"
              DISPLAY_TEXT="Question Unanswered"/>
    </QUESTION>
    <QUESTION QUESTION_ID="Q3" TYPE="NORMAL"
            SUB_TYPE="COMMENTS" HIGHLIGHT="true"
            LABEL="What country do you live in?">
      <BASELINE1 ANSWER="Question not asked"
              SHOW_LINK="false"/>
      <BASELINE2 ANSWER="Kildare" SHOW_LINK="false"/>
    </QUESTION>
    <QUESTION QUESTION_ID="Q4"
        LABEL="Do you have trouble breathing when you walk?"
        HIGHLIGHT="false" TYPE="NORMAL" SUB_TYPE="COMMENTS">
      <BASELINE1 ANSWER="yes" SHOW_LINK="true"/>
      <BASELINE2 ANSWER="yes" SHOW_LINK="false"/>
    </QUESTION>
  </QUESTIONNAIRE>
  <QUESTIONNAIRE QUESTIONNAIRE_ID="QE2"
              LABEL="Questionnaire 2">
    <QUESTION QUESTION_ID="Q8" LABEL="Question 8?"
        HIGHLIGHT="false" TYPE="NORMAL" SUB_TYPE="INT32">
      <BASELINE1 ANSWER="answer" SHOW_LINK="false"/>
      <BASELINE2 ANSWER="answer" SHOW_LINK="false"/>
    </QUESTION>
    <QUESTION QUESTION_ID="Q9" LABEL="Question 9?"
        HIGHLIGHT="true" TYPE="NORMAL" SUB_TYPE="INT32">
      <BASELINE1 ANSWER="answer" SHOW_LINK="false"/>
      <BASELINE2 ANSWER="different answer"
            SHOW_LINK="false"/>
    </QUESTION>
    <QUESTION QUESTION_ID="Q10" LABEL="Question 9?"
        HIGHLIGHT="false" TYPE="NORMAL" SUB_TYPE="INT32">
      <BASELINE1 ANSWER="answer" SHOW_LINK="false"/>
      <BASELINE2 ANSWER="answer" SHOW_LINK="false"/>
    </QUESTION>
  </QUESTIONNAIRE>
</EVIDENCE_COMPARISON>
```

Figure 1: Evidence Comparison

Table 12: Attributes of Evidence Comparison Element	
Attribute	**Description**
DETERMINATION_DELIVERY_ID	Unique identifier for this element, supplied as a parameter to the row hyperlink.
BASELINE1_LABEL	Label for the first set of answers.
BASELINE2_LABEL	Label for the second set of answers

Table 13: Attributes of Questionnaire Element	
Attribute	**Description**
QUESTIONNAIRE_ID	Unique identifier for this element, supplied as a parameter to the row hyperlink.
LABEL	Label for the questionnaire or question.

Table 14: Attributes of Question Element	
Attribute	**Description**
QUESTION_ID	Unique identifier for this element, supplied as a parameter to the row hyperlink.
HIGHLIGHT	Flag to highlight the row.
TYPE	The type is either NORMAL or CODETABLE.
SUB_TYPE	The behavior of the SUB_TYPE attribute value depends on the value of the TYPE attribute. In case where the TYPE attribute is specified as NORMAL, the SUB_TYPE refers to the actual data type to used value; where TYPE value is specified as CODETABLE, the SUB_TYPE value would be the name of the codetable to use.

Table 15: Attributes of Baseline Elements	
Attribute	**Description**
ANSWER	The answer text to the question.
SHOW_LINK	Flag to create a link from existing answer to another.
DISPLAY_TEXT	This text takes precedence over answer value.

Configuration

The Evidence Comparison can be configured by setting the CONFIG attribute on the UIM FIELD in question. The EvidenceComparisonConfig.xml XML configuration file must contain a configuration section with a unique identifier matching the text in the CONFIG attribute.

Where the field has not been configured on the UIM, the default will be used. The configuration can be used to change the link URL. The parameter names are stored in the configuration file and the parameter values are stored in the XML data. The following is a sample of this file:

```
<EVIDENCE_COMPARISON_CONFIG DEFAULT="MyConfiguration">
  <CONFIG ID="MyConfiguration">
    <ANSWER_SOURCE_LINK
      PAGE_ID="EvidenceComparisonViewAnswerSource"
      QUESTIONNAIRE_ID_PARAM="questionnaireID"
      QUESTION_ID_PARAM="questionID"
      DETERMINATION_DELIVERY_ID_PARAM="determinationDeliveryID"/>
  </CONFIG>
</EVIDENCE_COMPARISON_CONFIG>
```

Figure 2: Evidence Comparison Configuration

Table 16: Attributes of Evidence Comparison Config Element	
Attribute	**Description**
DEFAULT	The configuration to be used if it is not specified in the UIM.

Table 17: Attributes of Config Element	
Attribute	**Description**
ID	The unique identifier for the configuration.

Table 18: Attributes of Answer Element	
Attribute	**Description**
PAGE_ID	The ID of the target page for the link.
