Know Your Customer

Implementation Guide

7.1.1

Pegasystems Inc.
One Rogers Street
Cambridge, MA 02142-1209

Phone: (617) 374-9600
Fax: (617) 374-9620

www.pega.com

Document: Know Your Customer Implementation Guide
Software Version: 7.1.1
Updated: January 2014

Contents

About This Document

This document describes how to customize, deploy and extend the Know Your Customer Framework (KYC) for your initial development and production use.

Intended Audience

- **Business Managers** — responsible for evaluating the solution and possess a general, non-technical understanding of its features and capabilities
- **Project Managers / Business Analysts** — responsible for implementing a solution that can be applied to specific business requirements, ensuring compliance and continuous improvement across organizations
- **System Architects / Application Developers** — responsible for building, maintaining, modifying, and extending the solution
- **System and Database Administrators** — responsible for the installation, security, and ongoing operational functions of the framework such as access, tuning, and troubleshooting; presumed to be experienced with system operations.

Guide Organization

This guide contains the following chapters and appendix.

Chapter / Appendix	Description
Chapter 1: KYC Overview	Gives a business overview of the framework and describes the new features and enhancements in this release.
Chapter 2: What is Already Set Up	Provides information about what is already set up and configured when KYC is installed.
Chapter 3: KYC Baseline Performance Measures	Provides information about how to achieve the best system performance when configuring the KYC capabilities described in this guide.
Chapter 4: Configuring and Displaying KYC Types and Items	Describes and walks you through the process of adding and configuring a new KYC type rule and populating KYC items with data.
Chapter 5: Advanced Configuration Options	Describes the advanced options related to configuring KYC types and items.
Chapter 6: Integrating With a Content Management System	Provides instructions for integrating KYC with the Alfresco content server that supports CMIS.
Appendix A: Known Issues and Limitations	Lists known issues and limitations of this KYC release.

Documentation Set

In addition to this document, the KYC documentation set includes:

- *Know Your Customer Installation Guide V7.1.1* — describes how to install KYC 7.1.1
- *Know Your Customer Upgrade Guide V7*.1.1 — describes how to upgrade to KYC 7.1.1
- *Know Your Customer Release Notes V7.1.1* — describes new features of the 7.1.1 release, platform support and known/resolved issues

These documents are available on the Pega Discovery Network (PDN), a section of the Pegasystems Support Network located at **pdn.pega.com**.

Chapter 1: KYC Overview

The Know Your Customer Framework (KYC) provides a unified platform for streamlining compliance during the on-boarding and maintenance processes. KYC enforces common best practices and regulations while dynamically supporting unique regulations, policies and procedures by geography, line of business and product. Institutions can easily automate the steps required to comply with multiple regulations that effect on-boarding and time to revenue, including Enhanced Due Diligence (EDD), Suitability, FATCA, MiFID and FINRA requirements.

Solution Benefits

Compliance

- A master customer profile delivers a 360-degree view of individuals and entities across accounts, geographies, lines of business and complex direct/indirect and parent/ child relationships.
- Dynamic rules support specialization by country, line of business and product for KYC, EDD, FATCA, MiFID, FINRA and other compliance requirements.
- A unified platform with automated SLAs and escalation prioritizes and routes due diligence activities among multiple users for timely and accurate evaluation.

Accelerate On-Boarding and Time to Revenue

- Intent-led processes guide users, automatically adjusting the steps based on customer, product, geography, regulatory requirements and risk rating.
- End-to-end workflow automation with easy, quick integration to enterprise and third-party systems eliminates manual processing and repetitive document requests.

Improve Risk Assessment

- Easily configured rules assign risk rating based on customer, product and geography.
- Automatically recalculated overall case and risk rating identifies and drives additional KYC requirements throughout the customer lifecycle.

Streamline Reporting

- An easily configured, real-time reporting dashboard combines with robust out-of-the-box reporting on key operational risk metrics, including timeliness, volumes and trending, to deliver exceptional visibility, transparency and control.
- A date and time-stamped history provides a complete audit trail of all of manual and automated steps taken during evaluation, including related documentation and all system, rule and workflow changes.

Rapidly Adapt to Changing Regulations

- Familiar office tools enable rapid configuration and modification of rules, workflows, user interfaces and risk variables to support regulatory changes. KYC is also easily extended to “Know Your Employee” and “Know Your Broker” requirements.
- Reusable and extensible case and folder structures maintain all KYC information for an entity’s KYC profile.

Chapter 2: What is Already Set Up

This chapter describes defaults and samples that are set up and ready for your use. It is expected that you will use these as a basis for extending KYC.

The topics are:

- System Administrator Account
- RuleSet Hierarchy
- Work Classes
- Data Classes
- Implementation Layer
- Work Object Prefixes
- Organizational Structure
- Work Groups and Workbaskets
- Operators, Access Groups and Portals
- Case Types
- Work Parties
- Sample Database Model
- Data Tables
- Properties

System Administrator Account

You can use the following operator IDs to access the framework as a system administrator. A full list of users installed with the framework can be found in the Operators, Access Groups and Portals topic of this chapter.

Use this administrator ID to access and work with the framework rules and processes:

Operator ID: KYCSysAdmin

Password: install

RuleSet Hierarchy

The framework is built upon a number of Application Rules. Application rules define an ordered set of Rule Sets and versions that together identify the parts of a framework layer. In addition, application rules relate an application's objectives, use cases, requirements, and actors to cases that are created as part of PRPC Direct Capture of Objectives capabilities.

You can view a list of the application rules and their RuleSets that form the framework by selecting DesignerStudio™ > Application > Structure > RuleSet Stack landing page option.

The **RuleSet Stack** tab displays the high level RuleSet stack for each application rule defined in the framework. Expand an application to list its RuleSets.

The framework application name and version is PegaKYC:07.1

To see the RuleSet stack for any of the supporting application layers, click **+** next to the application to expand and display the list.

This table lists the current and built-on applications and their associated rulesets.

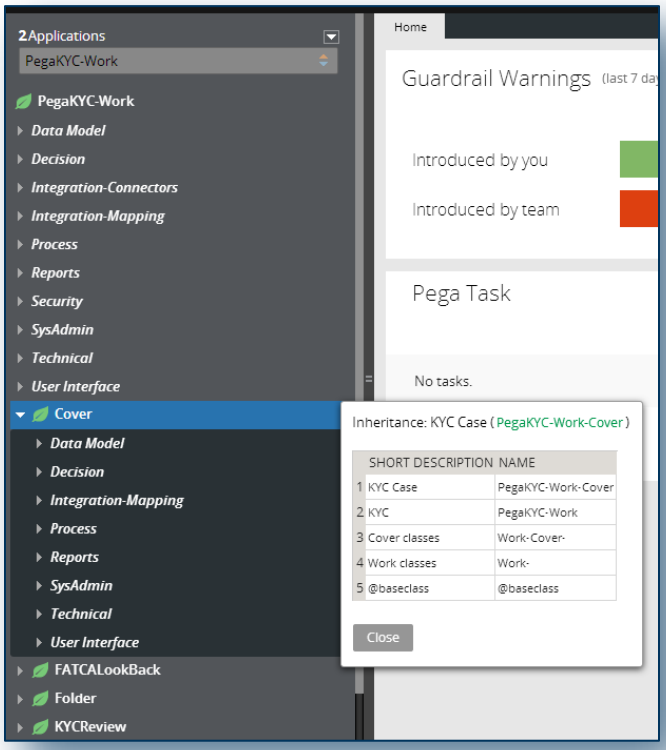
CURRENT APPLICATION	RULESET
PegaKYC:07.1	PegaKYC:07-01-05
	CMISPlus:01-02-01
	RelationshipViewer:01-02-01
	PegaFSUI:07-10-01
	PegaFWUI:07-10-03
	Pega-DecisionArchitect:07-10-08
	Pega-DecisionEngine:07-10-08
BUILT ON APPLICATION	
PegaRULES:07.10	Pega-ProcessCommander:07-10-99
	Pega-LP-ProcessAndRules:07-10-08
	Pega-LP-Integration:07-10-08
	Pega-LP-Reports:07-10-08
	Pega-LP-SystemSettings:07-10-08
	Pega-LP-UserInterface:07-10-08
	Pega-LP-OrgAndSecurity:07-10-08
	Pega-LP-DataModel:07-10-08
	Pega-LP-Application:07-10-08
	Pega-LP:07-10-08
	Pega-UpdateManager:07-10-08
	Pega-SecurityVA:07-10-08
	Pega-Feedback:07-10-08
	Pega-AutoTest:07-10-08

	Pega-AppDefinition:07-10-08
	Pega-ImportExport:07-10-08
	Pega-LocalizationTools:07-10-08
	Pega-RuleRefactoring:07-10-08
	Pega-ProcessArchitect:07-10-08
	Pega-Portlet:07-10-08
	Pega-Content:07-10-08
	Pega-IntegrationArchitect:07-10-08
	Pega-SystemArchitect:07-10-08
	Pega-Desktop:07-10-08
	Pega-EndUserUI:07-10-08
	Pega-Social:07-10-08
	Pega-EventProcessing:07-10-08
	Pega-Reporting:07-10-08
	Pega-UIDesign:07-10-08
	Pega-Gadgets:07-10-08
	Pega-UIEngine:07-10-08
	Pega-ProcessEngine:07-10-08
	Pega-SearchEngine:07-10-08
	Pega-IntegrationEngine:07-10-08
	Pega-RulesEngine:07-10-08
	Pega-Engine:07-10-08
	Pega-ProCom:07-10-08
	Pega-IntSvcs:07-10-08
	Pega-WB:07-10-08
	Pega-RULES:07-10-08

Work Classes

Work classes support the behavior and appearance of the work items created and processed in the framework. Each class maps to a table in the PegaRULES database and belong to the PegaKYC-Work class group which is designed to associate similar or related work- classes into one database table.

You can display the class inheritance by selecting the class from the explorer panel. Then right-click and select the **Inheritance** menu option.



Top level work class

PegaKYC-Work is the topmost work class in the framework. It directly inherits from Work-.

Classes and Work Types

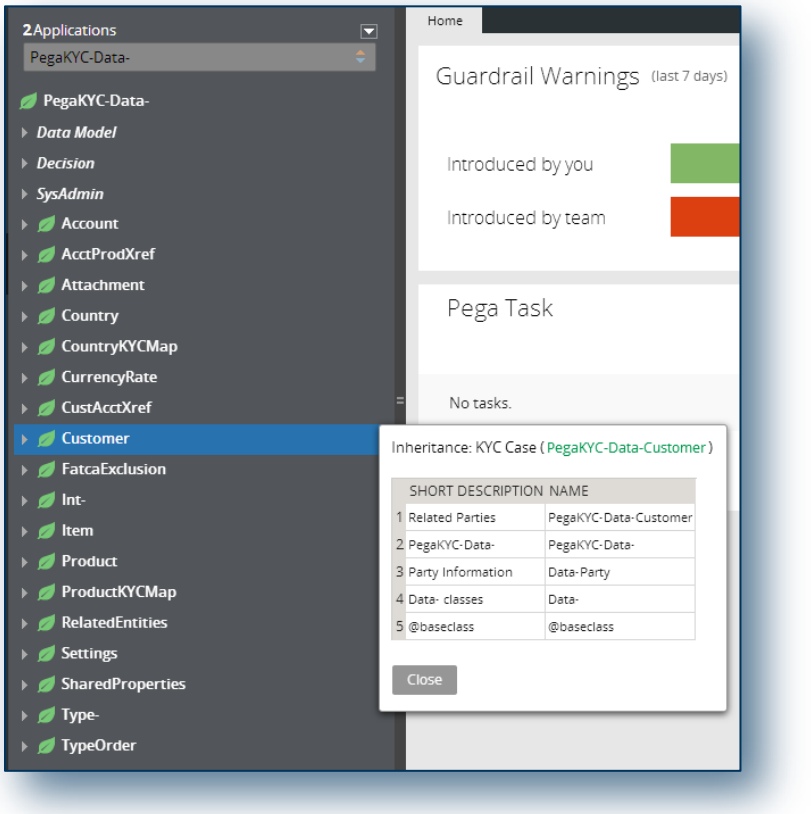
This table lists the work classes that support the KYC Case and Master work types. Data related to PegaKYC-Work-Cover is mapped to the KYC_WORK table in the PegaRULES database and belongs to PegaKYC-Work class group. Data related to PegaKYC-Work-Folder is also mapped to KYC_WORK in the PegaRULES database and belongs to the PegaKYC-Work class group.

Work Class	Case Type	Inherits from
PegaKYC-Work-NewBusiness	NewBusiness	PegaKYC-Work-Cover
PegaKYC-Work-FATCALookBack	FATCALookBack	PegaKYC-Work-Cover
PegaKYC-Work-KYCReview	KYCReview	PegaKYC-Work-Cover
PegaKYC-Work-Folder	Master	Work-Folder-

Data Classes

Multiple data classes support the capture and data storage for the processing of KYC cases.

You can display the data class structure by selecting the class from the explorer panel. Then right-click and select the Inheritance menu option.



This table lists the key data classes.

