



Cúram v8.2.1.0 Enhancements

Cúram v8.2.1.0 Enablement Material

November 2025

© 2025 Merative



Hello and welcome to this deep dive presentation that introduces the enhancements to Cúram that are available in 8.2.1.0

Agenda

Overview of Cúram v8.2.1.0 Enhancements

- Accessibility Improvements and Updates
- Documentation for Platform and Insurance Affordability Generated Tasks and Notifications (*Updates for all*)
- Enhanced Verification Proof Sharing (*Updates for caseworkers*)
- Application Case Evidence Management (*Updates for caseworkers*)
- OpenJDK Enablement for Batch and XML Servers (*Update for developers*)
- New validation on Application Level Authorization on Application Cases (*Updates for developers*)
- Hook point to specify additional Evidence validations on Application and Program Level Authorizations (*Updates for developers*)
- Pre-customization hook points for Product Delivery lifecycle (*Updates for developers*)
- Display money fields based on user's locale (*Updates for developers*)

Accessibility Improvements

© 2025 Merative

3

There is new legislation both in the US and Canada designed to ensure that users with accessibility needs receive equal access to the systems available to them.

In the US, a new rule was finalized revising Title II of the Americans with Disabilities Act. The rule requires that US state and local governments make their websites and mobile applications accessible for people with disabilities and comes into effect in April 2026.

In Canada, there's the Accessible Canada Act (ACA) which came into force in 2019. The overarching goal of the ACA is to realize a barrier-free Canada by 2040. The legislation benefits all Canadians, especially persons with disabilities, through the proactive identification, removal and prevention of barriers to accessibility.

This section details the Accessibility Improvements now available in 8.2.0.0 in the Caseworker Application

Accessibility Improvements 8.2.1.0 contains 17 accessibility improvements throughout the Cúram application	These changes will benefit many different types of users of Cúram. The following sections detail before and after screens and descriptions (where applicable). In addition, there are also some technical details regarding the change, and if any specific upgrade steps are required, this is also noted. Note: If a client has customized any of the pages mentioned, the changes made by Cúram will need to be taken into consideration in the client's code. Refer to the External Release Notes for additional details for any of the items listed below.
--	---

The following sections of this enablement material details before and after screens and descriptions, where applicable.

In addition, you will also find some technical details regarding the improvement made and if any specific upgrade steps are required to take advantage of the improvements.

If a client has customized any of the pages mentioned in the technical details' sections, these changes will need to be taken into consideration in the client's code.

To read additional information about any of the improvements, refer to the External Release Notes.

List of Accessibility issues that have been improved

1. Toggling between high contrast mode in Windows does not update on all screen elements
2. Hover state of the shortcut menu option has insufficient contrast with its adjacent background color
3. Legend text at the top of pages with mandatory fields is not descriptive
4. A descriptive label for a participant's age should be provided to clarify that the number represents the participant's age.
5. Focus order for the shortcuts panel is not working as expected for a keyboard user
6. The initial keyboard focus is not set correctly on view-only modal dialogs
7. Additional improvements for screen reader users (11 items)

List of Accessibility Improvements

1. Toggling between high contrast mode in Windows does not update on all screen elements
2. Hover state of the shortcut menu option has insufficient contrast with its adjacent background color
3. Legend text at the top of pages with mandatory fields is not descriptive
4. A descriptive label for a participant's age should be provided to clarify that the number represents the participant's age.
5. Focus order for the Shortcuts panel is not working as expected for a keyboard user
6. The initial keyboard focus is not set correctly on view-only modal dialogs

7. Additional improvements for screen reader users (11 items)

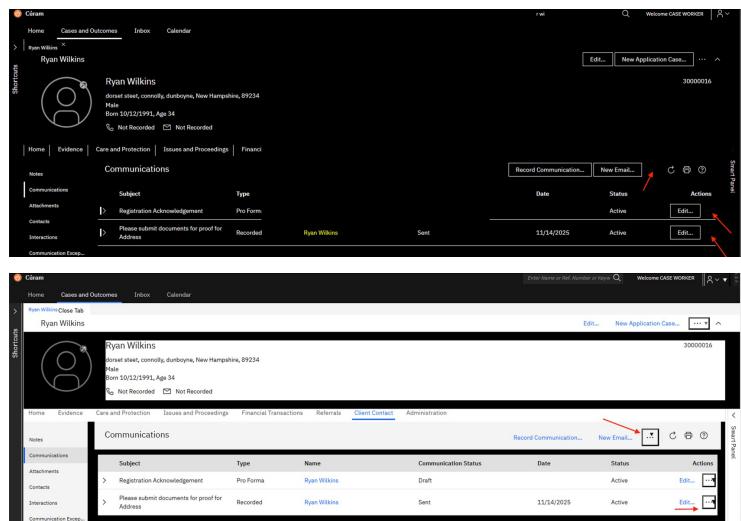
There is continued focus on accessibility. There are 17 Accessibility defects addressed in the 8.2.1.0 release. These cross many different parts of the application and are beneficial to many different types of users. There are improvements for sighted, screen reader, and keyboard users. This deep dive provides details in relation to 4 of these improvements. Items listed number 1 to 4, relate to individual accessibility enhancements. Numbers 5 and 6 correspond to improvements relating to Focus. Lastly number 7 is a list of all 11 screen reader Improvements. To see all accessibility improvements and further details on those contained in this material, consult the Accessibility section in the External Release Notes. If you want to review the WCAG category that is now achieved by each of the Improvements listed, see the Launch Readiness document.

1. Toggling between high contrast mode in Windows does not update on all screen elements

Before 8.2.1.0

Switching Windows' high contrast mode on or off does not update some on-screen elements to the correct contrast mode.

Icons for Page Actions menu and inline List Actions menu were not displayed properly when switching on / off high contrast mode.



© 2025 Merative

6

Toggling between high contrast mode in Windows does not update all on screen elements

1. Toggling between high contrast mode in Windows does not update on all screen elements

In 8.2.1.0

The Cúram application now properly refreshes visual elements to match the selected Windows' high contrast mode.

Icons for Page Actions menu and inline List Actions menu are automatically displayed when switching on / off high contrast mode.

Refreshing the browser is not required.

The screenshots illustrate the behavior of the Cúram application when high contrast mode is toggled on and off in Windows. The first screenshot shows the standard dark-themed UI. The second screenshot shows the application in high-contrast mode, where icons and text are white on a dark background. The third screenshot shows the application has refreshed its UI to match the high-contrast mode, displaying light-colored icons and text on a white background.