QUESTIONNAIRE_ID_PARAM	Parameter name for the link.
QUESTION_ID_PARAM	Parameter name for the link.
DETERMINATION_DELIVERY _ID_PARAM	Parameter name for the link.

DeterminationDelivery Class

The DeterminationDelivery class is used to maintain the questionnaires, decision matrices, and decisions for a determination based assessment.

Where change of circumstances is not supported, there is a one-to-one relationship between the determination based assessment and the determination delivery; where change of circumstances is supported, there can be multiple determination deliveries for the one assessment, each one representing a change in circumstance which has resulted in a new decision.

To understand the business processes for this class, it is helpful to have a high level summary of the following attributes in the DeterminationDelivery entity:

- determinationDeliveryID, assessmentCaseID, and determinationConfigID: This three attributes identify the decision assist assessment case, the determination configuration matched to this case to reach a decision, and the decision information stored as part of the determination delivery.
- determinationPackageVersionID: This is the version number of the determination package which is used to make a decision for the determination delivery.
- status: The determination delivery can have a status of in progress, canceled, active, or superseded.
- createdDateTime: This is the date and time the determination delivery record was created on the system.
- effectiveDate: This is the effective date of the determination delivery record.

The following table describes the business processes that make up the DeterminationDelivery Class:

Table 19: Description of DeterminationDelivery Business Processes	
Business Process	**Description**
cancelDetermination	Cancels a determination by modifying the determination status to 'Cancelled'. Any decisions created as a result of the determination are also cancelled.

Table 19: Description of DeterminationDelivery Business Processes (continued)	
Business Process	**Description**
completeDetermination	Completes a determination, modifying the determination status to 'Completed'. Prior to completing the determination, the system verifies whether the user has indicated that a decision should be made as part of the completion of the determination, and if so a decision is also created for the determination. The status of all the existing determination decisions are marked as 'Superseded'.
areDecisionsMadeForDetermination	Checks whether any decisions are made for the determination or not.
readDeterminationDelivery	Reads the details of a determination.
viewDeterminationDelivery	Reads the details of a determination. The determination details include the details of a determination configuration, a list of associated questionnaires and a list of third party request details, if the determination configuration supports multiple sources. It also retrieves a list of decisions made for the determination.
areNewAnswersProvided	Checks whether or not the determination answers have been modified since they were last consolidated.
makeDecision	Makes decisions for a determination. This applies the answers provided to the set of decision matrices associated with the determination to determine the appropriate results. A decision is created for each decision matrix that is defined for the determinaton. Each decision can have zero to many outcomes within it. A decision may also contain a score per outcome or an overall score.
readAllDeterminationStatusHistoryDetails	Returns the status history for a determination.
recordChangeOfCircumstance	Records a change of circumstance for the determination. A change of circumstance is a process of creating a new determination which may be from the same effective date as a previous determination or from a different effective date. If the new determination is from a different effective date, then the questionnaires associated with the determination may not be the same. Questions may have been added or removed from a questionnaire (different version of the questionnaire) or questionnaires may have been added or removed. No decisions are copied to the new determination. Only the answers to questions provided for the original determination where the question is again asked on the newly created determination are copied.
listDeterminations	Returns a list of determination details for an assessment.

ThirdPartyRequest Class

The ThirdPartyRequest class is used to request answers to questionnaires from multiple sources. It allows a user to define which questions require answers from multiple sources, to request these answers from the sources, and to record the third party answers.

Since third parties usually only answer portions of questionnaires, the functionality provided to record third party answers does not use the IEG questionnaire scripts, but rather comes with its own set of pages that an application user can access in order to record the third party answers.

Note, however, as part of creating the IEG questionnaire scripts, question aliases can be entered for each question. Question aliases are used to reword questions for third parties. For example, questions regarding a person's health can be reworded for a doctor to reference more specific medical information. For more information on IEG and questionnaires, see "Questionnaire Integration with IEG" on page 27.

There are two entities which are used to store third party request information. The ThirdPartyRequest entity contains the information relating to the actual request made to a third party to provide answers to specific questions. A ThirdPartyRequest record can be the parent to a number of ThirdPartyRequestQuestion records. Each one of these records stores information about a specific question to be answered by the third party in the third party request.

The following table describes the business processes that make up the ThirdPartyRequest Class:

Table 20: Description of ThirdPartyRequest Business Processes	
Business Process	**Description**
viewThirdPartyRequest	Reads the details of a third party request. The details include the list of questions and answers associated with the third party request for a determination.
createThirdPartyRequest	Creates a third party request. The types of a third party request are Customer, Third Party, Third Party Medical. A third party request is created for the third party type based on the question alias types given in the IEG script. A third party can be either a case participant or non-case participant but registered user. If the third party is not a registered user, then an option is provided to specify the third party details.
sendThirdPartyRequest	Modifies the status of the third party request to 'Sent'. It is important to note that this process does not actually send the request, as it is not aware of where the request should be sent to or how best to send it. It is up to the system integrator to determine where that information is sent to.
cancelThirdPartyRequest	Cancels a third party request.
getThirdPartyQuestionAnswerDetails	Reads a third party request and its associated question and details. The questions and answers are returned in an xml format.
recordThirdPartyAnswers	Records the third party answers. The status of the third party request is changed to 'Answers Received'. Consolidation details are modified for the determination questionnaire. If automatic consolidation is enabled for the determination, then consolidation indicators are set to true for the recorded third party answers.