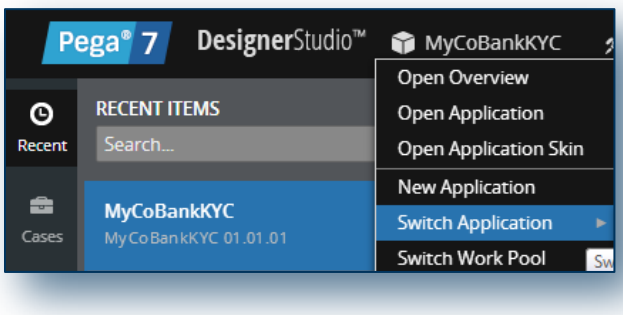
Data Class	Description	Directly Inherits from
PegaKYC-Data-Customer	Contains data pertaining to the Customer	PegaKYC-Data-
PegaKYC-Data-Account	Contains data pertaining to the Customer's Account(s)	PegaKYC-Data-
PegaKYC-Data-CustAcctXref	Contains cross reference data pertaining to each Account associated with a customer	PegaKYC-Data-
PegaKYC-Data-Product	Contains data pertaining to the Account's Product(s)	PegaKYC-Data-
PegaKYC-Data-Type-	Contains data and sub-classes pertaining to specific KYC requirements (FATCA, Standard Due Diligence, Documents, etc…)	PegaKYC-Data-
PegaKYC-Data-AcctProdXRef	Contains cross reference data pertaining to each Account associated with a product	PegaKYC-Data-

Implementation Layer

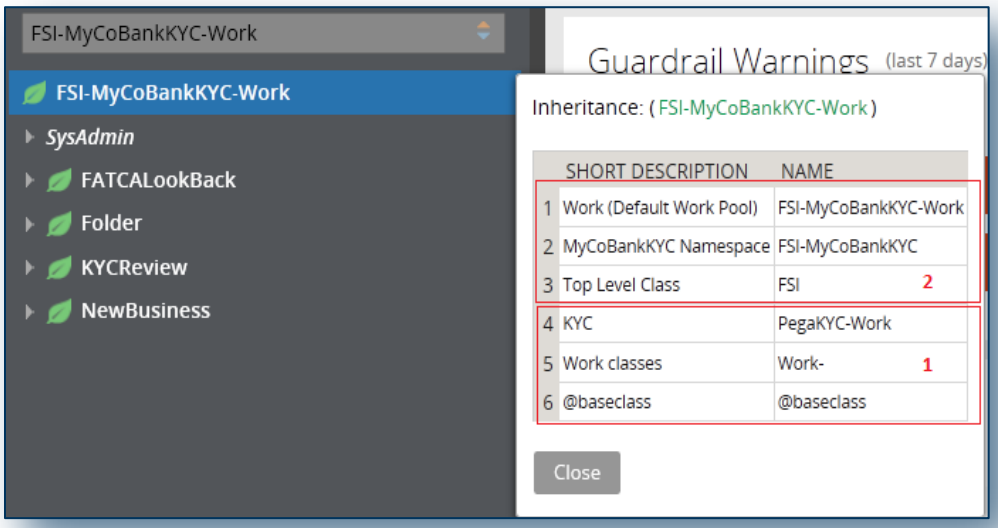
A sample implementation application layer is included with the KYC framework. This layer provides a working model of how specialized application layers can be extended from the KYC. This sample layer also contains pre-configured and re-usable KYC business rules built for the financial services industry.

The assets contained in this layer can be copied to a production implementation layer for re-use. The sample layer can then be removed from the application stack entirely.

While signed on as KYCSysAdmin, you can switch to the sample application by clicking the Application Name > Switch Application > Know Your Customer for Financial Services option from the portal header.



You can display the class inheritance by selecting the class from the explorer panel. Then right-click and select the **Inheritance** menu option.



Layer Descriptions

Layer 1:- PegaKYC Framework Layer. This layer holds all the out-of- the -box KYC Rules installed with the product.

Layer 2:- MyCoBankKYC Layer. This is sample implementation layer which is built on top of the PegaKYC Framework layer.

Important! The MyCoBankKYC Layer contains the **SampleFSDueDiligence** ruleset which contains all out-of-the-box KYC Type rules available in the system. To re-use these assets without having to extend from the sample layer, you can build your custom KYC application on top of the PegaKYC Framework layer and add the SampleFSDueDiligence ruleset to your custom application's ruleset.

Work Object Prefixes

Both the framework and implementation layers create folder work objects and cover work objects. Covers are created in the form of New Business, FATCALookback and KYC Review work objects. Each object type has a unique ID that is computed by combining a system-assigned number and a prefix defined in the **Details Tab** of the Application Rule.

This table lists the prefixes for the work objects and their associated work class for the KYC Framework Layer.

Prefix	Description	Applies To This Class
KYCNBO-	Used for each New Business Onboarding Case type work object created	PegaKYC-Work-NewBusiness
KYCFLB-	Used for each New Business Onboarding Case type work object created	PegaKYC-Work-FATCALookBack
KYCREV-	Used for each New Business Onboarding Case type work object created	PegaKYC-Work-KYCReview
KYCMASTER-	Used for each unique folder work object associated with a customer (one MASTER- per customer)	PegaKYC-Work-Folder

This table lists the prefixes for the work objects and their associated work class for the sample KYC Application Layer.

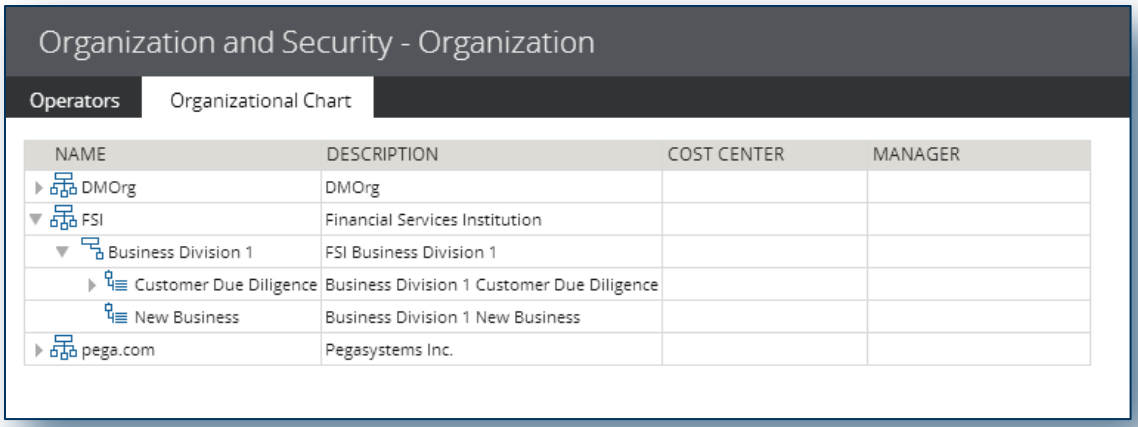
Prefix	Description	Applies To This Class
NBO-	Used for each New Business Onboarding Case type work object created	FSI-MyCoBankKYC-Work-NewBusiness
FLB-	Used for each New Business Onboarding Case type work object created	FSI-MyCoBankKYC-Work-FATCALookBack
REV-	Used for each New Business Onboarding Case type work object created	FSI-MyCoBankKYC-Work-KYCReview
MASTER-	Used for each unique folder work object associated with a customer (one MASTER- per customer)	FSI-MyCoBankKYC-Work-Folder

You can change a prefix or create additional prefixes by updating the **Details Tab** of the Application Rule.

Organizational Structure

The framework has a predefined organization, division, and organizational unit.

Select the DesignerStudio™ > Org & Security > Organization > Organization Chart landing page option to display the organizational structure.



Work Groups and Workbaskets

Users are given access to particular workbaskets for processing work. The work group of the user determines which workbaskets they can access.

Select the DesignerStudio™ > Org & Security > Tools > Work Baskets landing page option to display a list of workbaskets and their associated work groups.

This table lists the main default work groups and workbaskets installed with the framework. They are associated with the PegaKYC RuleSet.

Workbasket	Work Group
KYCCases	KYCAnalysts
RelationshipManagers	KYCRelationshipMgrs

Operators, Access Groups and Portals

The framework is installed with a sample set of operators, access groups, and user portals. The password is set to install. These allow you to access the Designer Studio to view and configure the underlying processes and rules or launch business processes from a variety of user roles including managers and users.

Select the DesignerStudio™ > Org & Security> Organization > Operators landing page option to display a list of operators.

Select the DesignerStudio™ > Org & Security > Security > Access Groups landing page option to display a list of access groups and access roles.

This table lists the operators, access groups, and their associated portal rule.

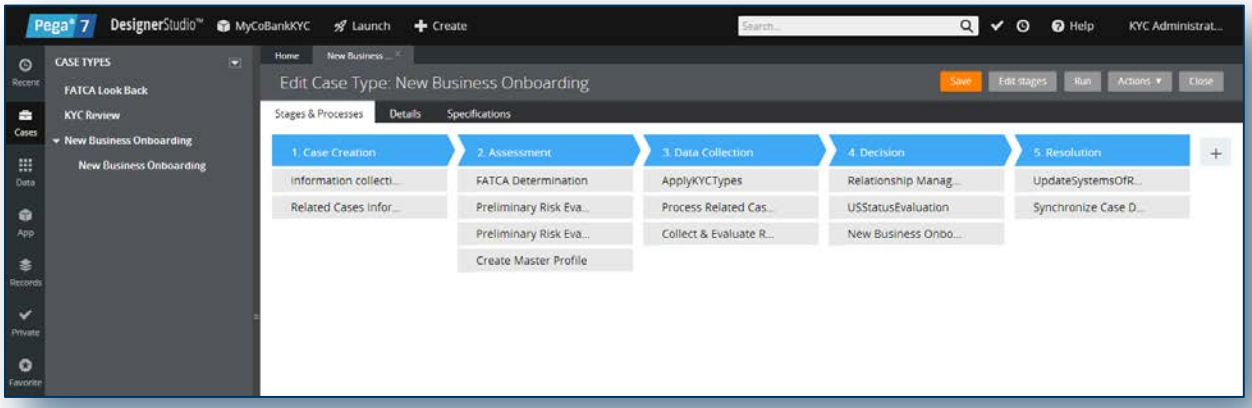
Operator ID	Access Group	Portal Rule
Diya.Cleveland@fsi.com	KYCAnalystManager	KYCCaseManager
Ben.Ballard@fsi.com	KYCAnalystUser	KYCCaseWorker
Anand.Bullock@fsi.com	KYCAnalystUser	KYCCaseWorker
Joan.Byrd@fsi.com	KYCAnalystUser	KYCCaseWorker
Sarah.Farley@fsi.com	KYCAnalystUser	KYCCaseWorker
Tyler.Fowler@fsi.com	KYCAnalystUser	KYCCaseWorker
Samuel.Gill@fsi.com	RelMgrSupervisor	KYCCaseManager
Aline.Hunter@fsi.com	RelMgrUser	KYCCaseWorker
Nathan.Harrell@fsi.com	RelMgrUser	KYCCaseWorker
Kane.Mason@fsi.com	RelMgrUser	KYCCaseWorker
Thane.Howard@fsi.com	RelMgrUser	KYCCaseWorker
Cynthia.Preston@fsi.com	RelMgrUser	KYCCaseWorker
Shannon.Sellers@fsi.com	KYCRuleManager	KYCRuleManager

Case Types

When implementing the framework, the system can be configured to utilize PRPC's Case Management capabilities. This provides you with:

- a less rigid structure
- more flexibility in the ordering of tasks and deciding which tasks are needed
- a view of the relationship of all cases

There are three case types shipped with the framework. You can create new case types from the Designer Studio by accessing the Cases landing page.



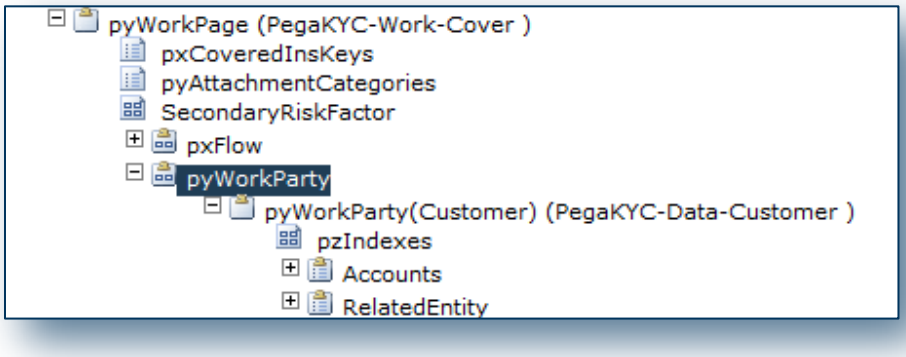
In KYC, Parent-Child related case creation is supported. For example: in KYC, the New Business Onboarding case type has a child of the same case type — New Business Onboarding.

Work Parties

Work parties rules define which participants (roles) can be associated with a work object. Each work object can contain many roles in addition to the required originator role. Some roles may participate with multiple occurrences. This rule also controls how and whether users can add parties using the user forms.

The preconfigured processes support the following work party:

- Customer



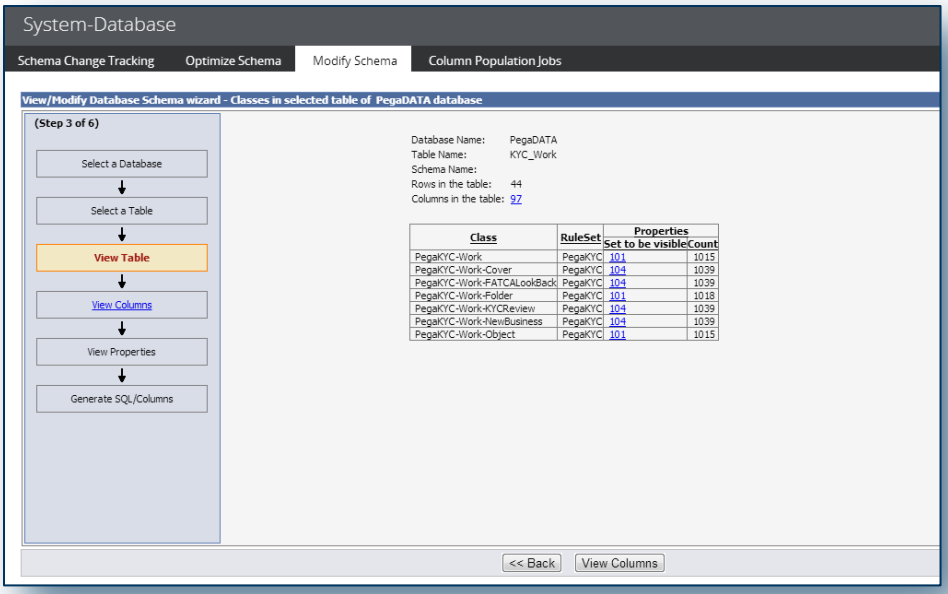
Sample Database Model

The framework uses a database table named KYC_Work in the PegaRULES database to store Case and Master Folder (Profile) data.