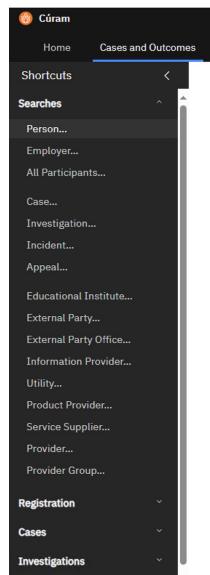
7

Toggling between high contrast mode in Windows does not update all on screen elements

2. Hover state of the shortcut menu option has insufficient contrast with its adjacent background color

Before 8.2.1.0

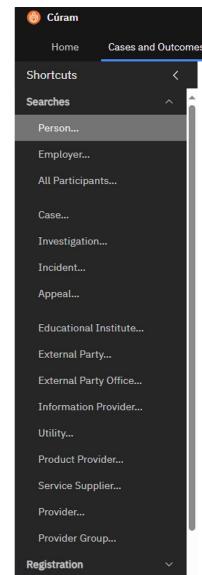
The hover color of the section shortcut panel had insufficient contrast with its adjacent background color



© 2025 Merative

In 8.2.1.0

The hover color of the section shortcut panel has been updated so that it has sufficient contrast with its adjacent background color.



8

Hover state of the shortcut menu option has insufficient contrast with its adjacent colour

Before 8.2.1.0

The hover color of the section shortcut panel has insufficient contrast with its adjacent background color

In 8.2.1.0

The hover color of the section shortcut panel has been updated, so that it has sufficient contrast with its adjacent background color.

3. Legend text at the top of pages with mandatory fields is not descriptive

Before 8.2.1.0

On a modal or wizard that has a form, the required legend displayed to caseworkers to inform them about mandatory fields is too simplistic and lacks clarity.

In 8.2.1.0

Modals and wizards with required input fields now display a legend

*Indicates a required field.

The image contains two side-by-side screenshots of a 'Register Person' form. Both screenshots show the same form structure: 'Registered Person Check' and 'Registration' tabs, 'Step 2: Registration' section, 'Reference Number' input, 'First Name *' input, 'Last Name *' input, 'Title' dropdown, 'Middle Name' input, and 'Suffix' dropdown. In the top screenshot (Before), a red box highlights the text '* required field' in the top right corner. In the bottom screenshot (In), a red box highlights the text '* Indicates a required field' in the top right corner. The bottom screenshot also includes a small number '9' in the bottom right corner.

Legend text at the top of pages with mandatory fields is not descriptive

4. A descriptive label for a participant's age should be provided to clarify that the number represents the participant's age

Before 8.2.1.0

The participant's age was displayed in brackets next to their name, but without a clear label indicating it was their age, potentially causing confusion for both visual and screen reader users about what the number represents.

The screenshot shows a software interface titled 'Edit Gender'. It includes fields for 'Received Date' (set to '10/14/2025'), 'Change Reason' (set to 'Reported by Client'), and 'Effective Date of Change' (a dropdown menu). Below these is a section titled 'Case Participant' containing a table with one row. The table has columns for 'Case Participant' (containing 'James Smith (45)') and 'Actions'. The 'Actions' column contains a red-bordered button labeled 'Edit'. Underneath this is a 'Details' section with a 'Gender' field set to 'Male'. At the bottom are 'Cancel' and 'Save' buttons.

© 2025 Merative

10

Before 8.2.1.0, the participant's age was displayed in brackets next to their name, but without a clear label indicating it was an age, potentially causing confusion for both visual and screen reader users about what the number represents.

4. A descriptive label for a participant's age should be provided to clarify that the number represents the participant's age

In 8.2.1.0

A descriptive label has been added to clarify that the number represents the participant's age, e.g., James Smith, Age: 55.

The screenshot shows a software interface titled 'Edit Gender Details'. It includes fields for 'Received Date' (set to '10/14/2025'), 'Change Reason' ('Reported by Client'), and 'Effective Date of Change' (date field). Below these, a section titled 'Case Participant' contains a list item 'James Smith, Age: 45'. This list item is highlighted with a red rectangular box. Underneath is a 'Details' section with a 'Gender' field set to 'Male'. At the bottom are 'Cancel' and 'Save' buttons, with 'Save' being highlighted in blue.

© 2025 Merative

11

In 8.2.1.0, A descriptive label has been added to clarify that the number represents the participant's age, e.g., James Smith, Age: 55.

5. Focus order for the shortcuts panel is not working as expected for a keyboard user

In 8.2.1.0

The shortcuts panel has been updated so that the focus order moves through the category titles and visible links from top to bottom, as would be expected.

6. The initial keyboard focus is not set correctly on view-only modal dialogs

In 8.2.1.0

The initial keyboard focus is now correctly set when a view-only modal dialog is opened by a caseworker, and also when the text content on the modal receives focus.

© 2025 Merative

12

5. Focus order for the shortcuts panel is not working as expected for a keyboard user

In 8.2.1.0

The shortcuts panel has been updated so that the focus order moves through the category titles and visible links from top to bottom, as would be expected.

6. The initial keyboard focus is not set correctly on view-only modal dialogs

In 8.2.1.0

The initial keyboard focus is now correctly set when a view-only modal dialog is opened by a caseworker, and also when the text content on the modal receives focus.

CÚRAM ACCESSIBILITY

Improvements for screen reader users

© 2025 Merative

13

Improvements for screen reader users

This section groups the list of screen reader improvements to aid the visually impaired users in navigating through the Cúram Case worker application

The following screen reader improvements are available in 8.2.1.0

Problem	Resolution	WCAG
1 The purpose of List actions is not clear in their accessible name	The aria-label for inline action menu buttons now includes additional contextual information, e.g., 'Add Addresses Row 2'.	1.3.1
2 The label for the toggle button that is used to expand rows in a list doesn't describe its purpose clearly	The aria-label for the toggle button in the details row in an expandable list now includes additional contextual information, e.g., 'Toggle details, Online Application Received Work Queue, Row 2'.	1.3.1
3 Input instructions before fields are not announced when modals have multiple ways to select a participant	The affected screens have been updated so that screen readers now announce the instructional text when reading associated form fields.	1.3.1
4 Non-relevant reading by screen reader due to incorrect aria-label on the Reference Number input field on Person Registration	The Reference Number input field on the Register Person wizard is now correctly communicated by the screen reader.	1.3.1
5 The hierarchical levels of nested tables are not programmatically available	Expandable lists within the application are now fully accessible. Screen reader and keyboard users can reliably identify, navigate and understand content in nested expandable lists and in-page navigation tabs, ensuring a clearer and more inclusive experience.	1.3.1

The first 5 screen reader improvements are described by detailing the Problem and the resolution applied. The associated WCAG Checkpoint reference is also supplied for each item.