Table 20: Description of ThirdPartyRequest Business Processes (continued)	
Business Process	**Description**
editThirdPartyAnswers	Modifies the third party answers. The status of the third party request is changed to 'Answers Received'. Consolidation details are modified for the determination questionnaire. If automatic consolidation is enabled for the determination, then consolidation indicators are set to true for the recorded third party answers.
createTPRQuestions	Creates the third party questions. The third party questions included are unanswered questions, unconsolidated questions and consolidated questions. The status of the third party request is set to 'Not Sent'.
readThirdPartyRequest	Reads the details of a third party request.
editThirdPartyRequests	Modifies a third party request. Questions are added or deleted based on the user selection for the third party request.

QuestionAnswer Class

The QuestionAnswer class integrates with the Cúram intelligent evidence gathering (IEG) runtime editor.

Refer to "Questionnaire Integration with IEG" on page 27. It helps users maintain the answers to questionnaires that are run through the **Decision Matrix** to reach a decision. It provides the **View Questions** and **Answers** pane that displays the questions that are asked, the answers tat are received, and, if multiple sources exist, the name of the source who provided the answer.

The main attributes for the DeterminationAnswer entity include the following components:

- questionID - This entry identifies the question asked.
- answer - This entry is the answer that is given.
- thirdPartyInd - This entry indicates whether the answer if from a third party.
- sourceID - This entry contains the source of the answer, if the thirdpartyInd is false. If the thirdPartyInd is true, the sourceID contains the ThirdPartyRequestID.
- consolidationInd - This entry indicates whether the answer was consolidated automatically.

The following table describes the business processes that make up the QuestionAnswer class:

Table 21: Description of QuestionAnswer business processes	
Business Process	**Description**
viewQuestionAnswers	Reads the answer details for a question within a determination. The answer details contain information that relates to answers, which includes the source of the answers, the case participant name, the consolidation details, and the question details, which include the questionnaire name, questionnaire version number, and the question text.
viewAssessmentAnswers	Returns a list of questions and answers for an assessment.

Table 21: Description of `QuestionAnswer` business processes (continued)	
Business Process	**Description**
`answerQuestionnare`	Reads an IEG execution key that is associated with the questionnaire for a determination. If the IEG execution key does not exist, then a new IEG execution object is created and its key is returned.
`saveAnswers`	Saves answers of an IEG execution object that is associated with the determination and the questionnaire. If answers already do not exist for the questions, they are created. Answers that exist for the questions are replaced with the newly provided answers. All the answers for the unanswered questions are removed. Consolidation details are modified for the determination questionnaire.
`consolidateAnswers`	Consolidates the answers for the questions of a determination, where answers for a question are received from different sources. The system sets the consolidation indicator to `true` for all the answers in the inputted XML. After consolidation, the system modifies the consolidation status and date for the determination questionnaire.
`readAnswersForConsolidation`	Reads the details of the questions and answers of a determination questionnaire in an XML format for consolidation.
`readConsolidatedAnswers`	Reads the consolidated answers in an XML format for a determination questionnaire. If an answer for a question is from a third party, then the system reads the answer from the third-party request.
`readDeterminationDetails`	Reads the details of a determination for an IEG execution object.
`viewQuestionnaireAnswersDetails`	Reads a questionnaire name and the list of questions and answers in an XML format for a determination and questionnaire.
`viewDeterminationAnswers`	Returns a list of questions and answers for a determination.

Evidence Review Widget

The Evidence Review Widget is used to consolidate answers from multiple sources. It is launched by clicking the Consolidate link in the View Assessment Details and View Determination Details pages.

The Consolidate link is only enabled when the multi source indicator is set to true and the auto consolidate indicator is false for the determination configuration.

When launched, the user will be prompted to select the appropriate answer for each question requiring consolidation in the question group.

This widget calls on two methods: ViewConsolidatedAnswers and ViewAnswersForConsolidation. The ViewConsolidatedAnswers method is called when answers have already been consolidated. It displays the different answers which were provided from the multiple sources and it displays the details of the answer which was already selected. The information returned is read only. The ViewAnswersForConsolidation displays the different answers which were provided where no answer has been selected, and thus, the answers can be consolidated.

The widget is associated with the EVIDENCE_REVIEW_XML domain. The data to display in the evidence review control is provided in XML format. The format for the XML data is described below. The control has two modes of operation: `view` (read-only) mode and `edit`, which allows for reviewing answers and choosing an appropriate one from the available options. In edit mode, a user can make or change a selection from the list of available answers. In view mode, the only difference is the absence of radio buttons, as it is meant for review only and not for any changes. The rest of the functionality remains the same and a user can review questions and actual answers displayed in section headers, as well as answer options within sections that cannot be collapsed (if JavaScript is enabled) - just as it could in `edit` mode.