Note: When you implement external interfaces, your interface activities should map your data back to this work table.

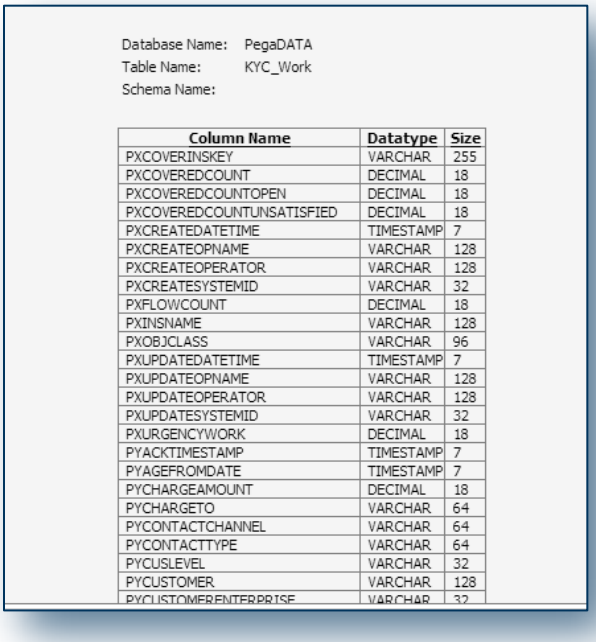
To view the database table and its properties

1. You can view a list of the database tables in the framework and PRPC from the Designer Studio by selecting the DesignerStudio™ > System> Database > Modify Schema landing page option.
2. Select the PegaDATA database and click **Next>>**.
3. Select the KYC_WORK table from the list and click **Next>>** to display the List of Classes in the table.



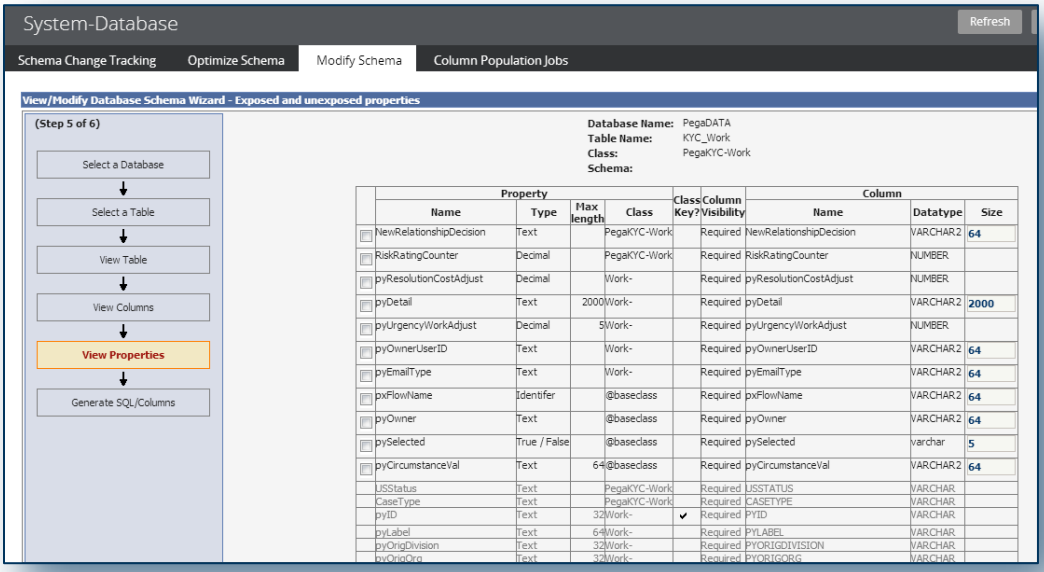
4. View the columns and the properties.

   Click the number after **Columns in this table** to view a table of database columns along with their data type and size.



5. Click **<<Back.**

6. Click the number in the Properties **Set to be visible** column to display a detailed table of the database properties.

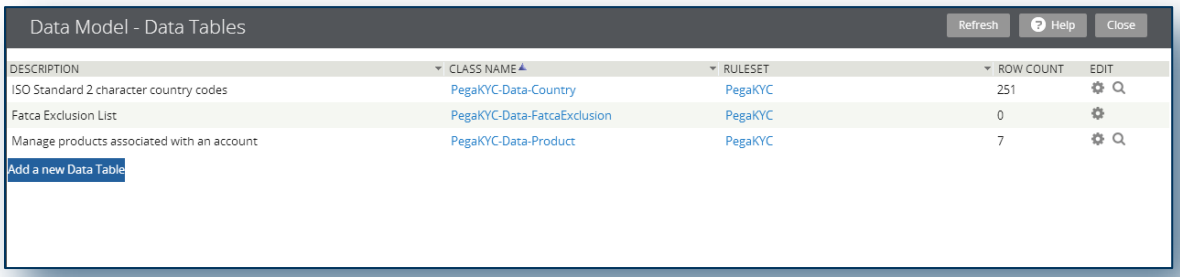


Note: As part of your deployment, it is recommended that you work with your database administrator to customize and tune the database tables for optimal performance.

Data Tables

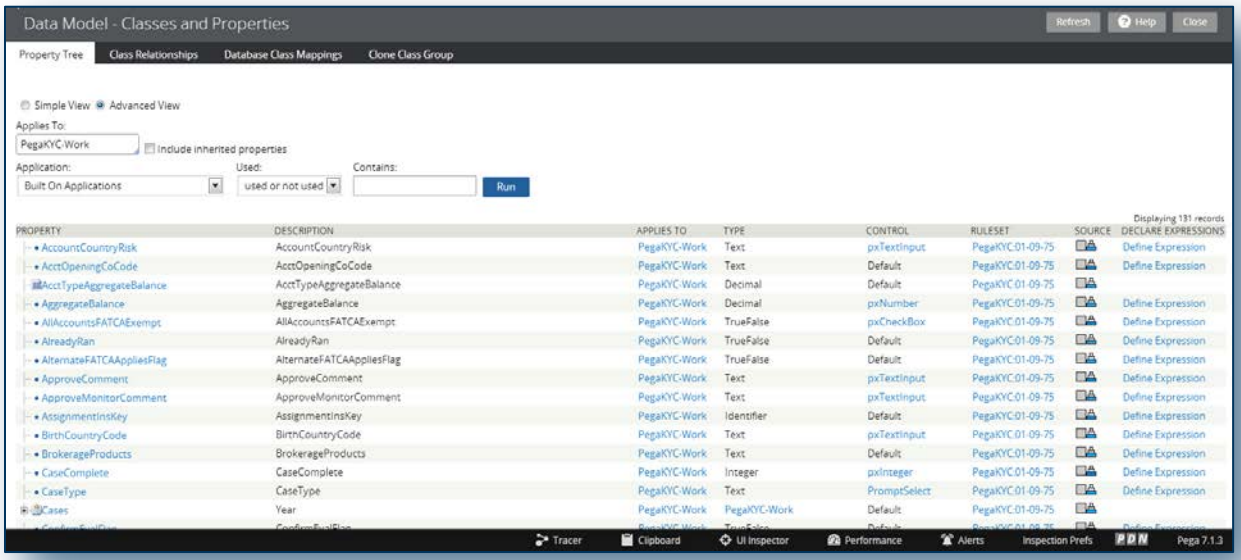
The framework contains a number of data tables that store data values referenced when processing KYC Cases.

The tables and the sample instances can be viewed and edited by selecting the DesignerStudio™ >Data Model > Data Tables > Data Tables landing page option.



Properties

You can display a list of properties in the framework by selecting the DesignerStudio™ > Data Model> Classes & Properties> Property Tree landing page option. Filtering the tree by the Applies To class and an Application you can drill down through the layers of the framework to view all properties



Chapter 3: KYC Baseline Performance Measures

This chapter provides information about how to achieve the best system performance when configuring the KYC capabilities described in this guide.

When designing the framework, a specific rule configuration was implemented to achieve optimal system performance. The result is a set of out-of-the-box system performance metrics focused on the primary KYC processing engine. The out-of-the-box configuration and metrics can be used as a benchmark for your customization and testing.

Performance Note: Incorrect customization of the core processing rules detailed in this guide can negatively impact your system performance. Consult the KYC product team for assistance if your implementation requires changes to out-of-the-box processing.

Specifically, the approach addressed maximizing performance metrics for:

- KYC Type selection
- Loading of data on the clipboard for dynamic evaluation and UI Display

Optimal performance was achieved as follows:

- A majority of standard activities involved in KYC Type and Item processing were changed to Data Transform rules.
- Using report definition rules for filtering available KYC Types to select which are applicable for the required scenario.
- Using Decision Table functionality to execute various condition rules.
- The logic for Item loading and initialization was enhanced to improve case creation time.
- Deferred initialization was implemented to gain a performance edge when there are high numbers of KYC Items. This new approach, unlike the older framework versions, loads the items but does not run the decision rules on them.
- Several of the processing tasks that were performed at the Item loading on older versions of the system are now performed while saving the KYC Type Rule
- The Data Collection form submission logic was changed to do faster submission.

Performance Associated Rules

This table lists the rules associated with type and item processing that was measured for performance.

Performance Configuration Note: Implementation teams, partners and clients should **not** customize any of the rules listed in this table without first consulting with the KYC product team to avoid degradation of your system performance.

Rule Type	Name	Class	Function	Affected Area
Activity	DetermineValidKYCTypes	PegaKYC-Work	Determines the applicable KYC Types and prepares them for further processing	Case creation
Report Definition	FilterKYCTypes	Rule-PegaKYC-Type	Used to filter KYC types based on selected countries and products in the KYC Type rules. This report is executed from the DetermineValidKYCTypes and ReEvaluateKYCTypes activities.	Case Creation and switching between the types
Report Definition	GetLatestKYCType	Rule-PegaKYC-Type	This report is used to get the highest version of KYC Types. This is a sub-report of the primary FilterKYCTypes report	Case Creation and switching between the types
Data Page	D_TempKYCTypes	Rule-PegaKYC-Type	This Data Page brings the list of Applicable KYC Types based on Selected Countries and Products. This Data Page sources the FilterKYCTypes Report Definition	Case Creation
Decision Table	EvaluateConditions	PegaKYC-Data-	The decision table is used to execute the various condition type for the KYC Type	Case Creation and switching between the types
Data Transform	PrepareKYCItems	PegaKYC-Work	KYC Type and KYC Type level logic; also uses InitializeKYCType as a sub data transform to load the items	Case creation
Data Transform	InitializeKYCType	PegaKYC-Data-Type	Loads the items in the following two conditions: 1. KYC type does not have any items	Case creation

Rule Type	Name	Class	Function	Affected Area
			2. KYC type copied from the master has an update version in the system It also checks if the initialization has to be deferred and defers the evaluation of the KYC items	
Report Definition	FilterKYCItems	Rule-PegaKYC-Type	This Report Definition brings the list of Associated Items of the KYC Types. This report is executed from the PrepareKYCItems Data Transform	Case Creation
Data Page	D_KYCItems	Rule-PegaKYC-Type	This Data Page brings the list of Applicable KYC Items based on selected KYC Type. This Data Page sources the FilterKYCItems Report Definition	Case Creation
Data Transform	ReevalValidQuestions	PegaKYC-Data-Type	Evaluates the KYC items for the various decision rules defined on the items; also sets the required properties on the KYC type based on the item initialization	Case creation and switching between types
Activity	SubmitWithValidation	PegaKYC-Work	Submits the data collection form while performing custom validation; allows submitting the data collection from any step on the screen	Data collection form submission

Baseline Performance Analysis Results

This chart shows the baseline performance metrics that you can expect to achieve when KYC is configured as described in the chapters of this guide.

				Creating Case			Switching between types			Submitting data collection form			
Number of Items	Display	Deferred Initialization /Number of items evaluated on first display	Decisioning Logic on the Items	Min	Max	Average	Min	Max	Average	Min	Max	Average	
100	Y	N	N	1	1.62	1.31	0.42	0.56	0.49	0.5	0.84	0.67	Base Line
100	Y	N	Y	2.1	2.44	2.27	0.59	0.78	0.685	1.12	1.81	1.465	Extreme
100	Y	Y/20	Y	1.6	1.89	1.745	0.66	0.89	0.775	1.3	1.73	1.515	
100	N	N	N	1.14	1.73	1.435							
100	N	N	Y	1.56	1.8	1.68							
100	N	Y/0	Y	0.88	1.56	1.22							
500	Y	N	N	3.14	4	3.57	1	1.27	1.135	1.4	1.76	1.58	Base Line
500	Y	N	Y	5	7.11	6.055	1.6	2.45	2.025	2.46	2.8	2.63	Extreme
500	Y	Y/100	Y	2	2.94	2.47	1.53	2.19	1.86	2.3	2.66	2.48	
500	N	N	Y	4	4.3	4.15							
500	N	N	N	1.4	1.6	1.5							
500	N	Y/0	Y	0.82	0.87	0.845							
1000	Y	N	N	5.69	6.1	5.895	1.7	1.9	1.8	2.75	2.89	2.82	Base Line
1000	Y	N	Y	9.45	10.6	10.025	2.34	2.8	2.57	2.68	2.8	2.74	Extreme
1000	Y	Y/200	Y	4	4.5	4.25	3	3.47	3.235	4.74	4.91	4.825	
1000	N	N	Y	7.43	8	7.715							
1000	N	N	N	3.29	3.5	3.395							
1000	N	Y/0	Y	1.8	2	1.9							

These metrics are the result of tests that focused on rules affecting case creation. Any initialization of data required **before** KYC types are selected and applied and any actions to be taken **after** KYC type selection and initialization are expected to be customized for implementation and not included in these readings. The baseline reading measurements are represented in **seconds**.