The following screen reader improvements are available in 8.2.1.0

Problem	Resolution	WCAG
6 Related checkbox fields in a cluster are not programmatically grouped on the Case Search page	Semantically related checkbox fields can now be programmatically grouped to provide context to screen reader users.	1.3.1
7 Remove unnecessary “*required field” messages from screen readers and hide them on screens without mandatory fields	The “*required field” message now only appears on pages with mandatory fields and is hidden from screen readers.	1.3.1
8 IEG: Hidden content (Skip to Form Content link) is readable with screen reader	The ‘Skip to main content’ link now becomes visible when users tab to it. It reliably moves the user into the main form area; it no longer causes confusing announcements for screen readers.	1.3.2
9 An incorrect aria-label is applied to the text-area element when a text-area element has no label specified in a UIM	When the user navigates to the Description text area for New Assessments, the screen reader correctly communicates ‘Description edit multi-line’.	2.4.6
10 The button accessible name ‘Modal Frame’ is announced instead of ‘Contact Details’	Screen readers will correctly read out relevant and meaningful context for the button	2.5.3
11 Screen Reader reading Navigation Left & Navigation Right icons which should be hidden	The navigation tab buttons (Navigation Left and Navigation Right buttons) are excluded from the accessibility tree, ensuring they are not announced by screen readers.	4.1.2

© 2025 Merative

15

The screen reader improvements (numbered 6 to 11) are described by detailing the problem and the resolution applied. The associated WCAG Checkpoint reference is also supplied for each item



CÚRAM

Cúram caseworker application

© 2025 Merative

16

The font size in the Cúram Caseworker application has been increased to improve readability

Based on customer feedback, the font size in key areas of the caseworker application has been increased from 12px to 14px to improve readability. This update supports accessibility and aligns with design best practices.

CÚRAM - UPDATES FOR ALL

Documentation for Platform and Insurance Affordability Generated Tasks and Notifications

© 2025 Merative

17

Documentation for Platform and Insurance Affordability Generated Tasks and Notifications

Business Problem

Customers have reported that too many out-of-the-box (OOTB) Tasks and Notifications are created in Cúram by default, leading to unmanageable task listings and work queues. This made it difficult for caseworkers to prioritize relevant actions and maintain efficient workflows.

Solution

Comprehensive documentation for all Platform and Insurance Affordability generated Tasks and Notifications is now available. With this documentation, customers can understand which OOTB tasks and notifications are generated by Cúram, identify which are relevant to their business workflows and configure the system to generate only the tasks and notifications that they require. This reduces wasted time and effort on managing redundant tasks and improves operational efficiency.

Before 8.2.1.0

There was no comprehensive documentation describing the Platform and Insurance Affordability Tasks and Notifications generated by the Cúram.

In 8.2.1.0

- Documentation for all Platform and Insurance Affordability generated Tasks and Notifications is now available. This documentation includes: Task or notification subject, business processes that initiate the creation, related workflow processes and configurability status with relevant property names.
- With this documentation, customers can understand which OOTB Tasks and Notifications are generated by Cúram, identify which are relevant to their business workflows and configure the system to generate only the Tasks and Notifications that matter, helping caseworkers and managers focus on the correct set. This reduces wasted time and effort on managing redundant tasks and improves operational efficiency.
- The Cúram Tasks and Notifications documentation is available from [Merative Support Portal](#). You must log in to access this documentation, request access if needed. Enter your credentials and navigate to Knowledge Base then Article Search, select "Curam Knowledge" as the Data Category Group and then select the relevant "Cúram Tasks and Notifications" Data Category.

Customers have reported that too many out-of-the-box (OOTB) Tasks and Notifications are created in Cúram by default, leading to unmanageable task listings and work queues. This made it difficult for caseworkers to prioritize relevant actions and maintain efficient workflows.

To address this gap, comprehensive documentation for all Platform- and Insurance Affordability -generated Tasks and Notifications is now available. With this documentation, customers can understand which OOTB tasks and notifications are generated by Cúram, identify which are relevant to their business workflows and configure the system to generate only the tasks and notifications that they require. This reduces wasted time and effort on managing redundant tasks and improves operational efficiency.

Prior to 8.2.1.0, there was no comprehensive documentation describing the Platform and Insurance Affordability Tasks and Notifications generated by the Cúram.

In 8.2.1.0, detailed documentation for all Platform and Insurance Affordability generated Tasks and Notifications is now available. This documentation

includes: Task or notification subject, business processes that initiate the creation, related workflow processes and configurability status with relevant property names. With this documentation, customers can understand which OOTB Tasks and Notifications are generated by Cúram, identify which are relevant to their business workflows and configure the system to generate only the Tasks and Notifications that matter helping caseworkers and managers focus on the correct set. This reduces wasted time and effort on managing redundant tasks and improves operational efficiency.

The Cúram Tasks and Notifications documentation is available from [Merative Support Portal](#). You must log in to access this documentation, request access if needed. Enter your credentials and navigate to Knowledge Base then Article Search, select "Curam Knowledge" as the Data Category Group and then select the relevant "Cúram Tasks and Notifications" Data Category.

CÚRAM - UPDATES FOR CASEWORKERS

Enhanced Verification Proof Sharing

© 2025 Merative

19

Enhanced Verification Proof Sharing

Business Problem

Previously, a verification proof was only shared when evidence was activated or in limited scenarios where a verification is satisfied on an application case. If proof was added after activation, it was not automatically shared to linked evidence records on target cases. This required caseworkers to manually record the verification proof on the target case, increasing workload and introducing the risk of errors and inconsistencies.

Solution

With this enhancement, caseworkers can now add verification proof to active evidence on any case type such as Product Delivery Case, Integrated Case, or Application Case either manually or by accepting documents submitted by the citizen. The system automatically shares the verification proof to linked evidence on configured target cases, eliminating the need for duplicate data entry and reducing errors.

Before 8.2.1.0

Verification proof was only shared when evidence was activated. Any proof added after activation had to be manually recorded on linked target cases, creating inefficiencies and potential inconsistencies.

In 8.2.1.0

- **Configuration Option:** Administrators can enable sharing of verification proof added post-activation using the `curam.aes.enableShareVerification.postActivation` property (default = True)
- **Automatic Sharing:** Verification proof added to active evidence is automatically shared with all linked target cases configured for verification sharing
- **Hook Point:** A new hook point triggers when verification proof is shared, allowing customers to implement custom logic (e.g., notifications or additional workflows)

This enhancement improves efficiency, reduces manual effort, and ensures consistency across cases.

Previously, a verification proof was only shared when evidence was activated or in limited scenarios where a verification is satisfied on an application case. If proof was added after activation, it was not automatically shared to linked evidence records on target cases. This required caseworkers to manually record the verification proof on the target case, increasing workload and introducing the risk of errors and inconsistencies.

The new enhancement resolves this gap - caseworkers can now add verification proof to active evidence on any case type, such as Product Delivery Case, Integrated Case, or Application Case either manually or by accepting documents submitted by the citizen. The system automatically shares the verification proof to linked evidence on configured target cases, eliminating the need for duplicate data entry and reducing errors.

Before 8.2.1.0, Verification proof was only shared when evidence was activated. Any proof added after activation had to be manually recorded on linked target cases, creating inefficiencies and potential inconsistencies.