XML Formats

The Evidence Review widget expects information in a specific XML format, which conforms to the appropriate schema, specified by `evidence-review.xsd` file located in the `lib/curam/xml/schema` folder of your CDEJ installation folder.

This format is the same for both modes and can be distinguished by the value of EVIDENCE_REVIEW_EDITABLE. For the Edit mode it is set to `true` and for view mode it is set to `false`. Below is an example of this:

```
<EVIDENCE_REVIEW EDITABLE="true|false">
  <QUESTION TYPE="NORMAL" SUB_TYPE="SVR_STRING" QUESTION_ID="Q1"
          LABEL="What gender are you?" CHOSEN_ANSWER_ID="A1">
    <ANSWERS>
      <ANSWER ANSWER_ID="A1" NAME="John Doe"
            DESCRIPTION="Claimant" VALUE="Male"/>
    </ANSWERS>
  </QUESTION>
  <QUESTION TYPE="NORMAL" SUB_TYPE="COMMENTS" QUESTION_ID="Q2"
          LABEL="What age are you?" CHOSEN_ANSWER_ID="">
    <ANSWERS>
      <ANSWER ANSWER_ID="A2" NAME="John Doe"
            DESCRIPTION="Claimant" VALUE="35"/>
    </ANSWERS>
  </QUESTION>
  <QUESTION TYPE="NORMAL" SUB_TYPE="BOOLEAN_TYPE"
          QUESTION_ID="Q3" CHOSEN_ANSWER_ID="A3"
          LABEL="Do you have trouble breathing when you walk?">
    <ANSWERS>
      <ANSWER ANSWER_ID="A3" NAME="John Doe"
            DESCRIPTION="Claimant" VALUE="true"/>
      <ANSWER ANSWER_ID="A4" NAME="Dr. Jones"
            DESCRIPTION="General Practitioner" VALUE="Yes"/>
    </ANSWERS>
  </QUESTION>
  <QUESTION TYPE="CODETABLE" SUB_TYPE="Cars" QUESTION_ID="Q5"
          LABEL="What car do You own?" CHOSEN_ANSWER_ID="">
    <ANSWERS>
      <ANSWER ANSWER_ID="A7" NAME="John Doe"
            DESCRIPTION="Claimant" VALUE="CR7"/>
      <ANSWER ANSWER_ID="A8" NAME="Mr. Jones"
            DESCRIPTION="Neighbour" VALUE="CR6"/>
    </ANSWERS>
  </QUESTION>
</EVIDENCE_REVIEW>
```

Table 22: Attributes of Evidence Review Question Element	
Attribute	**Description**
TYPE	The type is either NORMAL or CODETABLE.
SUB_TYPE	The behavior of the SUB_TYPE attribute value depends on the value of the TYPE attribute. In case where the TYPE attribute is specified as NORMAL, the SUB_TYPE refers to the actual data type to used value; where TYPE value is specified as CODETABLE, the SUB_TYPE value would be the name of the codetable to use.
QUESTION_ID	Unique identifier for this element.
LABEL	Label for the question.

Table 22: Attributes of Evidence Review Question Element (continued)	
Attribute	**Description**
CHOSEN_ANSWER_ID	Pre-defined answer suggestion or n/a if no answer suggestion.

Table 23: Attributes of Evidence Review Answer Element	
Attribute	**Description**
ANSWER_ID	Unique identifier for this element.
NAME	Specifies source of answer option.
DESCRIPTION	Qualifies the option source.
VALUE	The value of the question provided by named and qualified source.

Collapsible Cluster Support: Collapsible clusters are not supported for any cluster containing this widget.

Typical Picture Editor

The Typical Picture Editor is used to set typical picture values for questions in a questionnaire. It displays questionnaire details (name and version) and the typical answers for the questionnaire, which can then be edited by adding values or min/max parameter depending on the data type for the answer.

The UIM FIELD should have a target connection to an "action phase" server interface based on the domain TYPICAL_PICTURE_EDITOR_XML.

Typical Picture XML Format
The Typical Picture Editor Widget requires XML data that conforms to the typical picture editor schema. This schema is defined in the typical-picture-editor.xsd file located in the lib\curam\xml \schema folder of your CDEJ installation folder.

Below is an example of the kind of XML necessary:

```
<TYPICAL_PICTURE SHOW_MIN_MAX="true">
  <QUESTION QUESTION_ID="Q1"
          LABEL="What treatment is prescribed?"
          TYPE="NORMAL" SUB_TYPE="SOME_CODE_TABLE_DOMAIN"
          VALUE="CT3"/>
  <QUESTION QUESTION_ID="Q2" LABEL="Another Question?"
          TYPE="CODETABLE" SUB_TYPE="CodeTableBame"
          VALUE="PT3"/>
  <QUESTION QUESTION_ID="Q3"
          LABEL="Does the claimant smoke smoke?"