The following describes the meaning/context of the column labels displayed in the chart.

Number of Items

Refers to the number of KYC items configured in each KYC Type rule for performance testing

- 100 = 100 Items in 1 KYC Type
- 500 = 100 Items across 5 KYC Types
- 1000 = 100 Items across 10 KYC Types

Display

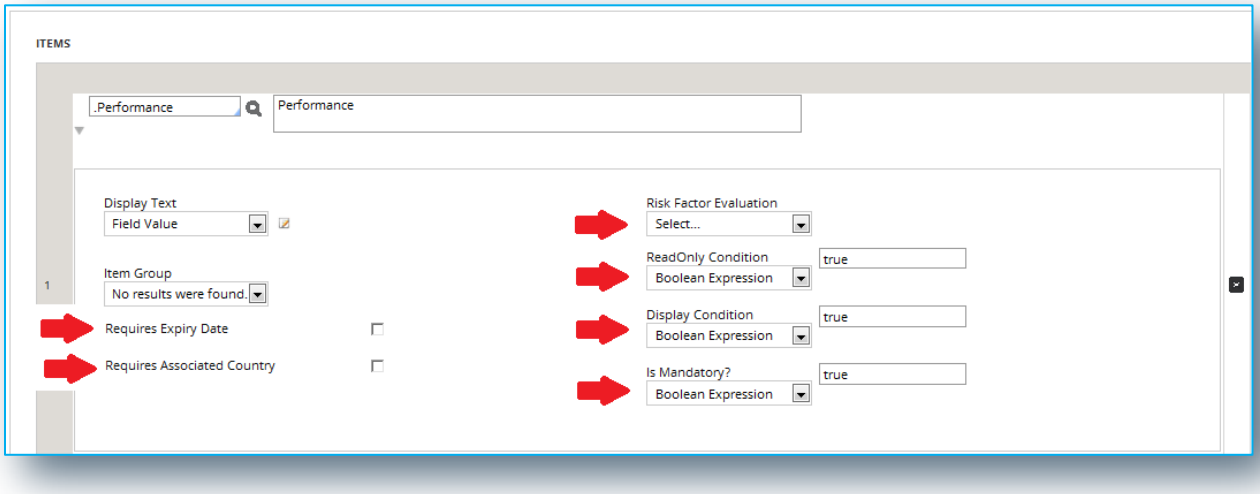
Refers to whether the KYC types are immediately displayed on the user interface after loading them. It has been observed that if numbers of items increase, then the time taken to display items on the UI also increases.

For the performance testing with the display turned off, the flow CreateKYCManually was updated to end after the clipboard was populated with the KYC types thus skipping the display.

Deferred Initialization

Deferred loading is an approach that can be used to reduce overhead when loading a high volume of Items across various KYC types. In this approach, a flag named DeferKYCItemInitialization has to be set in the dynamic system settings and the declare page Declare_PegaKYC_Settings has to be reloaded.

When this defer flag is set to **true**, the KYC questions are only loaded (initialized) on to the KYC Type page but **not evaluated** for display (mandatory, visibility, risk rating, etc…. This saves time in the case creation process when there is complex logic defined in the following item definition fields:



Number of Items evaluated on first display

When Deferred Initialization is configured, the item definitions shown above are evaluated only for the first KYC Type loaded for the case - this will depend on how the ordering of KYC Types has been configured. The metric on the chart refers to the number of items evaluated when the first KYC Type loaded during baseline testing.

Decisioning Logic on the Items

Refers to whether or not the KYC Items had attributes configured for display such as Is Mandatory, Read-only, etc.

Creating a case

These values were taken directly from the PRPC PAL utility (Performance Analyzer) for the given scenario and refers to the time taken to create a new case when KYC types are applied to it.

Switching between types

These values were taken directly from PAL for the given scenario and refer to the time taken to move from one KYC type to another during the evaluation.

Submitting data collection form

These values were taken directly from PAL for the given scenario and refer to the time taken to submit the case and all KYC type and item data after the KYC evaluation is complete

Baseline Testing Results and Observations

The performance testing and results provided insight and information about KYC processing that can be beneficial to you when customizing your implementation.

Testing Notes

- When the decision logic was applied for the baseline testing, it was applied to all the decision attributes available for each item. However, in a production environment, these attributes will not be configured the same way and for every item. Hence, the time required to create a KYC case may be reduced by several seconds.
- The **Baseline** value is the time taken to load the KYC types when no decision logic is applied to any of the items. In a typical scenario it takes 0.003 seconds to load one KYC item. However when any decision logic is applied to a KYC item, the time taken to execute the decision and related rules gets added to that time.
- The **Extreme** value is the time taken to load all KYC items (100, 500, 1000) with the decision rules defined on them.
- For testing, Boolean type decision rules - i.e. always or never - were configured in the **Display**, **is Mandatory** and **Read Only** attribute settings.
- For **Risk Factor Evaluation**, a slightly more complex decision rule with two columns and one row was configured.

Testing Observations

- As the number of KYC items configured for each type increases, the time taken to render the UI display also increases.
- When there are more KYC items on a particular KYC type, the time required to switch to that type increases.
- The data collection form submission (FINISH button) shows a minor increase of not more than 1 second for every 500 questions.

Configuration Recommendations

When possible, it is recommended that you follow these configuration guidelines when you are implementing your KYC processes.

- Limit the number of questions to a range of 50-75 per KYC type.
- Use the decision logic attributes on KYC items only when absolutely necessary.
- Limit the number of KYC types applied to a KYC case to 5-6.
- Divide the KYC questions applied to a KYC type into groups based on a specific visibility condition. These groups can then be put together in a KYC Type and the visibility condition can be applied on the type as a whole.

Chapter 4: Configuring KYC Types and Items

This chapter walks you through the steps required to create and configure KYC Type rules and their associated Items which are used to dynamically display due diligence questions and requirements to the end user within a KYC Case.

Topics are:

- Common KYC Terminology
- Configuring KYC Types and Items

Common KYC Terminology

This table describes common terms referenced in this document when describing the processing, configuration and user interface displays in the KYC framework.

Term	Definitions
Driver Data	Key customer information that includes, but not limited to: • Customer ID, name and contacts • Customer type / subtype information • Country of incorporation • Country of domicile • Products (accounts, insurance policies) This data is expected to be passed to the KYC system by on-boarding applications or retrieved by the KYC system during manual creation of a KYC Case. This data is a key component for evaluating risk and applying KYC requirements.
KYC Type	Custom rule that defines how and when data pages, called **KYCTypes,** are to be instantiated within a Case. Primary function of the KYC Type rule and **KYCType** pages is to conditionally display Items and store captured Item data. KYC regulations as well as internal policies and procedures translate to KYC Type rules. e.g. – Enhanced Due Diligence, Standard Due Diligence, etc.
Item	Items are conditionally displayed, individual data points such as a question and response or a document reference. Attributes that define the display of each Item are defined on an embedded page (of **KYCType** pages) called **ItemList**. Display attributes are defined within the **ItemList** page and presented to the user at runtime in various formats. Item's response value is stored on the associated **KYCType** page

Term	Definitions
Master Profile	PRPC folder work object used to store the most current and approved due diligence data. Also stores pertinent risk information and Case history. The folder stores completed and approved **KYCType** and embedded **ItemList** page data
KYC Case	PRPC cover work object used to manage and process an individual Item of work when due diligence must be performed. e.g. – address change, risk change, new business, etc.

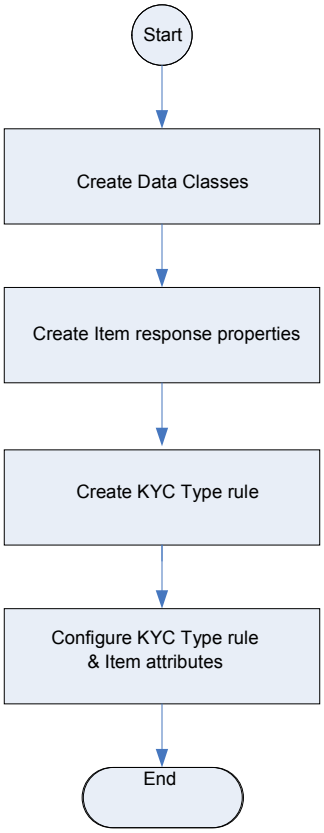
Configuring KYC Types and Items

This topic describes the process and steps required to set up the class structure to create and configure KYC types and Items for specialized KYC applications. At a high level, the steps are:

1. Create Data classes
2. Create Item response properties
3. Create instances of KYCType Rules
4. Configure KYC Types rules and Item attributes

Important! The custom KYC Type rule is the primary asset of the KYC framework. This custom rule is available for use when specialized KYC applications are built on top of the PegaKYC framework layer.

The following diagram summarizes the high-level sequence of steps when creating and configuring KYC Type rules for specialized KYC applications.



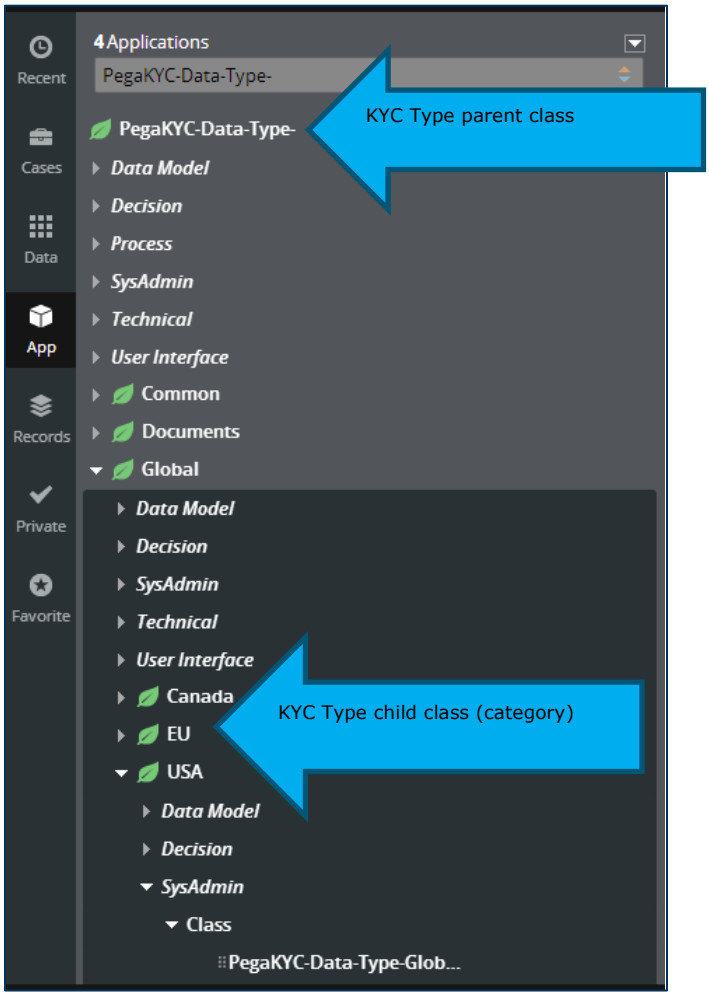
Step 1 - Create Data classes

Data classes are used to store and reference KYC Type rules. It is expected that when a specialized KYC application is implemented and extended from the KYC framework, unique data classes will be created to store specialized KYC Type rules.

The KYC framework is shipped with a sample set of data classes and KYC Type rules that serve as both a design model as well as reusable assets for production implementations.

You can view sample data classes and how they can be structured by signing on to the KYC framework as **KYCSysAdmin** with the password of: **install**

In the Application Explorer, navigate to the PegaKYC-Data-Type- class. This top-level data class, and the child classes that extend and inherit from it, contain pre-configured KYC Type rules that belong to the SampleFSDueDiligence ruleset.



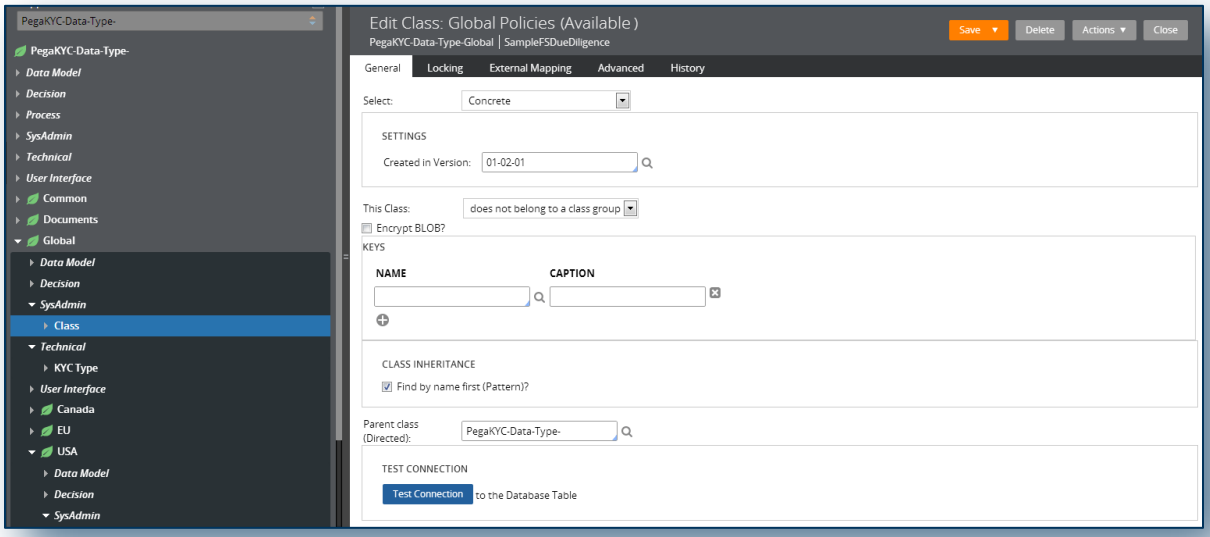
Important! Upon implementation of a specialized KYC application and associated RuleSet, specialized KYC Types must be created in new, specialized data classes that are designed to best organize the institution's unique KYC requirements.