In 8.2.1.0, Administrators can enable sharing of verification proof added post-activation using the `curam.aes.enableShareVerification.postActivation` property (default = True). Verification proof added to active evidence is automatically shared with all linked target cases configured for verification sharing. A new hook point triggers when verification proof is shared, allowing customers to implement custom logic (e.g., notifications or additional workflows).

This enhancement improves efficiency, reduces manual effort, and ensures consistency across cases.

Enhanced Verification Proof Sharing

System Administrator – new system admin property

- To support post-activation verification sharing, a new system admin property, Enable Verification Sharing Post Activation has been introduced. By default, this property is set to True.
- This configuration applies only to evidence where verification sharing is set to 'Always'.
- When enabled, any verification item added to an active evidence record on the source case is automatically shared with all target cases that are linked, both directly and indirectly, by configuration to the source case.

The screenshot shows the Curam Property Administration interface. In the search criteria, 'enable verification sharing' is entered. The results table shows one item:

Name	Value	Description	Category	Actions
Enable Verification Sharing Post Activation	true	If this value is set to true, verification sharing will apply post-activation of evide...	Application - Miscellaneous settings	Edit Value... ...

Below the table, there is a note: "If this value is set to true, verification sharing will apply post-activation of evidence. This property is only applicable to evidence where verification sharing is 'Always'." The value is listed as true. Under Technical Information, the ID is curam.ase.enableShareVerification_postActivation and the Type is Boolean Type.

To support post-activation verification sharing, a new system admin property, Enable Verification Sharing Post Activation has been introduced. By default, this property is set to True.

Note, This configuration applies only to evidence where verification sharing is set to 'Always'.

When enabled, any verification item added to an active evidence record on the source case is automatically shared with all target cases that are linked, both directly and indirectly, by configuration to the source case.

Enhanced Verification Proof Sharing

Scenario

- An agency has separate integrated cases for Food Assistance and Cash Assistance. The same benefit evidence is configured for both integrated cases.
- The agency has sharing configured, Trusted Source is set to 'Yes' and Share Verifications is set to 'Always' for the Benefit evidence between the Food Assistance and Cash Assistance Integrated cases.
- Optional Verifications are configured for the Benefit evidence.

The screenshot shows the Carem Administration Workspace interface. On the left, there's a sidebar with various menu items like Home, Administration Workspace, Sites, Calendar, Participants, Case, Rules and Evidence, Intelligent Evidence Gather..., Dynamic UEM, Verifications, Service Plans, Case Audit, and Workflow. The main area is titled 'Evidence Sharing' and 'Configurations'. It shows a 'Sharing Configuration' table:

Source	Target	Actions
Cash Assistance IC (IC)	Food Assistance IC (IC)	Add Identical Evidence... Edit... ...

Below this, it says: "In this configuration, the following types of identical evidence are shared." Then there's another table:

Evidence Type	Treated Source	Share Verifications	Actions
Benefit	Yes	Always	Edit... ...

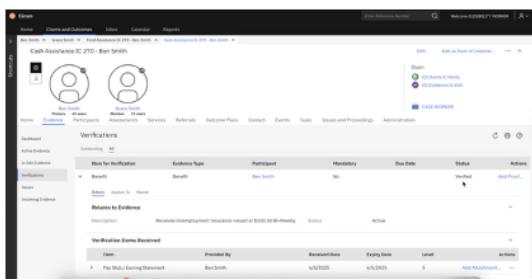
This scenario demonstrates how the enhancement is applied in an agency setup. In this scenario, the agency manages two separate Integrated Cases — one for Food Assistance and another for Cash Assistance. Both cases use the same Benefit evidence type, which is configured to be shared between the two case types. The agency has sharing configured, with Trusted Source set to Yes and Share Verifications set to Always, ensuring any verification proof attached to the Benefit evidence is automatically shared across both cases.

Additionally, Optional Verifications are configured for the Benefit evidence.

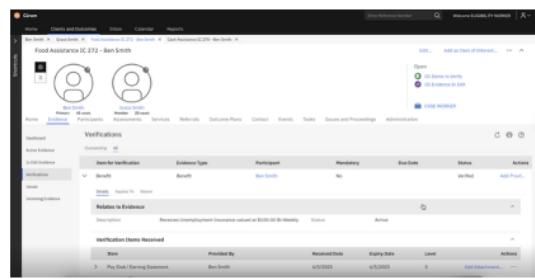
Enhanced Verification Proof Sharing

Caseworker updates

Verification Proof is added to Cash Assistance Integrated Case post activation of Benefit evidence.



Verification Proof is now shared to the linked Benefit evidence on the Food Assistance Integrated Case, and is also redetermined as Verified.



© 2025 Merative

23

A citizen, Ben, has a Food Assistance Integrated Case with his daughter. Ben subsequently contacts the agency to apply for Cash Assistance. The caseworker processes his application and sets up a separate case of Cash Assistance.

Ben contacts the agency and reports that he is receiving 'Unemployment Insurance' Benefit. The caseworker records this new Benefit evidence on Ben's Cash Assistance IC.

The Benefit evidence is created with an outstanding verification.

The caseworker activates the evidence and navigates to Ben's Food Assistance IC and sees that the Benefit evidence and the outstanding verification is added to the case. Because the verification proofs required are optional as per the configuration settings, the caseworker applies evidence changes without adding verification proofs to activate the evidence

On activation, the Benefit evidence is shared to the Food Assistance IC as active evidence with the outstanding verification.

The agency receives verification Proof from Ben. The caseworker adds the verification

proof to the Cash Assistance IC and saves.

The status of the verification is re-determined as Verified.

On the Food Assistance IC, the caseworker sees that the verification proof applied to Cash Assistance IC is now shared to the linked Benefit evidence on the Food Assistance IC and the verification is also re-determined as Verified.

Previously, the uploaded verification proofs added post-evidence activation were not shared to the target cases. Caseworkers had to manually upload the proofs again.

With the new enhancement, the verification proofs are shared to the linked target case automatically without caseworker intervention.

Enhanced Verification Proof Sharing

Upgrade Implications

Before 8.2.1.0

Verification proof added before activation will be shared. Any verification proofs added post-activation will not be shared.