          TYPE="NORMAL" SUB_TYPE="BOOLEAN_TYPE"
          VALUE="true"/>
  <QUESTION QUESTION_ID="Q4"
          LABEL="How many cigarettes smoked per day?"
          TYPE="NORMAL" SUB_TYPE="NUMERIC_DOMAIN" VALUE="15"
          MIN="5" MAX="20"/>
</TYPICAL_PICTURE>
```
Figure 3: Typical Picture Editor

Table 24: Attributes of Typical Picture Editor Question Element	
Attribute	**Description**
SHOW_MAX_MIN	Flag to indicate whether the maximum and minimum values will be shown.
QUESTION_ID	Unique identifier for this element.

Table 24: Attributes of Typical Picture Editor Question Element (continued)	
Attribute	**Description**
LABEL	Label for the questionnaire
TYPE	The type is either NORMAL or CODETABLE.
SUB_TYPE	The behavior of the SUB_TYPE attribute value depends on the value of the TYPE attribute. In case where the TYPE attribute is specified as NORMAL, the SUB_TYPE refers to the actual data type to used value; where TYPE value is specified as CODETABLE, the SUB_TYPE value would be the name of the codetable to use.
VALUE	The answer value to the question.
MIN	The minimum answer value to the question.
MAX	The maximum answer value to the question.

When the page is submitted the widget will populate the target field with different XML, for example:

```
<TYPICAL_PICTURE_XML>
  <QUESTION QUESTION_ID="Q1" VALUE="CT1"/>
  <QUESTION QUESTION_ID="Q2" VALUE="PT2"/>
  <QUESTION QUESTION_ID="Q3" VALUE="false"/>
  <QUESTION QUESTION_ID="Q4" VALUE="10" MIN="9" MAX="50"/>
</TYPICAL_PICTURE_XML>
```

Figure 4: Typical Picture Editor Response

Evidence Tab Container Widget

The Evidence Tab Container Widget allows users to use a set of tabs where each tab represents a questionnaire. Each tab contains a list of questions and answers about that questionnaire.

The UIM FIELD should have a target connection to an "action phase" server interface based on the domain EVIDENCE_TAB_CONTAINER_XML.

Evidence Tab Container XML Format
The Evidence Tab Container Widget requires XML data that conforms to the evidence tab container schema.

This schema is defined in the evidence-tab-container.xsd file located in the lib/curam/xml/ schema folder of your CDEJ installation folder. Below is an example of the kind of XML necessary:

```
<EVIDENCE_TAB_CONTAINER DETERMINATION_DELIVERY_ID="12345">
  <QUESTIONNAIRE QUESTIONNAIRE_ID="QE1" LABEL="Questionnaire 1">
    <QUESTION TYPE="NORMAL" SUB_TYPE="INT32" QUESTION_ID="Q1"
            LABEL="What is your gender?">
      <ANSWER DISPLAY_TEXT="Question Not Asked" VALUE="1000"
            SHOW_LINK="true"/>
    </QUESTION>
    <QUESTION TYPE="NORMAL" SUB_TYPE="COMMENTS" QUESTION_ID="Q2"
        LABEL="Do you have trouble breathing when you walk?">
      <ANSWER VALUE="yes" SHOW_LINK="false"/>
    </QUESTION>
  </QUESTIONNAIRE>
  <QUESTIONNAIRE QUESTIONNAIRE_ID="QE2" LABEL="Questionnaire 2"
            SELECTED="true">
    <QUESTION TYPE="NORMAL" SUB_TYPE="INT32" QUESTION_ID="Q3"
            LABEL="Question 8?">
      <ANSWER VALUE="100000" SHOW_LINK="false"/>
    </QUESTION>
    <QUESTION TYPE="NORMAL" SUB_TYPE="CAR_CODE" QUESTION_ID="Q4"
            LABEL="What car do you drive?">
      <ANSWER VALUE="CR1" SHOW_LINK="false"/>
    </QUESTION>
    <QUESTION TYPE="CODETABLE" SUB_TYPE="Cars" QUESTION_ID="Q5"
            LABEL="First car you drove?">
      <ANSWER VALUE="CR2" SHOW_LINK="true"/>
    </QUESTION>
    <QUESTION TYPE="NORMAL" SUB_TYPE="BOOLEAN_TYPE"
            QUESTION_ID="Q6" LABEL="Are you a citizen?">
      <ANSWER VALUE="true" SHOW_LINK="false"/>
    </QUESTION>
  </QUESTIONNAIRE>
</EVIDENCE_TAB_CONTAINER>
```

Figure 5: Evidence Tab Container

Table 25: Attributes of Evidence Tab Comparison Element	
Attribute	**Description**
DETERMINATION_DELIVERY_ID	Unique identifier for this element, supplied as a parameter to the row hyperlink.

Table 26: Attributes of Questionnaire Element	
Attribute	**Description**
QUESTIONNAIRE_ID	Unique identifier for this element, supplied as a parameter to the row hyperlink.