Careful design and planning of the class structure are required so as to promote re-use and to minimize maintenance of KYC Types.

Your institution's specialized data classes must ultimately inherit from the PegaKYC-Data-Type- class to inherit the core processing assets of the KYC framework.

When a specialized KYC application is extended from the KYC framework, the top-level MyCoKYC data class should be configured to directly inherit from PegaKYC-Data-Type-. If necessary, the specialized application's top-level data class can be used to store overrides of the core KYC processing rules and assets.

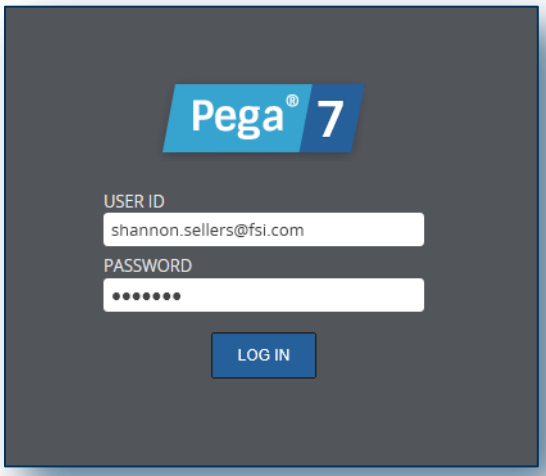
A sample Global class is shipped with the KYC framework. This class is included in the sample application layer and contains several sub-classes and out-of-the-box KYC Type rules that can be copied and re-used as necessary. The Global class directly inherits from PegaKYC-Data-Type-.



Upon implementation, other classes (categories) can be created to inherit from the MyCo-Data-Type-Global class, which will directly inherit from the parent PegaKYC-Data-Type- class. These specialized classes become categories of KYC Types and are used to store KYC Type rules organized and specialized by country, region, product, etc..

You can see a sample of this class structure and how it is displayed to a business user by signing on to the system with a sample business user operator ID shipped with the KYC framework.

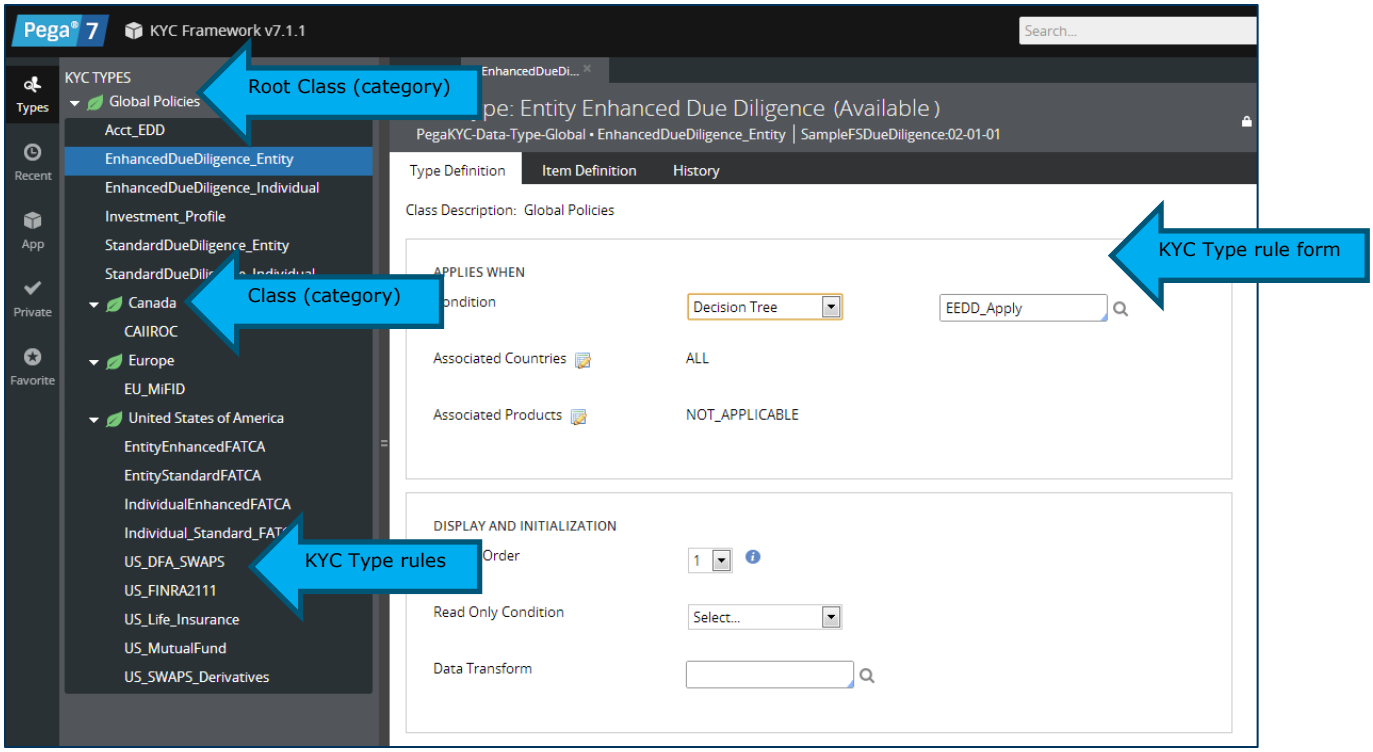
Log off and log back on to the system as **Shannon.sellers@fsi.com.** The password is **install**.



The business user's portal is a simplified version of the developer portal that allows the user to focus only on the creation and maintenance of KYC Type rules.

The sample KYC application shipped with the KYC framework provides the access, operator and portal settings required to view the data classes containing sample KYC Type rules.

The data classes defined in the specialized KYC application appear as categories on the left panel which can be expanded to display the KYC Type rules that have been saved to it.

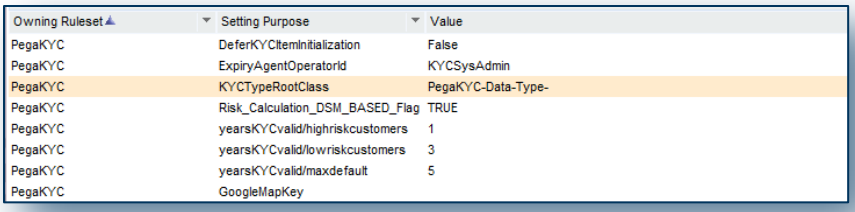


Important! To define which data class appears as the **root**class in the left panel, you must configure the **KYCTypeRootClass** dynamic system setting. For implementation, the **common / global** data class should be configured as the root category on the business user portal.

To configure the KYCTypeRootClass setting:

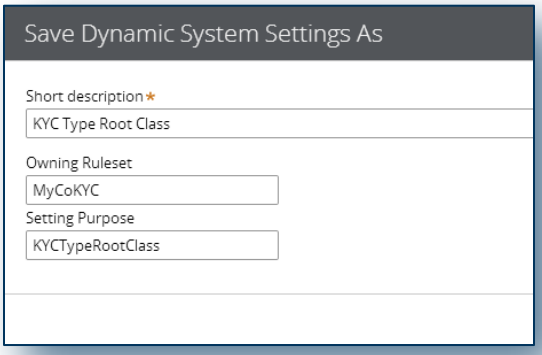
1. Log on as system administrator.
2. Navigate to the Rules Explorer in the developer portal.
3. Open the SysAdmin rule category.
4. Click Dynamic System Settings.

   In the list of dynamic system setting rules, locate and open the KYCTypeRootClass rule associated with the PegaKYC ruleset.



Owning Ruleset	Setting Purpose	Value
PegaKYC	DeferKYCItemInitialization	False
PegaKYC	ExpiryAgentOperatorId	KYCSysAdmin
PegaKYC	KYCTypeRootClass	PegaKYC-Data-Type-
PegaKYC	Risk_Calculation_DSM_BASED_Flag	TRUE
PegaKYC	yearsKYCvalid/highriskcustomers	1
PegaKYC	yearsKYCvalid/lowriskcustomers	3
PegaKYC	yearsKYCvalid/maxdefault	5
PegaKYC	GoogleMapKey	

5. Save a new copy of the rule.
6. Enter the application's owning ruleset and leave the setting purpose name — KYCTypeRootClass — as is.
7. Click **Save As**.

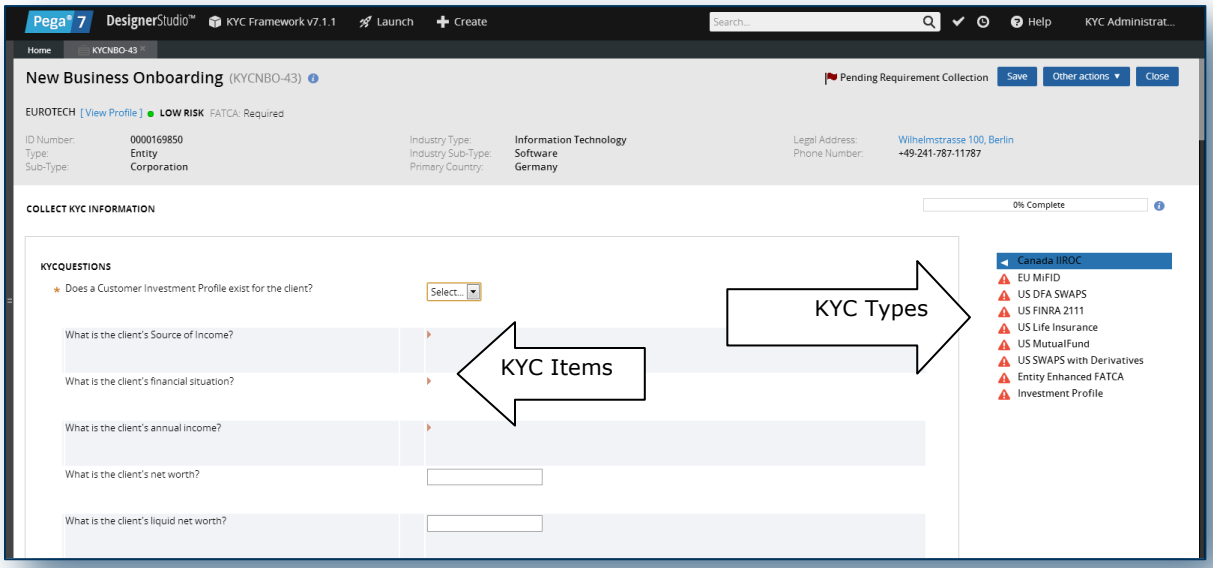


8. In the new rule form, specify the owning ruleset, click **Save** and then specify the class that will be the root category on the business user portal. For example: the MyCo-Data-Type-Common- class.

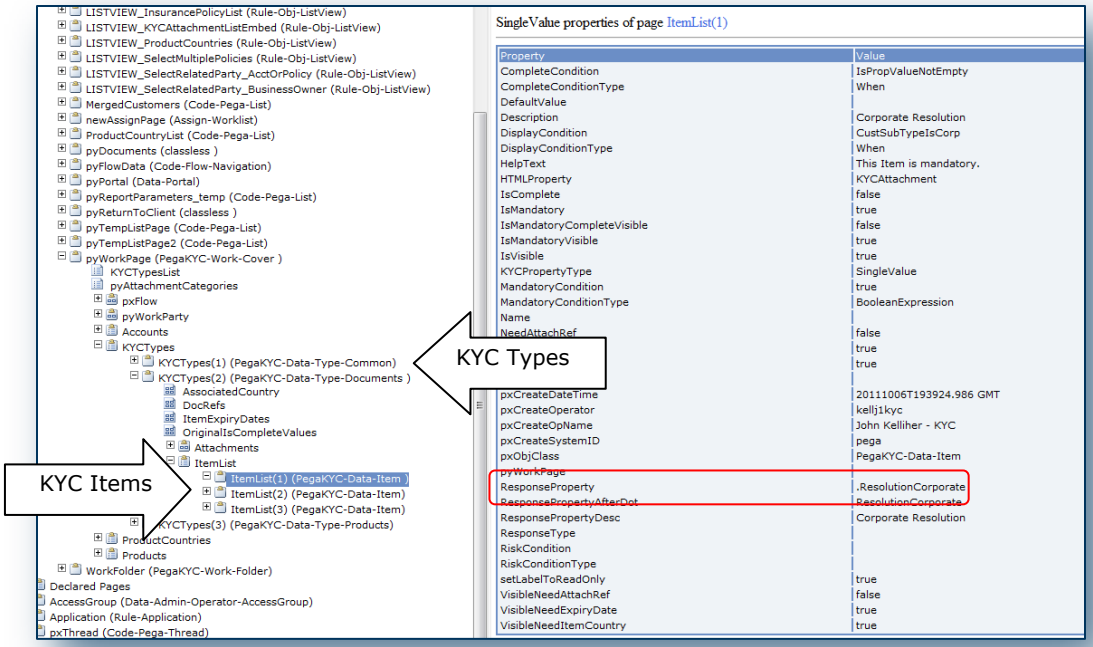
Once the proper class structure is designed and implemented, you can now create KYC Item response properties in the application's data classes.

Step 2 - Create Item Response Properties

In a KYC case, items are represented on KYC Type pages as individual rows where data is selected or entered. Each row's data capture field and associated attributes (expiration date, associated country, etc.) display only when a KYC Type's embedded **ItemList** page's logic conditions are met.



User Interface display of KYC Types and Items



Representation of KYC Types and Item on the clipboard

Each embedded **ItemList** page that is applicable and instantiated within a case's type, contains a singular property called .ResponseProperty.