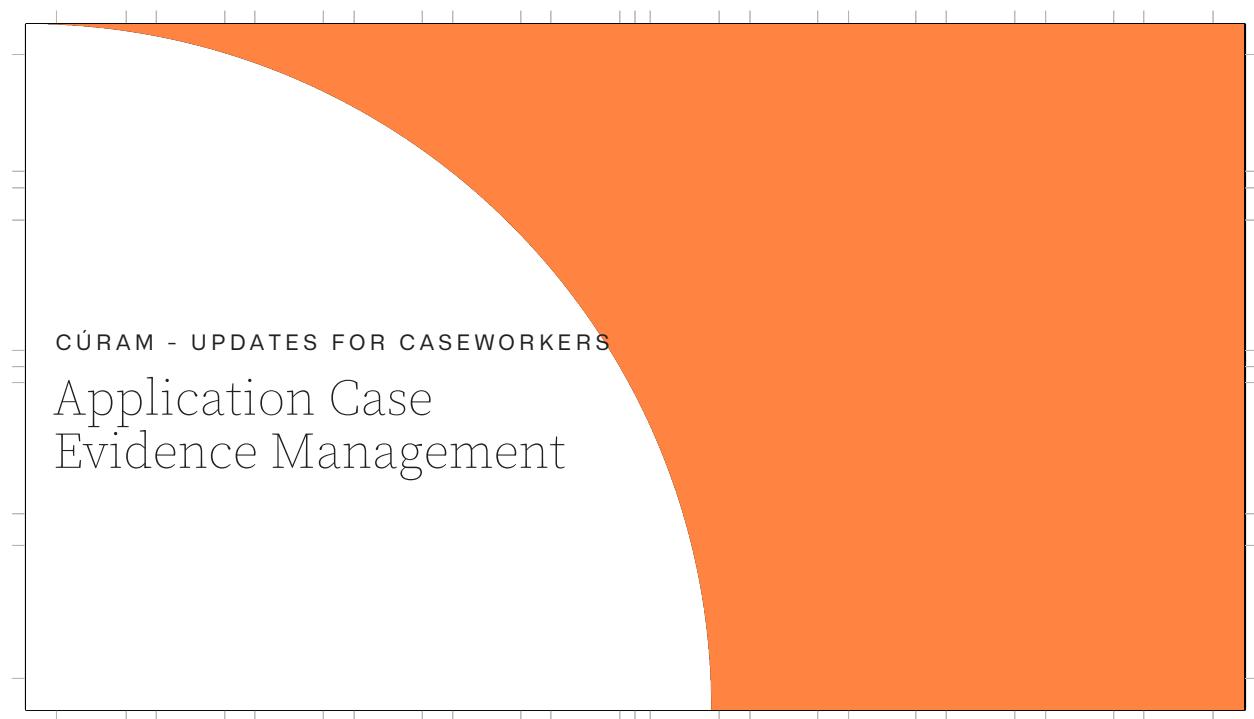
In 8.2.1.0

- When Enable Verification Sharing Post Activation is set to ‘True’, any new verification proof added after evidence activation will automatically be shared with all AES-linked evidence.
- If the property is set to ‘False’, the existing behaviour applies, i.e., verification proof added post-activation will not be shared.

Before 8.2.1.0, verification proof added before activation will be shared. Any verification proofs added post-activation will not be shared.

In 8.2.1.0, when Enable Verification Sharing Post Activation is set to ‘True’, any new verification proof added after evidence activation will automatically be shared with all AES-linked evidence.

If the property is set to ‘False’, the existing behaviour applies, i.e., verification proof added post-activation will not be shared.



CÚRAM - UPDATES FOR CASEWORKERS
Application Case
Evidence Management

Application Case Evidence Management	
Objective	Before 8.2.1.0
Introduce enhanced evidence management for application cases, so that caseworkers can control what evidence to activate, and in turn when evidence is shared to related ongoing cases for redeterminations.	Application cases support automatic activation of evidence only, with no option for caseworkers to manually apply changes.
Business Value	In 8.2.1.0
Supports organizations managing complex application cases with multiple programs and program level authorization enabled, ensuring better control and accuracy in eligibility determinations.	Application cases support multiple options for evidence activation, allowing caseworkers to manually apply changes if required by the organization.
Organizations can configure the application case so that caseworkers can control when evidence is applied on the application case and shared to the related ongoing cases.	
Gives caseworkers the ability to check the impact of the evidence changes they are making on eligibility, before applying those changes on the application case.	
<small>© 2025 Merative</small>	

This enhancement updates the application case platform feature, adding new configuration options for the application case that customers can leverage to enable the full evidence lifecycle on an application case.

Currently on application cases, all evidence updates are automatically applied, and there's no option for the caseworker to add in-edit evidence and apply changes. This works well for application processes where all evidence gathered via the application script can be automatically activated, the application case closes immediately on authorization, and all of the evidence, ongoing eligibility and case processing transfers to the integrated case.

However for multi-program application cases using program-level authorization there are scenarios where the application case and integrated case will be open in parallel, for example if one program has been authorized and one remains pending on the application case. In these scenarios, when the caseworker saves an update to an evidence record on the application case, that evidence will be automatically applied, shared to the integrated case and result in redetermination of eligibility on the related integrated case. Where the change involves multiple evidence types, every individual change on the application case will trigger sharing to the integrated case,

and result in unnecessary redeterminations based on incomplete evidence.

The new application case configurations provide enhanced support for differing application processing requirements, in particular complex application cases with multiple programs and program level authorization

Caseworkers can review the impact of evidence changes on eligibility, and manually apply evidence changes. Caseworkers can apply multiple evidence changes together, allowing evidence updates to be shared as a group from the application case to the related ongoing cases, helping to minimize unnecessary eligibility redeterminations and other downstream processing.

Application Case Evidence Management

Overview

- In the 8.2.1.0 release, administrators have the option to specify the **Evidence Activation Strategy** for an application case type:
 - **Auto apply evidence:** The system automatically applies all evidence updates on the application case. This is the default option and retains the functionality that was available before 8.2.1.0 where all evidence changes are automatically applied.
 - **Manually apply evidence:** Caseworkers manually apply all evidence changes on the application case.
 - **Conditionally auto apply evidence:** The system automatically applies evidence that is added/updated via the application script or advanced evidence sharing (AES), when there are no outstanding mandatory verifications. Caseworkers apply all manual evidence updates on the application case.
- Enable caseworkers to **manage in edit evidence** and **manually apply evidence changes** to application cases.
- Caseworkers can **apply multiple evidence changes together**, allowing evidence updates to be shared as a group with related ongoing cases, helping to minimize unnecessary eligibility redeterminations and other downstream processing.
- Enable caseworkers to **check eligibility based on active** evidence only, OR on **active and In edit** evidence.
- **Provide hook points** to enable customers to further customise evidence management on the application case, for example trigger a workflow/task if there's in-edit evidence on the application case following intake processing.

© 2025 Merative

Configuration options are provided to allow administrators to choose the evidence activation strategy for an application case type, based on their organization's requirements.

Note: The default configuration, Auto Apply Evidence, retains the current functionality where all evidence changes are automatically applied, so existing application cases are not impacted.

The new configuration options enable full evidence management on the application case for caseworkers.

Hookpoints are also provided to enable further customization of evidence management on the application case.