LABEL	Label for the questionnaire or question.

Table 27: Attributes of Question Element	
Attribute	**Description**
QUESTION_ID	Unique identifier for this element, supplied as a parameter to the row hyperlink.
SELECTED	Which tab is in view when loaded.
TYPE	The type is either NORMAL or CODETABLE.

Table 27: Attributes of Question Element (continued)	
Attribute	**Description**
SUB_TYPE	The behavior of the SUB_TYPE attribute value depends on the value of the TYPE attribute. In case where the TYPE attribute is specified as NORMAL, the SUB_TYPE refers to the actual data type to used value; where TYPE value is specified as CODETABLE, the SUB_TYPE value would be the name of the codetable to use.

Table 28: Attributes of Answer Element	
Attribute	**Description**
VALUE	The answer text to the question.
SHOW_LINK	Flag to create a link from existing answer to another.
DISPLAY_TEXT	This text takes precedence over answer value.

Configuration

The Evidence Tab Container can be configured by setting the CONFIG attribute on the UIM FIELD in question.

The EvidenceTabContainerConfig.xml XML configuration file must contain a configuration section with a unique identifier matching the text in the CONFIG attribute. Where the field has not been configured on the UIM, the default will be used. The configuration can be used to change the link URL. The parameter names are stored in the configuration file and the parameter values are stored in the XML data. The following is a sample of this file:

```
<EVIDENCE_TAB_CONTAINER_CONFIG
    DEFAULT="MyConfiguration">
  <CONFIG ID="MyConfiguration">
    <ANSWER_SOURCE_LINK
      PAGE_ID="EvidenceComparisonViewAnswerSource"
      QUESTIONNAIRE_ID_PARAM="questionnaireID"
      QUESTION_ID_PARAM="questionID"
      DETERMINATION_DELIVERY_ID_PARAM="determinationDeliveryID"/>
  </CONFIG>
</EVIDENCE_TAB_CONTAINER_CONFIG>
```

Figure 6: Evidence Tab Container Configuration

Table 29: Attributes of Evidence Tab Container Config Element	
Attribute	**Description**
DEFAULT	The configuration to be used if it is not specified in the UIM.

Table 30: Attributes of Config Element	
Attribute	**Description**
ID	The unique identifier for the configuration.

Table 31: Attributes of Answer Element	
Attribute	**Description**
PAGE_ID	The ID of the target page for the link.
QUESTIONNAIRE_ID_PARAM	Parameter name for the link.
QUESTION_ID_PARAM	Parameter name for the link.
DETERMINATION_DELIVERY _ID_PARAM	Parameter name for the link.

Collapsible Cluster Support: Collapsible clusters are not supported for any cluster containing this widget.

Decision Assist Integration

Cúram Decision Assist integrates with Cúram assessments, integrated cases, and Intelligent Evidence Gathering (IEG). This chapter describes these integration points.

The Decision Assist Integration chapter also describes how the Determination Package entity integrates with existing case and participant information for matching a Determination Package to a client.

Component Dependencies

Cúram Decision Assist depends on the IBM Cúram Social Program Management Platform, Intelligent Evidence Gathering (IEG) and Cúram Data Store (CDS) components.

IBM Cúram Social Program Management Platform is used to integrate Decision Assist with Assessments. IEG is used to create scripts for the Decision Assist questionnaires. CDS is used to create schemas, data store entities, and attributes that are used for the IEG scripts.

Decision Assist Integration with Assessments

Cúram Decision Assist integrates with Cúram Assessments making existing assessment functionality available to Determination Based Assessments. This includes the storage of general assessment details, client name, assessment name, reference, type, assessment date, and status.

The Decision Assist assessment type includes its own set of pages and extended functionality unique to Decision Assist assessments. There is a page which displays a list of Decision Assist assessments; there is also a page which displays information specific to Decision Assist assessments.

When a user selects to view an assessment of type decision assist, the user is presented with the usual assessment information, as well as with the list of determination configurations associated with the assessment. A user can delete or add existing determination configurations to the assessment. Note, however, if the first determination configuration associated with the assessment is marked as a stand alone determination configuration, then no more determination configurations can be associated with the assessment.

Assessment actions which are available to decision assist assessments include cloning and canceling. There are also a number of decision assist specific assessment actions including the ability to make a decision and complete assessment.

Updates were made to the existing Cúram Assessment statuses to include Decision Assist statuses. There is also a view history option which returns a status history. This view history option is available to all assessment types.

A hook has been implemented in Cúram Decision Assist which allows Decision Assist modules to be registered. The hook is off the CreateAssessment business process and allows Decision Assist to be extended on a per module basis. This means that the same methods can be in each module's interface, but can be customized according to what is required for that component.