The .ResponseProperty property actually references a property stored on the **KYCTypes** page that the embedded **ItemList** page belongs to.

Important! Each KYCTypes page stores data values associated with a KYC Item referenced by the KYC Type while each embedded ItemList page defines only the presentation of the KYC Item on the user interface. ItemList pages do not store response property values.

When creating KYC Type properties to store due diligence data, they can be created within a specific data class or in a higher level parent class for re-usability. These properties can be defined as a yes or no value, prompt select values or strings of text such as a document reference.

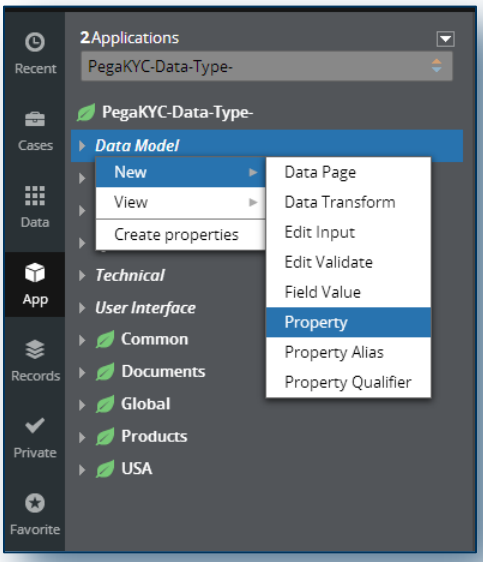
When referenced within a KYC Type rule, the .ResponseProperty property and other data attributes become part of the KYC Type's data structure when the page is instantiated in a case or Master Profile (folder).

To create a new KYC Item property that stores a data value on a KYC Type page:

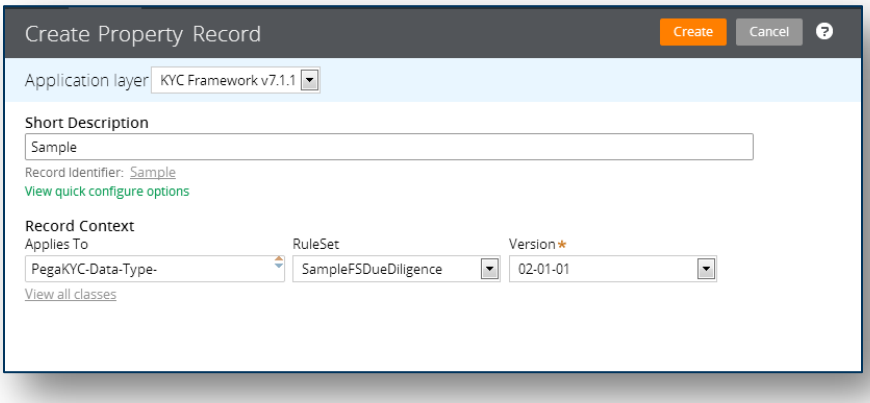
1. Right click the Data Model category in the data class where the property is to be referenced

Important! Commonly used properties should be created in a class that utilizes PRPC class inheritance to promote reusability. For example, a property that captures a value that can be referenced by other KYC Types stored in various classes, should be saved to a top-level class. i.e. – MyCo-Data-Type-Common-

2. Select **Data Model > New > Property**



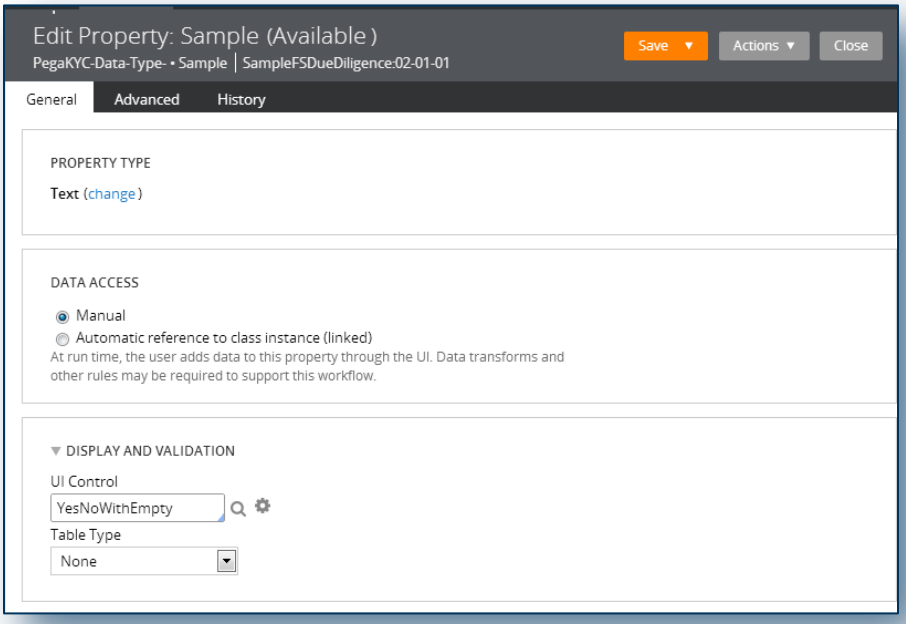
3. In the **Short Description** field, enter the name of the property.



4. Click **Create**.

5. In the **Display and Validation** section of the property, enter YesNoWithEmpty in the **Control** field.

   In this example, assume that the new property stores a YES/NO response value for a specific item.



Note: YesNoWithEmpty is a PegaKYC custom HTML property that is shipped with the framework. Upon implementation, new HTML properties can be created, or standard prpc properties can be referenced, to control the display of response properties in accordance with client's business requirements.

6. Click **Save**.

This table lists the most commonly used item property types and their respective controls.

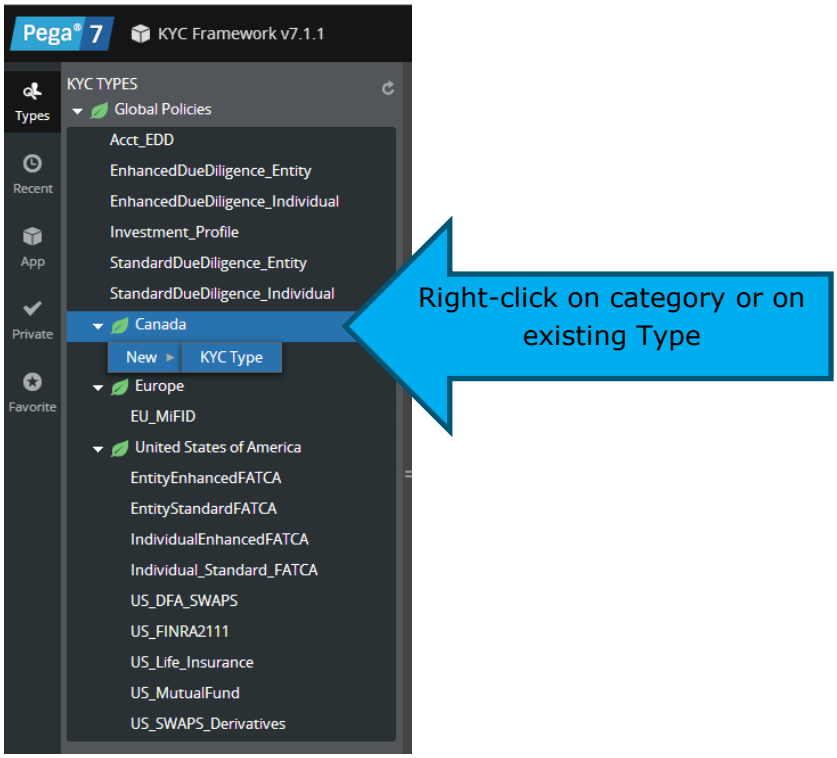
Property Type	Control
Text	Text
True/False	YesNoWithEmpty
Date/Calendar	Date
Attachment	KYCAttachment
TextArea	TextAreaWithExpandSmall
Currency	CurrencyAmount

Step 3 - Create instances of KYCType rules

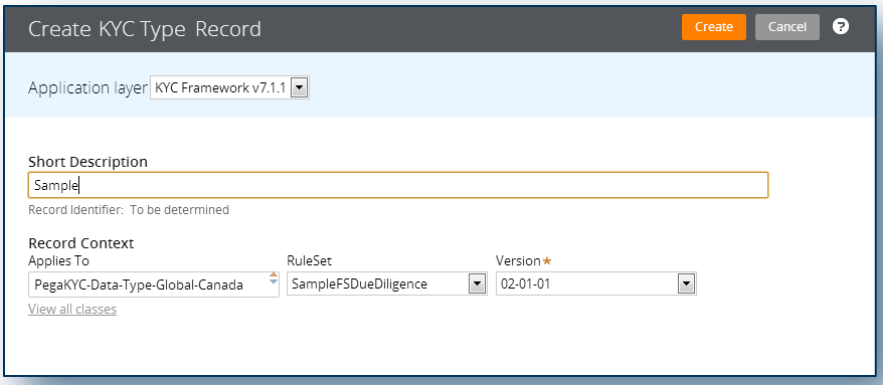
Now that the appropriate data class structure has been created and Item response properties have been created and saved to the appropriate class, you can create new **KYCType** rules (or modify existing ones). These rules will reference one or more Item properties, store their data values and define how the Item is presented to the user on the case user interface at run-time.

1. Log on as a business user to access the business user portal.
2. Navigate to the KYC TYPES Explorer view which shows the defined class structure.

   In the left panel displaying the available classes (categories), right-click on a category or within an expanded category and select New >KYCType.



3. The **Create record** rule form displays. Configure the rule as follows.
   a. In the **Short Description** field, enter the name of the KYC Type rule.
   b. Select the appropriate class name (category), ruleset and version that the rule will belong to.



4. Click **Create**.

   The new KYC Type rule form displays in the main panel for configuration and you are now ready to configure the KYC Type rule

This table lists the pre-configured KYC Types includes in KYC v7.1 and higher.

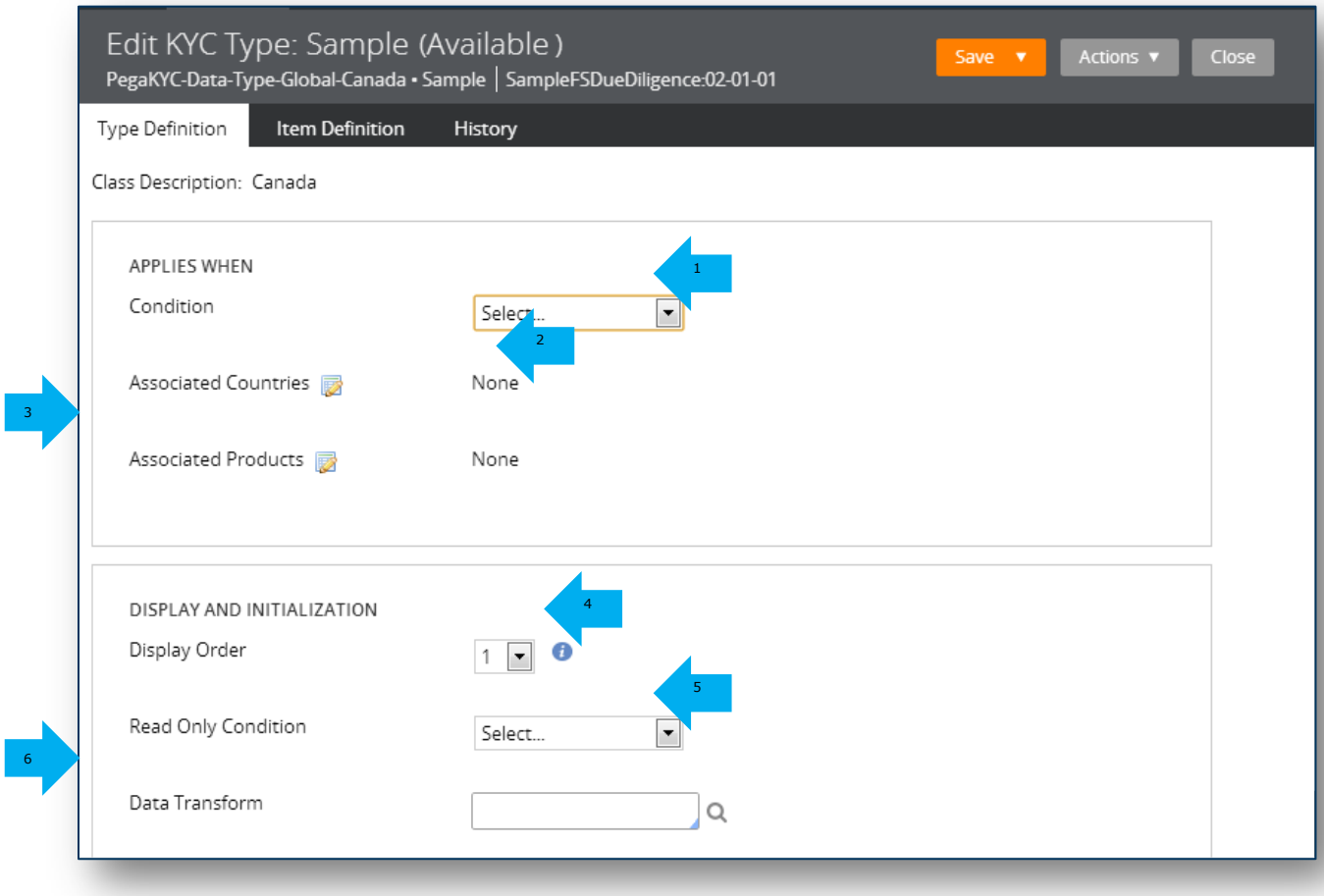
Global
ENTITY – Standard Due Diligence
INDIVIDUAL - Standard Due Diligence
ENTITY – Enhanced Due Diligence
INDIVIDUAL – Enhanced Due Diligence
Investment Profile
Account Enhanced Due Diligence
Suitability
CA IIROC
US FINRA 2111
US DFA SWAPS
EU MiFID
Products
US Brokerage
US Life Insurance
US Mutual Funds
US SWAPS with derivatives
US FATCA
Standard FATCA - Individual
Enhanced FATCA - Individual
Standard FATCA - Entity
Enhanced FATCA - Entity

Step 4 - Configure the KYC Type rule and Item attributes

The KYC Type ruleform is segmented into 3 main parts – Type Definition, Item Definition and History. This form lets business users configure all aspects of when and how a KYC Type and its associated Items appear in a case.