Administration Application: Evidence Activation Strategy New Configuration Option

The screenshot shows the 'New Application Case' configuration screen. At the top, there are fields for 'Issues Owner' and 'Resolution Home Page'. Below these, under 'Evidence Management', is a dropdown menu titled 'Evidence Activation Strategy'. This dropdown is highlighted with a red box and contains four options: 'Auto Apply Evidence' (selected), 'Auto Apply Evidence', 'Conditionally Auto Apply Evidence', and 'Manually Apply Evidence'. At the bottom of the screen are buttons for 'Cancel' and 'Save'.

The application case configuration screens, available from the Universal Access section of the Administration Application, have been updated to include a new configuration option: **Evidence Activation Strategy**.

Administrators can select the evidence activation strategy when configuring a new application case or editing the configuration of an existing application case.

Caseworker Application: Auto Apply Evidence Activation Strategy

All evidence changes are automatically applied. Retains the functionality available before 8.2.1.0.

When the evidence activation strategy is set to **Auto Apply Evidence**, all evidence added, updated and deleted on the application case is automatically applied as follows.

Evidence added automatically:

- Evidence mapped from the application script is automatically activated.
- Evidence shared from a related case is automatically applied.

Evidence updated by a caseworker:

- Newly added or updated evidence is automatically activated.
- Deleted active evidence is automatically applied.

Summary:

- Retains the automatic evidence processing on the application case that was available before 8.2.1.0.
- Supports straight through processing of applications, from intake through to benefit delivery.

Caseworker Application: Auto Apply Evidence Activation Strategy

All evidence changes are automatically applied. Retains the functionality available before 8.2.1.0.

The screenshot shows the Caseworker Application interface for a Job Seekers Application (Case ID 282). The application belongs to Ava Avery, 35 years old. The Evidence tab is selected. The page displays two pieces of evidence:

Type	Client	Description	Period	Latest Activity	Actions
Addresses	Ava Avery	Private address is 1, Ave A, Midway, Utah, 55522	11/3/2025 -	Created by SUPER USER on 11/4/2025	Edit... ...
Employment Status	Ava Avery	Is Employed full time.	-	Created by SYSTEM on 11/4/2025	Edit... ...

When **Auto Apply Evidence** is configured:

- The application case displays a **single evidence list** where the caseworker worker can add, edit, or delete records.
- The evidence dashboard is displayed, if configured, for the application case. **Note:** for consistency with other case types, the positioning of the Dashboard navigation link has been updated so it appears above the evidence list.

Caseworker Application: Conditionally Auto Apply Evidence Activation Strategy

Evidence changes are conditionally auto applied. New evidence management option available in 8.2.1.0.

When the evidence activation strategy is set to **Conditionally Auto Apply Evidence**, evidence is processed as follows on the application case.

Evidence added automatically:

- Evidence mapped from the application script is automatically activated when no mandatory verifications are outstanding.
- Evidence shared from a related case is automatically activated when no mandatory verifications are outstanding.
- The organization can define additional conditions for automatic activation exclusions through customization.
- When the conditions for automatic activation are not met, the evidence is saved as In Edit.

Evidence updated by a caseworker:

- Newly added or updated evidence is saved as In Edit and must be manually applied to the application case when there are no outstanding mandatory verifications
- Deleted active evidence is saved as Pending Deletion and must be manually applied to remove it.

Summary:

- Provides caseworkers with greater control over evidence activation and deletion, particularly when an ongoing case is open and updates to application case evidence may trigger automatic redetermination or other downstream processing.
- Supports multi-program application cases where program-level authorization is configured, and straight-through processing is required when evidence is completed and verified via the application script.

Caseworker Application: Conditionally Auto Apply Evidence Activation Strategy

Evidence changes are conditionally auto applied. New evidence management option available in 8.2.1.0.

The top screenshot shows the "In Edit Evidence" page. It displays a table with two rows of evidence items. The first row is for an "Address" entry for "Ava Avery" with the description "Private address is 1, Ave A, Midway, Utah, 55522". The second row is for an "Employment Status" entry for "Ava Avery" with the description "Is Employed full time.". The bottom screenshot shows the "Evidence Dashboard". It displays a grid of evidence categories: Household (Addresses, Current Asset, Asset Ownership), Earnings (Employment Status), and Medical Expense (Medical Assessment). Each category has a checkbox and a plus sign next to it.

When **Conditionally Auto Apply Evidence** is configured:

- The application case displays **separate Active and In Edit evidence list pages**, where the worker can add, edit, or delete records and apply changes.
- In Edit indicators** are displayed on the application case context panel and on the evidence dashboard (if configured).

Caseworker Application: Manually Apply Evidence Activation Strategy

All evidence changes are manually applied. New evidence management option available in 8.2.1.0.

When the strategy is set to **Manually Apply Evidence**, evidence is processed as follows on the application case.

Evidence added automatically:

- Evidence mapped from an application script is saved as In Edit.
- Evidence shared from a related case when a case member is added to the application case is saved as In Edit.
- Evidence shared when activated on a related case is saved as Active.

Evidence updated by a caseworker:

- Newly added or updated evidence is saved as In Edit and must be manually applied to the application case when there are no outstanding mandatory verifications
- Deleted active evidence is saved as Pending Deletion and must be manually applied to remove it.

Summary:

- Provides workers with full control over evidence activation and deletion on the application case.
- Supports organizations where applications require manual review and updates before authorization, for example, when interviews are part of the application process.

Caseworker Application: Manually Apply Evidence Activation Strategy

All evidence changes are manually applied. New evidence management option available in 8.2.1.0.

The top screenshot shows the 'In Edit Evidence' page. It displays a table with two rows: one for 'Addresses' (Private address is 1, Ave A, Midway, Utah, 55522) and one for 'Employment Status' (Is Employed full time). The bottom screenshot shows the 'Evidence Dashboard'. It lists categories like Household, Earnings, Employment Status, Medical Expense, etc., each with a count of items to verify, issues, and evidence in edit.

When **Manually Apply Evidence** is configured:

- The application case displays **separate Active and In Edit evidence list pages**, where the worker can add, edit, or delete records and apply changes.
- In Edit indicators** are displayed on the application case context panel and on the evidence dashboard (if configured).

34

Technical Overview

Configuration options

- A new column 'evidenceActivationStrategy' has been added to the ApplicationCaseAdmin database table. This column is nullable so that existing customers aren't impacted.
- A new codetable 'ACEvActivationStrategy' contains the values for Auto Apply Evidence, Conditionally Auto Apply Evidence and Manually Apply Evidence. The default code is set to Auto Apply Evidence so that any new configurations always populate. The code interprets NULL as the default, Auto Apply Evidence, which ensures existing behaviour for customers.