Decision Assist integration with Integrated Cases

Cúram Decision Assist integrates with Cúram Integrated Cases. The entity `PDAssessmentConfiguration` links assessment configurations to products. This entity supports the association of assessments to both integrated cases and stand-alone product deliveries

As part of system administration, users can access a list of assessment configurations that are associated with an integrated case and add or remove associations

At the case level, a user can create an Integrated Case of any type, link to assessments within that Integrated Case. The clients from the Integrated Case are selectable as the primary client of the decision assist assessment. The assessment type is selected from the list of decision assist assessments.

Alternatively, the initially configured Sample Benefit Product has a link to Assessments from its home page that displays a list of Assessments that are associated with the Sample Benefit Product. Selecting **New** on this list page opens the **Create Assessment** page. This action is different from Create Assessment for Integrated Case, as the primary client defaults to that of the Sample Benefit Product and is not selectable.

Questionnaire Integration with IEG

Cúram Decision Assist integrates with Cúram Intelligent Evidence Gathering (IEG). This integration allows for the creation of and maintenance of Decision Assist questionnaires by using the IEG editor. This process includes creating new questionnaire sections, pages, clusters, questions, and question aliases.

The required data store entities, attributes, and the data types of these attributes are configured by using Data Store Editor. These data store entities and their attributes then can be used for the question pages and questions in the IEG editor. For more information on Data Store features, see the *Creating Datastore Schemas* guide.

Question aliases are used to rephrase questions in different ways for different audiences. To configure a question alias, it is necessary to enter how the question needs to appear according to the alias type. For example, the question, "suicidal or risk of self harm," might be presented to a third-party medical representative, such as a doctor. The wording of the question for the doctor reads, "Based on your medical evaluation, is this person suicidal or at risk for self harm?"

Questionnaires are associated with determination packages. As part of Determination Based Assessment processing, IEG runtime functions are used to complete and to store the answers. These answers then can be used in decision making.

IEG runtime is started from the Answer link on the **View Assessment Details** page. The first question group in the decision assist questionnaire is presented to the user in IEG runtime. The next button will bring the user to the next question group in the decision assist IEG script that is created as part of decision assist administration.

All features of the IEG editor and IEG run time are available to maintain Decision Assist questionnaires. For more information on these features, see the *Authoring Scripts using IEG* guide.

Localization Support
Localization support is provided for questionnaires. Using the Cúram intelligent evidence gathering (IEG) editor, questionnaire scripts can be translated into different languages. This process includes translations for question pages and questions. When a questionnaire script is run in a locale other than the default locale, the questions, the script text, legislation links, and help on the questions can all be set up to appear in translated languages.

`TransactionInfo.getProgramLocale()` is used to determine the locale for that particular execution. To run the questionnaire script in the appropriate language, the method initiates the following actions:

- It uses the locale that is pushed from the client that comes from the URL.
- If that is not there for some reason, for example, likely for WebServices, batch, the method checks the Users table and uses the default locale.

- If the User is not on the Users table, it calls an application programming interface (API), which is implemented by the application to return the information for external users.

Determination Package Integration with Case and Participant Information

Cúram Decision Assist supports the association of multiple determination packages with the same Decision Assist assessment. The purpose of integrating Determination Packages with existing case and participant information is to be able to use this information in order to select the determination package best suited to meet a client's needs.

For example, the Asthma Medical Assessment has two determination packages: one for male clients and one for female clients. In this example, it makes sense that male and females would be evaluated differently when it came to their healthcare needs.

To support this integration, the Determination Package entity includes the following attributes which act as parameters for matching a client with a determination package:

- MinAge and MaxAge - if these attributes are set, then the determination package will only be used to assess the primary client provided that primary client's age falls within the age parameters
- Product - if this attribute is set, then only the determination package associated with the related product delivery case will be used to assess the primary client
- Gender - if this attribute is set, then the determination package will only be used to assess the primary client provided that the gender parameter matches the primary client's gender

The remainder of this section describes the process of matching a determination package to a client; it also describes in more detail the rules used during this matching process.

Process of Matching a Determination Package to a Client

The process of matching a determination package to a client begins with the retrieval of case and participant information for that client using his or her participant ID and case IDs (where the client is the primary client of those cases).

It is important to note that the latest released Determination Package versions for the effective date are considered.

The system compares the information retrieved to the attributes associated with each determination package associated with the Decision Assist assessment. The process then diverges depending on whether or not the Decision Assist assessment is associated with an integrated case or a product delivery case.

If maintained at the Integrated Case level, the system continues as follows:

1. The system looks for Determination Packages where the Product attribute is blank. If all the determination packages have the Product attribute set, then an informational message is returned indicating that no match could be made.
2. The system checks for age and/or gender matching by looking at the client's age and gender information and the Determination Package's Age and/or Gender attributes. If all the determination packages have these attributes set, but they do not match the client's information, then an informational message is returned indicating that no match could be made.
3. If a determination package has both an age and gender setting which matches the client's age and gender, then this will be matched to the client.
4. Alternatively, the system checks for determination packages with no gender, and/or age attributes which conflict and makes a match accordingly.