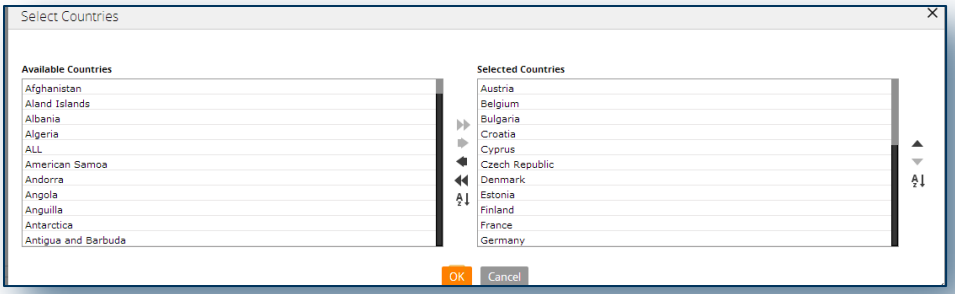
To help you better understand each configuration element of the KYC Type ruleform, each configuration attribute is numbered in the image below and a corresponding description is provided.

Type Definition



1. **Condition** — Used to configure when the KYC Type is applicable and instantiated for the case. In the sample, the type is set to always apply when the associated Data class applies by using a boolean expression of true. Specialized When conditions, Map Values, Decision Tables and Decision Trees can also be used.

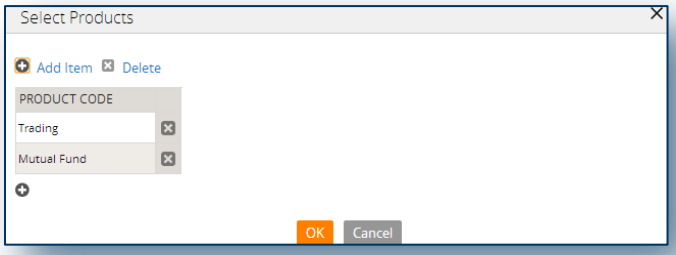
2. **Associated Countries** – Used to configure one or more applicable countries for KYC Type selection.
   Use the edit icon to open the list-to-list for country selection.



These data values are sourced from various systems (or manually entered) and are expected to be present on the clipboard. Having this driver data on the clipboard allows for more refined selection of KYC Types based on the presence of one or more countries.

NOTE: When creating a new Type rule, select **ALL** in the Associated Countries list-to-list if the Type does not apply to a specific country or set of countries. This ensures the Type will be considered for applicability. If the KYC Type is applicable only for a specific country or group of countries, then select the countries in the list-to-list modal window.

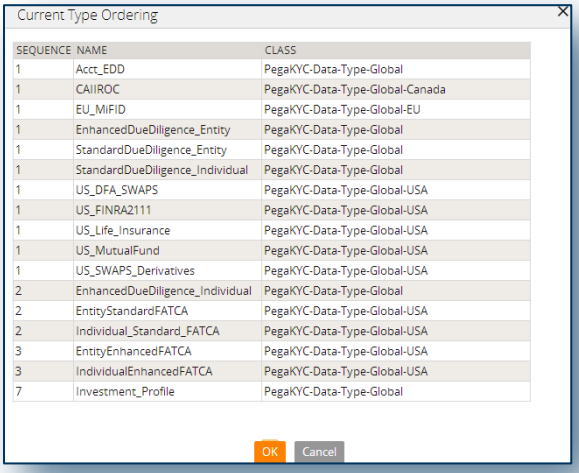
3. **Associated Products –** Used to configure one or more applicable products for KYC Type selection.
   Use the edit icon to open the modal dialog to enter one or more products.



These data values are sourced from various systems (or manually entered) and are expected to be present on the clipboard. Providing this data on the clipboard allows for more refined selection of KYC Types based on the presence of one or more products.

4. **Display Order –** Used to configure the Display order for the KYC Type on the case user interface when the KYC Type is selected. To view the overall KYC Type Ordering, click on the info icon to the right of the drop down list.

In this example, if StandardDueDiligence is selected to apply to the case, then it will be ordered first on the case UI because its sequence number is set at 1.

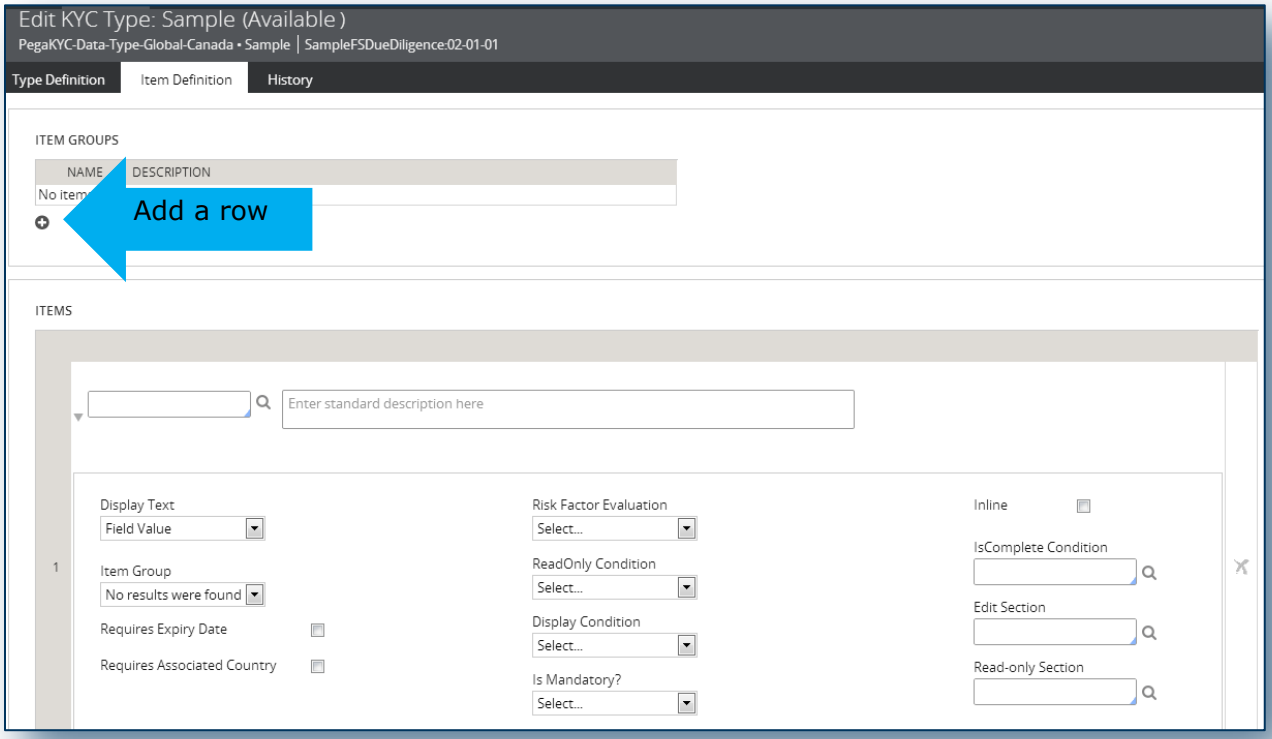


5. **Read Only Condition** — Used when the entire Type (all KYC Items) should be read only to the end user. In the example, the Type is set to be read-only when the associated Data class applies by using a boolean expression of true. Specialized when conditions, map values, decision tables and decision trees can also be used. When this read only condition evaluates to true all the Items associated to this type will be rendered as read-only to the user.

6. **Data Transform** — If the clipboard contains pages of data and values that can be used to set the values for applicable Items, then a data transform rule can be configured to initialize Item Response properties with the data values at the time of KYC Type initialization. This feature is typically used to prevent users from having to re-enter data if it already exists elsewhere on the clipboard.

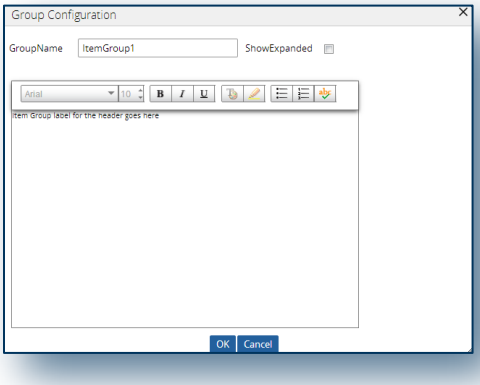
NOTE: In most cases you will also want to set the **IsComplete** property for the Item to true within the DataTransform rule as well. This will ensure the green checkmark indicating completeness is displayed.

Item Definition



Item Groups — Displays a list of available Item groups for the Type. Item Groups are used to logically group the display of Items together within a KYC type.

a. To add a new group, click the Add a Row icon in the Group Configuration list. A **GroupConfiguration** window displays. Configure the fields as described below.

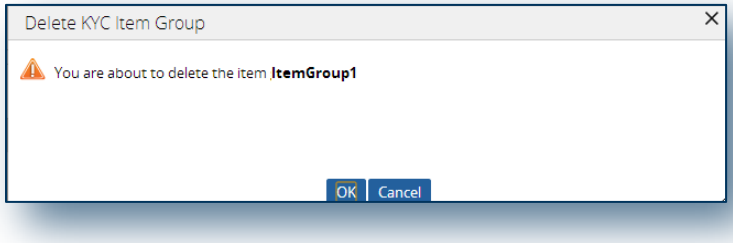


- **Group Name —** Use this property to define a group name for a group. Each group is uniquely identified by the group name. Any Item in the type can be associated to a group by using the group name.
- **Show Expanded —** Use this property to control the behavior of the expand/collapse feature used to display the group questions for a case. Checking the checkbox results in an expanded display of the group.
- **Group Description —** Use this rich text field to define a description for a group. This description will be visible on the header of the group for a case.

b. To delete a group use the Delete icon on the group list**.**
If there are any Items associated with a group being deleted, a modal window displays listing all Items associated with it.

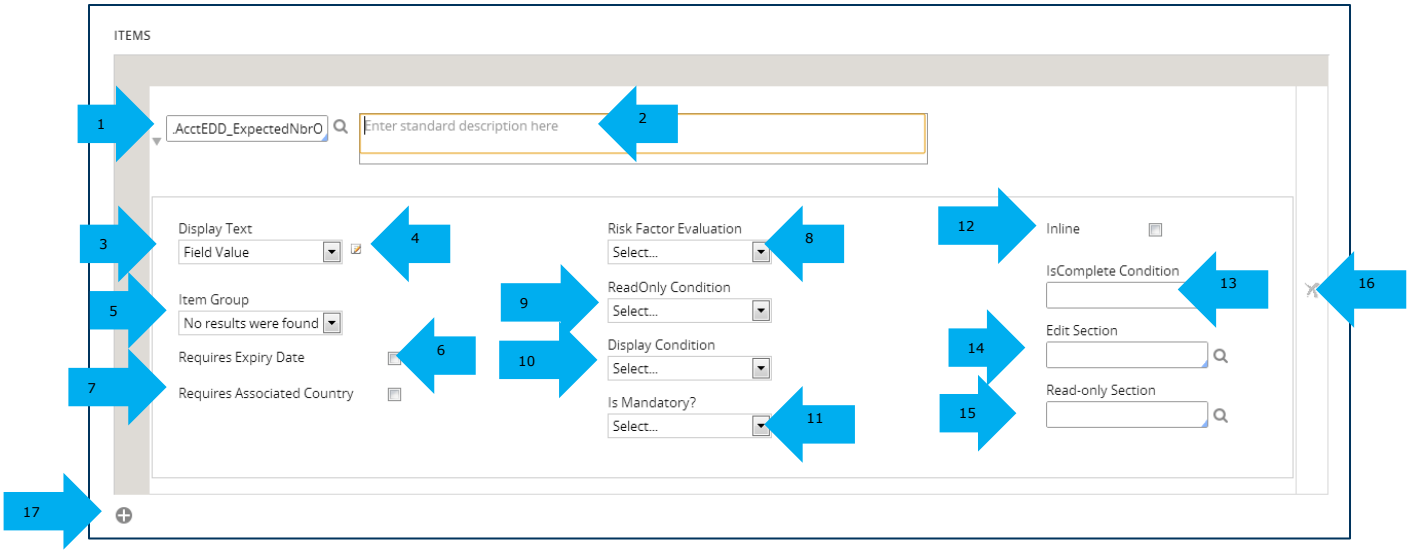
Access the attribute settings for the listed properties and remove the item group setting from the item.

Once the properties are unlinked from the group, you can continue to delete the entire Item Group.



Item Configuration

The configuration of each Item attribute that belongs to a KYC Type is configured as shown below. A reference number and associated description for each attribute is provided.



1. **Item Property reference** — Locate and select available properties using the smart prompt.

2. **Description** — This is visible on the KYC Rule form only and used for reference.

3. **Display Text** — Select the type of rule that will be used to display the Item text at run time on the case UI. The possible values for the display text options are field value or paragraph rule. (both can be localized)

4. **Edit icon —** Click to create new, or edit existing, field value or paragraph rule.

5. **Item Group** — If Items Groups have been created for this Type, use the dropdown to select from the list of available groups. When associated with an Item Group, the Item will be displayed in the specified group on the case. Leave this setting blank to keep the Item in the default pool.