Customisation options

A number of hook points have been added to enable customization of the new evidence management logic:

- ***Curam.commonintake.authorisation.impl.ACEvidenceActivationEvaluatorHook***
 - 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 behaviour can be implemented for both Intake and AES by providing an implementation for the ACEEvidenceActivationEvaluatorHook.shouldActivateOnIntake() and ACEEvidenceActivationEvaluatorHook.shouldValidateOnSharing() methods.
- ***Curam.commonintake.impl.ACConditionalEvidenceInEditHook***
 - 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.

CÚRAM - UPDATES FOR DEVELOPERS

OpenJDK Enablement for Batch and XML Servers in Cúram

© 2025 Merative

36

OpenJDK Concerns

Business Problem

Current State

Proprietary JDK Dependency

Reliance on proprietary IBM/Oracle JDK for keystore creation increases costs and limits flexibility in server communication.

Impact

Cost and Flexibility Concerns

Lack of OpenJDK adoption leads to higher ownership costs and reduced infrastructure optimization options.

Need for Secure Open Solutions

A flexible and cost-effective approach to secure communication is essential for future Cúram system improvements.



© 2025 Merative 37

The current state of communication between Cúram application servers and Batch/XML servers raises critical questions regarding the necessity of proprietary JDKs, specifically IBM or Oracle JDKs, for keystore creation. Is this dependency a requirement leading to increased costs and limited flexibility?

If OpenJDK and its keystore capabilities are not utilized, organizations face a higher cost of ownership due to reliance on proprietary software. This situation underscores the need for a more flexible and cost-effective approach to secure communication within Cúram systems.

What's Changed?

Documentation Updates and Results

Step	Description
Key Documentation Updates	Three major Cúram guides (mentioned in Notes below) were updated to include OpenJDK keystore creation instructions for secure communication.
Keystore Creation Process	Documentation details how to create JCEKS keystores using OpenJDK and syncing with IBM JCEKS via PKCS12 format.
Improved Security and Compatibility	Secure communication can now be done between Cúram and Batch/XML servers using a mix of IBM and OpenJDK environments.
Benefits of Update	Reduces reliance on proprietary JDKs, supporting open-source solutions for cost-effective, sustainable operations.

Significant updates have been made to three key Cúram guides—Server Development Guide, Curam_InstallDevEnv_82.pdf, and the Cúram Security Handbook—to support the adoption of OpenJDK for keystore creation. These updates provide clear instructions on how to use OpenJDK's keystore capabilities to facilitate secure communication between Cúram application servers and Batch/XML servers. The revised documentation explains the process of creating JCEKS keystores using OpenJDK and synchronizing them with IBM JCEKS keystores through a PKCS12 intermediary. As a result, organizations can now successfully run batch jobs and maintain secure communication using a mix of IBM and OpenJDK environments. This change reduces dependency on proprietary JDKs and aligns with open-source practices, offering a more sustainable and cost-effective solution.

Reference documents, search Keystore in:

- CuramSecurityHandbook_821.pdf
- CuramServerDeveloperGuide_821.pdf
- Curam_InstallDevEnv_821.pdf

How it Works?

Keystore Creation and Synchronization Process

Step	Description
Create Keystore	Use OpenJDK with 'createkeystore' Ant target to generate JCEKS keystore
Export Key	Use OpenJDK keytool to export secret key to PKCS12 keystore
Import Key	Use IBM keytool to import secret key from PKCS12 to IBM JCEKS keystore
Package	Pack both keystores into CryptoConfig.jar for deployment

The process of enabling OpenJDK support for keystore creation involves several key steps. First, the 'createkeystore' Ant target is used with OpenJDK to generate a JCEKS keystore. In environments where multiple JDKs are used, the secret key from the OpenJDK keystore must be copied into a temporary PKCS12 keystore. This intermediary format is compatible with both IBM and non-IBM JDKs. Next, the IBM keytool is used to import the secret key from the PKCS12 keystore into an IBM JCEKS keystore. Finally, both keystores are packed into the CryptoConfig.jar file, ensuring that secure communication can be maintained across mixed JDK environments. This documented and repeatable process allows organizations to leverage OpenJDK while ensuring compatibility and security across their infrastructure.

Transition to OpenJDK for Batch and XML	
AS-IS	TO-BE
IBM or Oracle JDK required for keystore creation	OpenJDK for keystore creation for Batch and XML Server
Proprietary dependency	Open-source, cost-effective
Manual sync unclear	Documented process for syncing keys

© 2025 Merative

40

The transition from proprietary IBM JDK to OpenJDK for keystore creation represents a significant shift in the Cúram infrastructure strategy. In the As-Is state, IBM JDK is required for keystore creation, leading to proprietary dependency and unclear manual synchronization processes. The To-Be state introduces OpenJDK as the preferred solution for keystore creation specifically for Batch and XML servers. This change promotes the use of open-source tools, reduces costs, and provides a documented process for syncing keys across different JDK environments. The updated documentation ensures that organizations can confidently adopt OpenJDK without compromising security or compatibility, paving the way for a more flexible and sustainable infrastructure.

Benefits of OpenJDK Adoption

Cost Reduction
OpenJDK adoption lowers costs by removing the need for proprietary IBM JDK licenses.

Enhanced Flexibility
OpenJDK enables secure communication across mixed JDK environments without compatibility issues.

Improved Security Management
Documented keystore synchronization process simplifies infrastructure management and maintains security.

Future-Ready IT Strategy
Adopting OpenJDK aligns with open standards and vendor neutrality for evolving technology landscapes.

© 2025 Merative 41

Adopting OpenJDK for keystore creation in Cúram environments offers several key benefits. First and foremost, it lowers the cost of ownership by eliminating the need for proprietary IBM JDKs. This shift enables organizations to leverage open-source tools, which are more accessible and cost-effective. Additionally, OpenJDK provides greater flexibility, allowing for secure communication across mixed JDK environments without introducing compatibility issues. The documented process for keystore synchronization ensures that security is maintained while simplifying infrastructure management. Finally, this approach aligns with future-ready IT strategies that prioritize open standards and vendor neutrality, making it easier to adapt to evolving technological landscapes.

Reference documents search Keystore

CuramSecurityHandbook_821.pdf

CuramServerDeveloperGuide_821.pdf

Curam_InstallDevEnv_821.pdf

8.2.1.0 Prerequisites updates:

J8=> https://curam-spm-devops.github.io/wh-support-docs/spm/prerequisites/821_java8

Modern Java =>https://curam-spm-devops.github.io/wh-support-docs/spm/prerequisites/821_modernjava

CÚRAM - UPDATES FOR CASEWORKERS

New validation on
Application Level
Authorization on
Application Cases

New validation on Application Level Authorization on Application Cases

Before 8.2.1.0

The system does not prevent application level authorization if there are outstanding mandatory verifications on an application case.

In 8.2.1.0

The system checks for any outstanding mandatory verifications before completing application level authorization, adopting the approach that already exists for program level authorization.

A new system wide property is provided to give organizations flexibility to enable or disable the check based on their business requirements.

Note: The check is disabled by default, so existing application cases with application level authorization are not impacted.

Technical Overview

New System Property

Name: Enable mandatory verification validation for application level authorization

Description: 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.

Category: "Application - Authorization Settings"

ID:

curam.commonintake.application.authorization.mandatory.validation.enabled

Values: "true" (enabled) / "false" (disabled)

- **Default Value:** "false"
- **Upgrade Impact:** Upgraded systems will have this property defaulted to a value of 'false'

Behaviour:

- **Value = true**
 - the system will validate that there are no outstanding mandatory verifications on authorization of the application.
- **Value = false**
 - the system will not check if there are outstanding mandatory verifications on authorization of the application (as-is behaviour).

CÚRAM - UPDATES FOR DEVELOPERS

Hook point to specify
additional Evidence
validations on Application
and Program level
Authorizations

Technical Overview

This release introduces a new authorization validation hook point for application cases to allow customers to implement custom checks before authorizing an application or program. Using this hookpoint, organizations can enforce additional business rules such as requiring a minimum set of evidence to be captured before authorization is permitted.

- **Curam.commonintake.authorisation.impl.CIAuthorisationValidationHook**

- 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 behaviour can be implemented for both program authorization and application authorization by providing an implementation for the `curam.commonintake.authorisation.impl.CIAuthorisationValidationHook.validateProgramAuthorisation()` and `curam.commonintake.authorisation.impl.CIAuthorisationValidationHook.validateApplicationAuthorisation()` methods.

CÚRAM- UPDATES FOR DEVELOPERS

Pre-customization hook
points for Product Delivery
lifecycle

Pre-customization hook points for Product Delivery lifecycle

Business Problem

The current Product Delivery lifecycle only supports post-delivery customization hooks. This limitation restricts customers from applying custom logic before critical delivery actions occur, reducing flexibility and potentially leading to non-compliant customizations. A mechanism is needed to allow pre-delivery customization to ensure compliance and tailored behaviour throughout the lifecycle.

Solution

The enhancement introduces pre-delivery customization hook points within the Product Delivery lifecycle. By supporting both pre- and post-delivery stages, customers can implement custom logic at critical points, ensuring compliance and reducing the risk of non-compliant customizations. This approach also minimizes the impact of product upgrades on existing customizations, providing a more robust and flexible customization framework.

Before 8.2.1.0

Prior to this release, the Product Delivery lifecycle offered only post-delivery customization hooks. Customers could apply custom logic after delivery actions were completed, but there was no capability to intervene before delivery actions occurred.

In 8.2.1.0

The following pre-delivery customization hook points are now provided:

- preProductDeliverySubmittedForApproval
- preProductDeliveryApproved
- preProductDeliveryActivation
- preProductDeliveryReactivation
- preProductDeliveryRejection
- preProductDeliveryClosure

These hooks allow customers to execute custom logic before key delivery actions, enabling greater flexibility and compliance across the entire Product Delivery lifecycle.

© 2025 Merative 48

Previously, the Product Delivery lifecycle only supported post-delivery customization hooks. This limitation restricted customers from applying custom logic before critical delivery actions occur, reducing flexibility and potentially leading to non-compliant customizations. A mechanism is needed to allow pre-delivery customization to ensure compliance and tailored behaviour throughout the lifecycle.

This enhancement introduces pre-delivery customization hook points within the Product Delivery lifecycle. By supporting both pre- and post-delivery stages, customers can implement custom logic at critical points, ensuring compliance and reducing the risk of non-compliant customizations. This approach also minimizes the impact of product upgrades on existing customizations, providing a more robust and flexible customization framework.

Before 8.2.1.0, the Product Delivery lifecycle offered only post-delivery customization hooks. Customers could apply custom logic after delivery actions were completed, but there was no capability to intervene before delivery actions occurred.

In 8.2.1.0, The following pre-delivery customization hook points are now provided

- preProductDeliverySubmittedForApproval
- preProductDeliveryApproved
- preProductDeliveryActivation
- preProductDeliveryReactivation
- preProductDeliveryRejection
- preProductDeliveryClosure

These hooks allow customers to execute custom logic before key delivery actions, enabling greater flexibility and compliance across the entire Product Delivery lifecycle.

CÚRAM - UPDATES FOR DEVELOPERS

Display money fields
based on user's locale

Display money fields based on user's locale

Business Problem

Previously, there was no placeholder for the CURAM_MONEY data type when formatting messages. As a result, tasks that included monetary amounts in the subject field displayed incorrectly, missing currency symbols and containing unwanted commas in the amount. This created usability issues and reduced clarity for users.

Solution

The message infrastructure has been enhanced to support a new placeholder for monetary values. This allows messages to dynamically format money fields based on the user's locale, ensuring accurate and readable presentation of currency in multilingual environments.

Before 8.2.1.0

Message files did not support a placeholder for the CURAM_MONEY data type. Monetary values were displayed without proper currency symbols and formatting, leading to inconsistencies and poor user experience.

In 8.2.1.0

Message files now support locale-based formatting for monetary amounts. To enable this capability, system administrators can configure how monetary amounts are formatted based on locale preferences. When this property is enabled, the system automatically presents monetary values in the format appropriate for the selected locale, enhancing usability and readability for users working in multilingual environments.

Previously, Cúram message formatting did not include a placeholder for the CURAM_MONEY data type. This meant that whenever a task subject or message included monetary values, the amounts were displayed incorrectly - . This created usability issues and reduced clarity for users.

To address this issue, message infrastructure has been enhanced to support a new placeholder for monetary values. This allows messages to dynamically format money fields based on the user's locale, ensuring accurate and readable presentation of currency in multilingual environments.

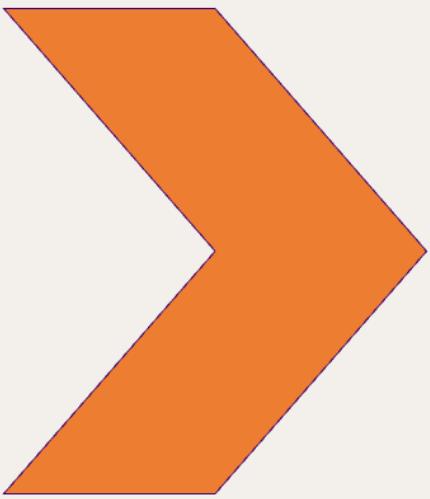
Prior to 8.2.1.0, message files did not support a placeholder for the CURAM_MONEY data type. Monetary values were displayed without proper currency symbols and formatting, leading to inconsistencies and a poor user experience.

In 8.2.1.0, Message files now support locale-based formatting for monetary amounts. To enable this capability, system administrators can configure how monetary amounts are formatted based on locale preferences. When this property is enabled, the system automatically presents monetary values in the format appropriate for the

selected locale, enhancing usability and readability for users working in multilingual environments.



Thank You



© 2025 Merative