5. If no match can be made, an informational message is returned.

If the Decision Assist assessment is maintained at the product delivery level, then the process for matching a determination package is as follows:

1. The system first tries to match the Product attribute for a determination package to any products the client of the decision assist is a primary client of.

2. If the determination package has a matching product, the system then checks for age and/or gender matching. If no match is found, then an informational message is returned indicating that no match could be made.

3. Alternatively the system checks for determination packages with no product, gender, and/or age attributes which conflict and makes a match accordingly. If a determination package has no attributes set, then this will be matched to the client.

4. If no match can be made, an informational message is returned.

Determination Package Matching Rules
To support the process of matching a Determination Package to a client, validations have been added to the methods for creating and updating Determination Packages which ensure the following rules are followed:

• Only one Determination Package associated with a Decision Assist assessment can have all three attributes as null, i.e. not entered. Validation will be thrown if a user attempts to add a second Determination Package without entering information for at least one of these attributes.

• Only one Determination Package can exist with a particular Product and no Gender and Age specified. Validation will be thrown if a user attempts to add a second Determination Package which contains the same Product setting and which does not have a Gender or Age specified.

• The same applies if there is a Gender but no Product or Age specified or if there is an Age but no Gender or Product specified.

• Many Determination Packages can exist with a particular Product and Age and no Gender specified, if Age overlap does not occur and if no Determination Package exists with Product and Gender specified with no Age.

• When Age is specified and no Product and no Gender specified, many Determination Packages can exist as long as the Age ranges do not overlap.

• When all three optional criteria are specified, many packages can exist, if there is no overlap in Age, Gender, and Product.

Important: If the matching criteria for a determination package configuration is changed as part of decision assist administration, all determination based assessments which include determination packages based on the changed determination package configuration cannot be cloned. This is to ensure that the determination package matching rules are not broken within cloned assessments.

Decision Assist Batch Processes

This information lists the batch processes which provide Decision Assist functionality. The following sections list the Cúram Decision Assist batch processes that are configurable in the system administration application.

Batch Parameters and Processing Date

Quite a number of the batch jobs within the application take 'processingDate' as a parameter. When this is specified, it will be the date the batch job is run for. When not specified, the batch job is run for the system date.

BatchUploadICDCodes

This is the batch process that is used to perform batch upload of the ICD Code files that contain the classification of the medical conditions.

This batch job populates ICDCODE and ICDCODEVERSION tables. Following are the input parameters that are passed to this batch :

Table 32: BatchUploadICDCodes Input Parameters

Input Parameter	Description
classification	The category of ICD code.
fileName	Name of the ICD codes file to be read.
filePath	Full path of the ICD codes file to be read.
processingDate	Processing date of the batch.
versionNumber	Version no. to be associated with the ICD Code.
volume	Volume of the ICD Code Version.

Batch Process Class and Method: The class and method for this batch process is
ICDCodes.batchUploadICDCodes.

Localization of Question Text

You can retrieve questions from the questionnaire in the QUESTION table and translate the text. Then, you can store the retrieved questions in multiple locations and view them from those locations.

Technical Details

The NAMETEXTID column in the QUESTION table is associated with the LOCALIZABLETEXTID column in the LOCALIZABLETEXT table and TEXTTRANSLATION table.

You can store your question text in the TEXTTRANSLATION table, linked to your location, so you can insert a question or release a questionnaire. You can also retrieve question text from the TEXTTRANSLATION table. If your question text is not found in the TEXTTRANSLATION table when you insert a question or release a questionnaire, then the question text stored in the QUESTION table will be used.

Customization

If you want to view the question text in a location other than the location that is used for inserting the question or releasing the questionnaire, make the relevant translated text available.

To make questions available in multiple locations, complete the following steps:

- Identify the question text from the database through QUESTION table when you insert the question.
- Store the questions in the TEXTTRANSLATION table as shown in the following table.
- The NAMETEXTID column of the QUESTION table is associated with the LOCALIZABLETEXTID column of the LOCALIZABLETEXT and TEXTTRANSLATION tables.
- Create text a translation record for a location, as shown in the following table, to view the translated question text for that location.

Table 33: TEXTTRANSLATION TABLE

TEXTTRANSLA TIONID	LOCALIZABLET EXTID	LOCALECODE	TEXT	UPPERSHORTT EXT	SHORTTEXT
-2345	4455	en_US	null	WERE YOU HAVING FEVER	Were you having fever
6754	4455	fr	null	WARE HAVE FEVERA	ware have fevera

Notices

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

COPYRIGHT LICENSE:

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

Privacy Policy considerations

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

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

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

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

Trademarks

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

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

Java™ and all Java-based trademarks and logos are trademarks or registered trademarks of Oracle and/or its affiliates.

Linux is a registered trademark of Linus Torvalds in the United States, other countries, or both.

Microsoft, Windows, Windows NT, 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 names may be trademarks of their respective owners. Other company, product, and service names may be trademarks or service marks of others.

IBM®

Part Number:

(1P) P/N: