Constellation Modernization Assistant Component
Version 1.0.8

Introduction

This assistant will analyze your application and identify some rules that do not follow best practices and will need to be updated to modernize your application and support Constellation UI.

The tool provides 2 modes of operation:

- **An analysis mode** will report a list of issues that will need to be addressed to modernize your application. An Excel file can be downloaded with the list of issues for each rule.
- **A migration mode** will create a new application based on Constellation UI and a new ruleset and access group – the new application will be built-on your current application – Migrated rules will either be saved in a branch while maintaining the original ruleset or in the newly created ruleset. During the migration, all the sections will be converted to views and the properties will be marked as relevant records.

Note: before running the migration, **your current application rule must be locked,** and **all the rulesets and ruleset versions of your applications must be locked**. An error message will be reported and prevent the migration if these configuration changes are not applied. This is a change from version 1.0.7 and earlier where the application rule must be unlocked.

Depending on the user action, the rules will either be created into a **single** ruleset that will be named from the first 10 characters of your application name or will be stored in a new branch that will maintain the same ruleset name. When selecting to save the rules in a branch, the new application will be configured with branch development turned ON.

This component is supported on Pega Platform 8.7 or higher, but **the migration functionality is only supported on Pega 23.1 or higher**.

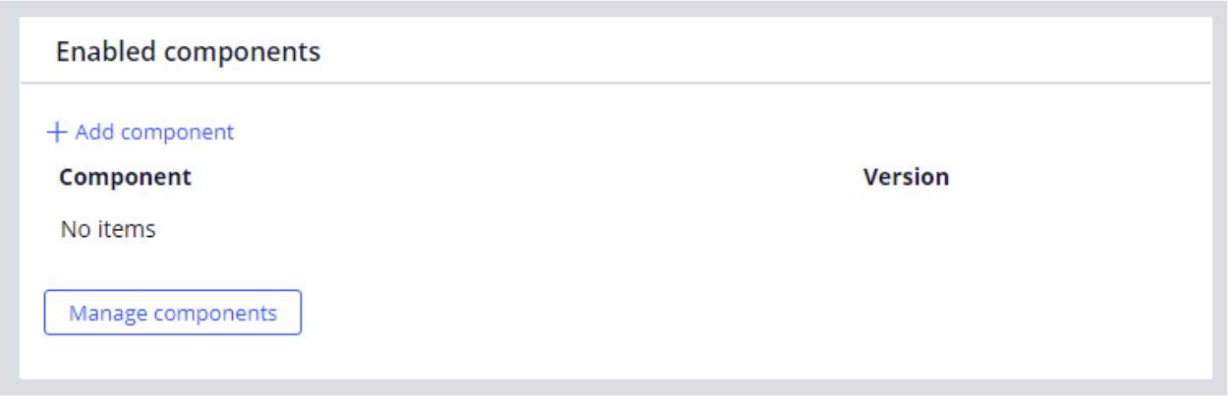
The complete documentation for the Constellation Modernization assistant component is available here: https://docs.pega.com/bundle/platform/page/platform/user-experience/application-modernization-assistant-component.html

If you have any questions using the tool, please send an email to constellation@pega.com

Installation

1/ **Open your Application Rule** - Navigate to your Application Rule through the Developer Studio in your Pega environment.

2/ **Manage your components**
Locate and click on the "Manage components" button in your Application Rule



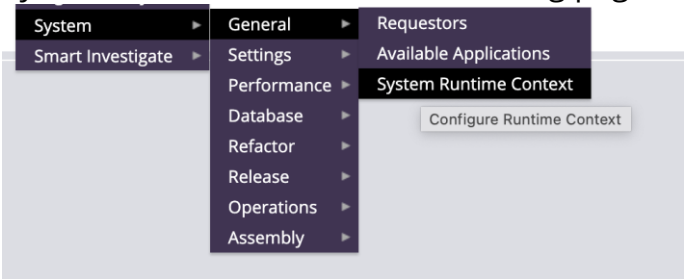
3/ Install the new component
Click on "Install new" to add a new component to your application.

4/ Select the component
Browse and select your downloaded copy of the Pega Constellation Modernization Assistant

5/ Complete the installation
Follow the on-screen instructions to complete the installation process. Once done, enable the component in your App by checking the "Enabled" box.

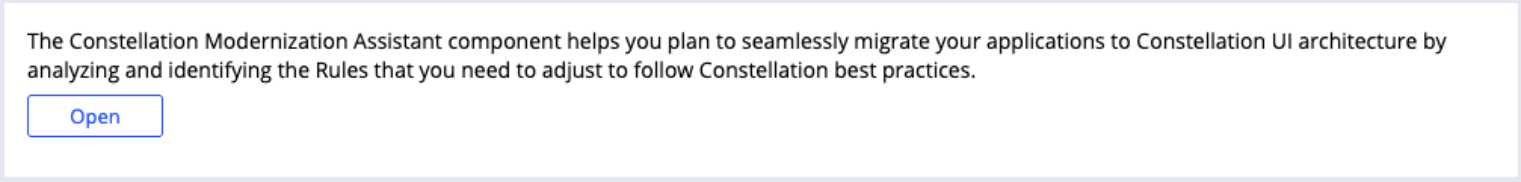
6/ Save your application
Ensure to save your application after the component has been installed and enabled. This action will make the component readily available for use in your application.

7/ Add your application to the System Runtime Context
The analysis will run as a queue processor task and will take several minutes to complete. Make sure that you have the queue processor correctly configured and that your application is present in the System Runtime Context. This landing page is available in Dev Studio

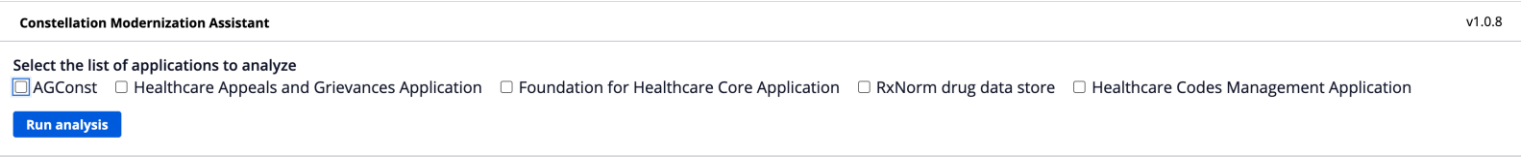


Getting Started

On the home page of designer studio, you should see a button to launch the Constellation Modernization Assistant.

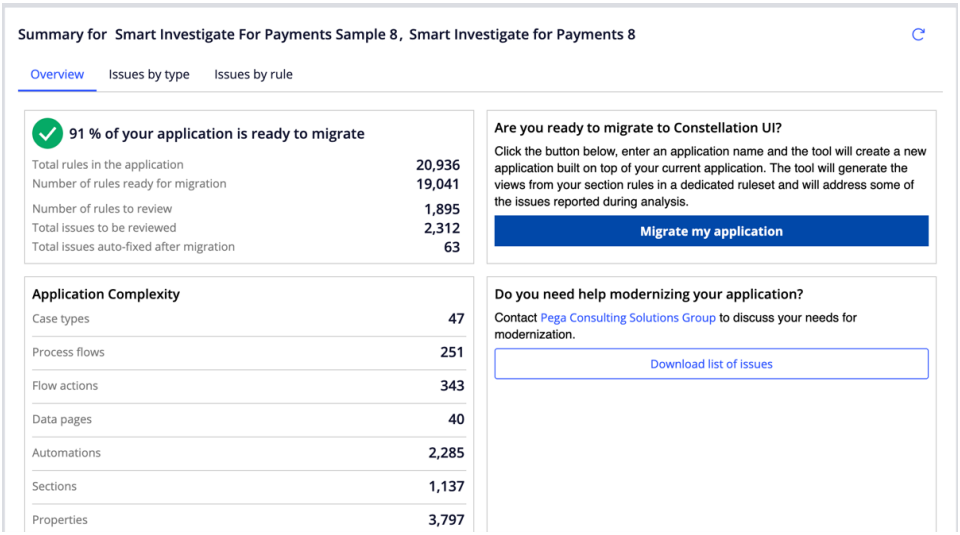


Select the application stack that you want to analyze. You can analyze one or multiple application stack.

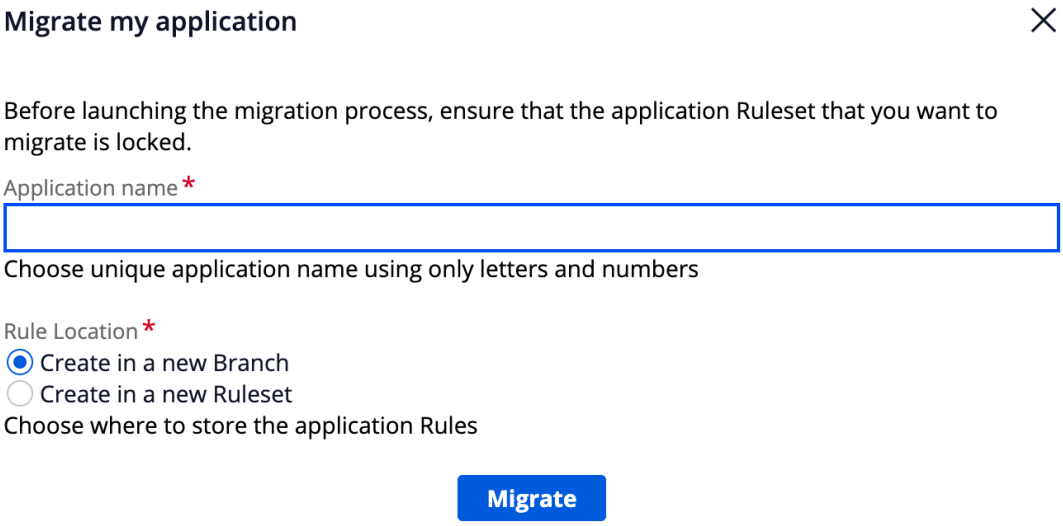


After selecting a list of application to analyze, the tool will execute the analysis in background using a queue processor – You can close the page or re-open the page at any time – the analysis will be available to all the users of the application and the results are stored as instances of the Pega-Constellation-Migration class and subclasses.

The summary will list some high-level information about your application, the number of rules that will be addressed for modernization, as well as a detailed list of the issues by type and rule. You can download an excel doc that will contain the list of all the rules (by pzInsKey), the issue ID and the number of occurrences of the issue in the given rule.



If you are running on Pega 23.1 or higher, you can migrate your application to Constellation UI. The tool will ask for a new application name that will be used to create a new application rule, a new access group and a new ruleset. It will also ask if the migrated rules should be saved in a branch or in the newly created ruleset. The recommendation is to use the branch mode.



During migration, the following changes will be performed:

- All the sections in the selected application lists will be converted to views and saved in the new ruleset.
- All the flow actions in the selected application lists will be upgraded to include the view name and saved in the new ruleset.
- All the pyDefault case type rules will be preserved in the new ruleset and a Rule-ClassMeta created for these classes if not present.
- All the properties not marked as relevant records will be automatically added if they are not at @baseclass.
- Access group will be automatically added to your user, and you will be able to switch to the new application once the migration is completed.

Important notes: Before running the migration tool, make sure that the following conditions are present:

1. **Your current application must be locked** so that the new ruleset can be temporarily added to your current application while the migration is running.
2. Ensure that **all the ruleset versions of your application and in your built-on applications are locked**.
3. Ensure that you are **running with sysAdm4 privileges or higher** – the tool will create new rules in different classes, and the user needs to have the right privileges

Limitations

1. The migration tool is still an early version and will not convert 100% of the section model to a view – Some components like table and Repeating Dynamic Layouts (RDL) are not yet supported by the conversion of sections to views – We plan to improve this functionality in the upcoming releases of the tool.

2. The migration does not run as a queue processor and will run as the current user and will lock the user from doing any other activities while the migration is happening – While the migration is happening in the context of the new application, the new ruleset needs to be temporary added to your current application.

3. The v1.0.x of the tool detects more than 50 modernization issues. The list of issues detected by the tool is not an exhaustive list, and more issues will be detected in the upcoming versions of the tool.

4. The tool does not help migrate your application to the Common Data Model (CDM). If you modernize your application to run on **Sales Automation on Constellation or Customer Service on Constellation**, you will need to manually convert your application to the new data objects supported by CDM.

5. After migrating your application and running the Constellation Modernization Assistant, you may observe increased occurrences of *OBJCLASS_MISSING_DEFAULT_LIST_PAGE* and *OBJCLASS_MISSING_DEFAULT_LOOKUP_PAGE issues.* We plan to resolve this issue with the future release of the tool.

FAQs

1. The Constellation UX tab returns the "**Error in UI Authoring: Mismatch between View metadata and Template definition regions.**" error message

This error is presented because you configure a traditional case type to run in Constellation using the following procedure: https://docs.pega.com/bundle/platform/page/platform/user-experience/adding-traditional-case-constellation.html

The error in UI authoring occurs because the *pyDetails* view, saved in a custom class and ruleset, incorrectly uses the *pyDetailsCoexistenceTemplate*, which is only intended for traditional application blending within Constellation. To address the issue, delete the custom *pyDetails* view for the classes specific to case type and use the one at work - class which is in the OOTB ruleset.

History

Version	Release Date	Changes
1.0	1/10/2024	Initial version of the tool
1.0.3	2/20/2024	Enhancements: • Improve the detection around custom controls. Added 2 new modernization issues called HTMLPROPERTY_CUSTOM_CONTROL and HTMLPROPERTY_CUSTOM_MODERNIZE to identify cases where a Constellation DX component should be created or not Usability improvements: • Add 2 new 'severity' and 'complexity' columns for each issue to help identify the issues that should be addressed first (high severity) and the ones that require more time to address (high complexity). • Improvements in the table for issues by type: better filtering of the modernization issues, switch to optimized table for grouping and filtering, add new columns like rule class, ruleset name and version and show the rule ID instead of the Inskey. Other issues: • Fix false-positive detection of CASE_TYPE_MISSING_CREATE_STAGE – Added check for initialStage only for first stage for case types • Fix false-positive detection of FLOW_ACTION_RULE_NAME_MISMATCH - Checks for view name only if it is not empty inside flow actions • Fix issue where SECTION_INVALID_PAGELIST_PROPERTY_REFERENCE was not flagged for sub sections
1.0.4	3/6/2024	Enhancements: • Improved detection around data pages: o DATAPAGE_EDITABLE: Data pages that are editable are not supported in Constellation UI, use Savable data page instead. o DATAPAGE_NODE_LEVEL: Node level data pages are not supported in Constellation UI o DATAPAGE_NOTAVAILABLE: Single type of Datapage should be marked as API and available o DATAPAGE_QUERYABLE: Data page should be mark as queryable, API and available and reload once per interaction • Improved detection around case and data types: o OBJCLASS_MISSING_CLASSMETA - No Rule-ClassMetadata found o OBJCLASS_MISSING_DEFAULT_LIST_PAGE - Default list data page is missing in Rule-ClassMetadata

		○ OBJCLASS_MISSING_DEFAULT_LOOKUP_PAGE - Default lookup data page is missing in Rule-ClassMetadata • Improved detection for picklist properties ○ PROPERTY_INVALID_PICKLISTSRC - Picklist properties support only Prompt List and Data Pages as data sources Other issues: - Address issue with removal of duplicate issues,
1.0.5	3/15/2024	Enhancements: • 6 new detections added: Improved detection for charts in section, picklist properties in section and template properties in section • Improved UI to filter by rule • Improved xls doc with rulename and classname Other issues: • Fix issue with xls file being empty • Expose rulename, classname and ruleset in the xls document • Increase size of some of the exposed db columns
1.0.6	12/27/2024	Enhancements: • Added support for reporting issues that could be auto fixed during the migration. Three types of issues are currently fixed in this version: FLOW_ACTION_RULE_NAME_MISMATCH, PROPERTY_NOT_RELEVANT_RECORD and REPORTDEF_CONVERT_INSIGHT
1.0.7	3/7/2024	Enhancements: • Fixed issue where flow actions did not contain the view name after upgrade on 24.2 - After this change, ALL the flow actions are resaved in the migration ruleset and the view reference is updated • Resave all case types pyDefault rule into the migration ruleset and generate the Rule-ClassMeta if not present • Other improvements around identifying affected rules by different applications
1.0.8	5/23/2024	Enhancements: • Introduced the ability to store all modified rules in a dedicated branch during migration. This enables a new "branch development mode" within the migrated application, facilitating easier parallel development and review of changes. • Enforced the requirement to lock application rulesets prior to initiating the migration process. This ensures data integrity and prevents unintended modifications during migration. Bug Fixes:

		• Resolved an issue that prevented installation on Oracle databases by limiting the size of the *pyMessage* column to 4000 characters. • Fixed a bug where newly created rulesets were sometimes incorrectly added to the traditional application instead of the new one. • Addressed a *NullPointerException* that was occasionally observed in the logs for the *myStepPage* within the *pyc11BuildNewApp* activity.