6. **Requires Expiry Date** — When selected, the Item displays an additional, calendar property that allows the user to set a future, expires when date. Expiration dates can be set manually, automatically by an external system or by default. The earliest expiration date is rolled up to the type level and also rolled up to the Master Folder level. The Check Expiration agent will process each master folder on a user-defined schedule. If an expiration date is found, then a case is automatically created and the expired types that contain that contain the expired Item(s), are presented for reevaluation.

7. **Requires Associated Country** — When selected, the Item property displays an additional prompt-select property listing countries. The selected value will be stored on a separate page and NOT as a value on the **KYCType** page

8. **Risk Factor Evaluation** — When configured, conditional decision logic is evaluated to set or adjust the overall risk rating or a user-defined risk property. If a decision

rule is configured and referenced here, this logic is evaluated during the EvalSecondaryRisk process.

9. **ReadOnly Condition —** Used to configure when a specific Item is read only. The read only condition can be set as a simple boolean expression or specialized When conditions, map values, decision tree or decision table rules can also be used.

> **Important!** If a type's read-only condition is set on the KYC Type tab, then its logic will override logic set at the Item level.

10. **Display Condition** — Use decision rules (when, map value, decision tree, decision table or boolean) to create condition(s) when the Item is displayed on the case UI

11. **Is Mandatory?** — When a decision logic rule is configured and referenced in this setting, a response will be required for the Item and denoted with the default orange * icon. (Boolean Expression set to true makes the Item required by default.)

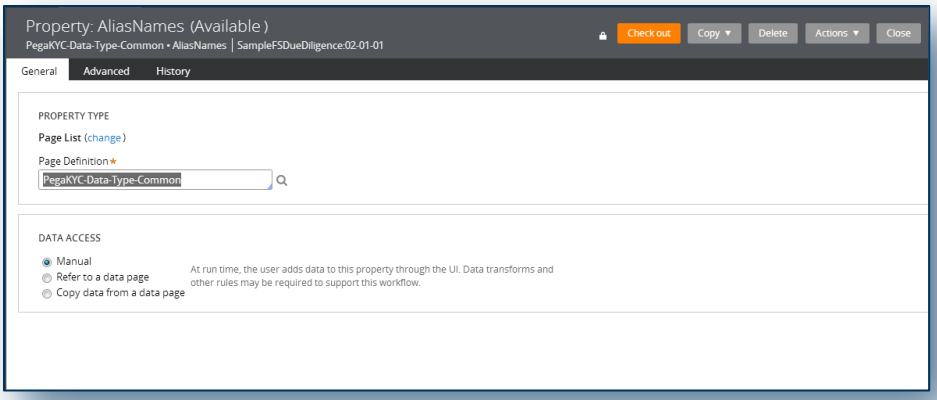
    If the Item is set as mandatory, then the Item will be marked as complete when the response property contains a value. This attribute works only for single value properties.

12. **Inline? —** When selected, the edit section will appear inline with the rest of the Items for the KYC Type. This is especially useful if there are complex and multiple line items to be completed and would not be well suited for a pop-up modal dialog. Using the prior, AliasNames example, you can see that the section is displayed inline with the other Items when the Type is rendered in the case user interface.

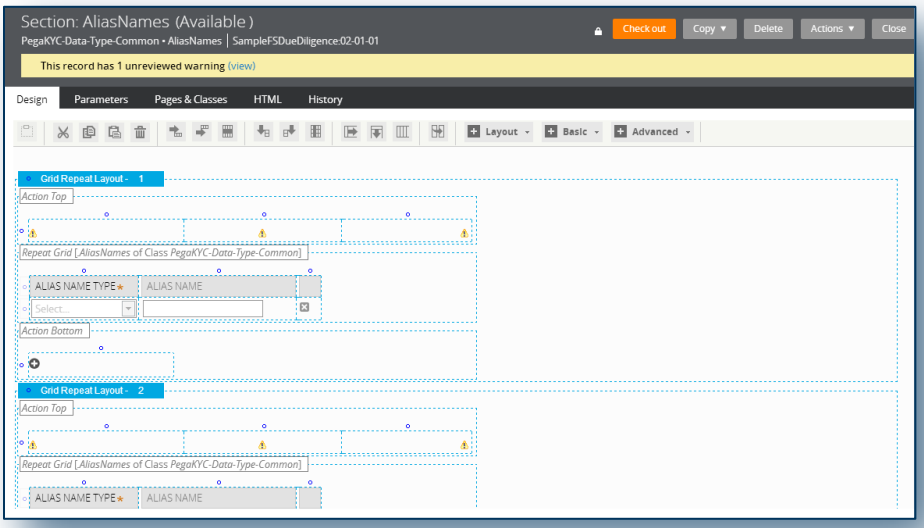
13. **Is Complete Condition** — When inline or read-only sections are configured within a KYC Type, use this attribute to define logic that evaluate completeness. This will warrant creation of a custom activity or data transform that can evaluate the values of a complex property and mark the Item as complete.

> **Important!** The **Is Mandatory?'** attribute will not work for complex data types referenced in sections.

14. **Edit Section** — When a KYC Type requires responses that consist of complex data types (Page lists, value lists, etc.), complex properties and sections can be created to display and capture the complex data values.
    An example of a complex response property and associated section rule is included in the PegaKYC data classes – the AliasNames page list property.

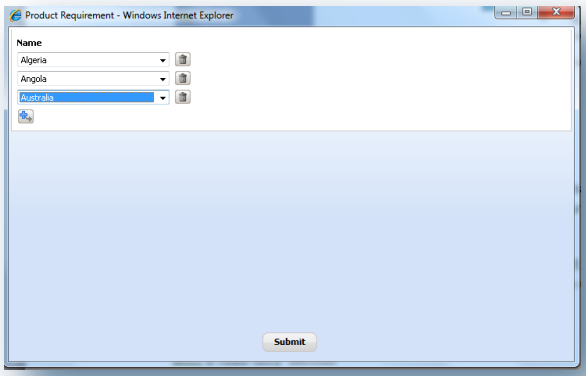


This property is referenced and configured within a section called AliasName which is contained in the same class as the response property.



The section is referenced within the Standard Due Diligence KYC Type's **Edit Section** attribute.

15. **Read-Only Section —** This attribute should be configured if business requirements require the capture of data in a modal window pop-up and when saved, the captured data should appear as read-only to the user.
    A sample of this feature is provided with the framework when the **Expected Activity** KYC Type is applied to a case. Within this Type, an Item is defined to prompt the user to add one or more values to a page list in a modal window section.



Clicking **Submit** saves the data to the clipboard property but, unlike the inline configured section, the section appears under the Item description and as read-only to the end user.

16. **Delete Row** — Click the Delete icon to delete an Item when defining type requirements.

17. **Add Row** — Click the Add icon to create a new Item when defining type requirements.

Adding Custom HTML Property rules

On implementation or for proof of concept, additional custom HTML property rules can be created for KYC Items. This is an optional step that is only needed if the out-of-the-box HTML properties do not meet your custom KYC Item response requirements.

The auto-generated UI that captures KYC Item responses uses ON CHANGE events to process responses and re-evaluate applicable KYC Items based on the response captured.

Important! Newly added HTML Properties must trigger **ON CHANGE** when the property value changes so that the system can process the response and re-evaluate applicable KYC Types and other Items based on that response.

Chapter 5: Advanced Configuration Options

This chapter describes the advanced options related to configuring KYC types and Items. Depending on your organization's specific business requirements, some or all of these options may be required for your implementation.

Topics are:

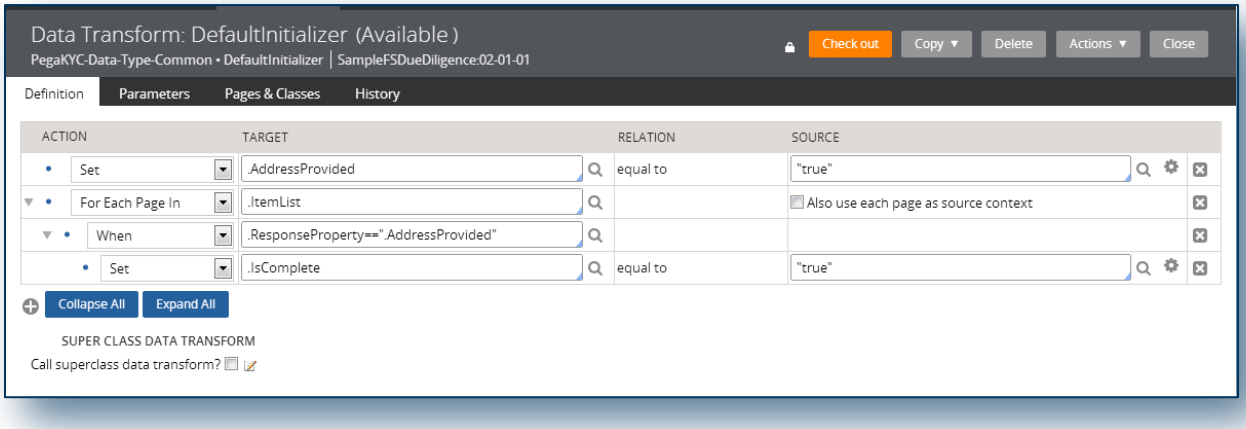
- Initializing KYC Type Data via Data Transform Rules
- Validating KYC Type Data
- Attaching Documents via Custom Controls
- Using Rich Text for Items
- Sharing Item Data Values Across KYC Types

Initializing KYC Type Data via DataTransform Rules

Data Transform rules can be referenced within the KYC Type to initialize Item response properties at the time of KYC Type inititialization.

The following is an example of the initialization transform rule that is specified on the Type configuration form.

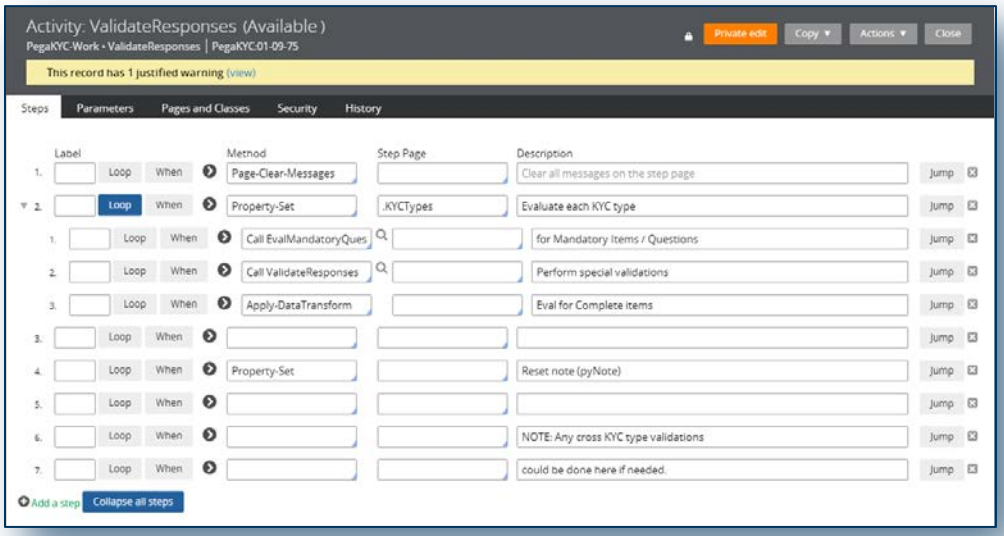
NOTE: In most cases you will also want to set the **IsComplete** property for the item to **true** in the DataTransform rule as well.



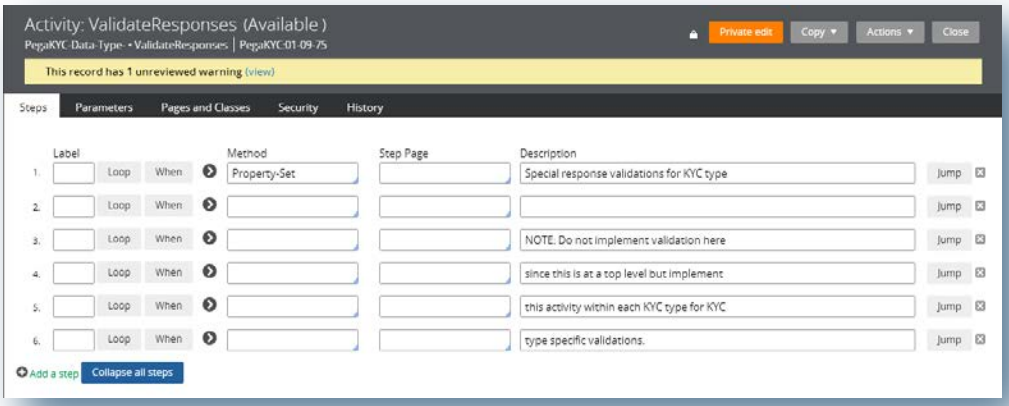
Validating KYC Type Data – Extension Point

KYC provides an extension point for performing complex item validations within a type as well as item validations across types. The extension point is provided in an activity named ValidateResponses that gets called when the case is submitted for approval in the ProcessKYCCase process flow.

Specific type validations can also be further configured by implementing this activity in a specific data class. The calling activity is ValidateResponses in the PegaKYC-Work class.



The called placeholder activity for type evaluation is stored in the PegaKYC-Data-Type- class. This activity must be saved to a specific KYC Type implementation class and configured to perform validations for responses contained in that Type.



This validation step can also be used for complex validations apart from simple local response value validations. Some of the complex validations could include:

- Validation of responses against data captured/modified at the case level
- Validation of responses against data captured/modified in external systems; i.e. data retrieved from an external system could have changed since it was retrieved by the time responses are submitted so this could include retrieval of such data and validation to make sure KYC data is in sync with external data source
- Although data captured within a type is expected to be independent of data captured in other types, there may be situations where complex cross-validations are needed. Cross-validation may involve validating responses against data captured in other types or data captured and stored in the Master Folder.

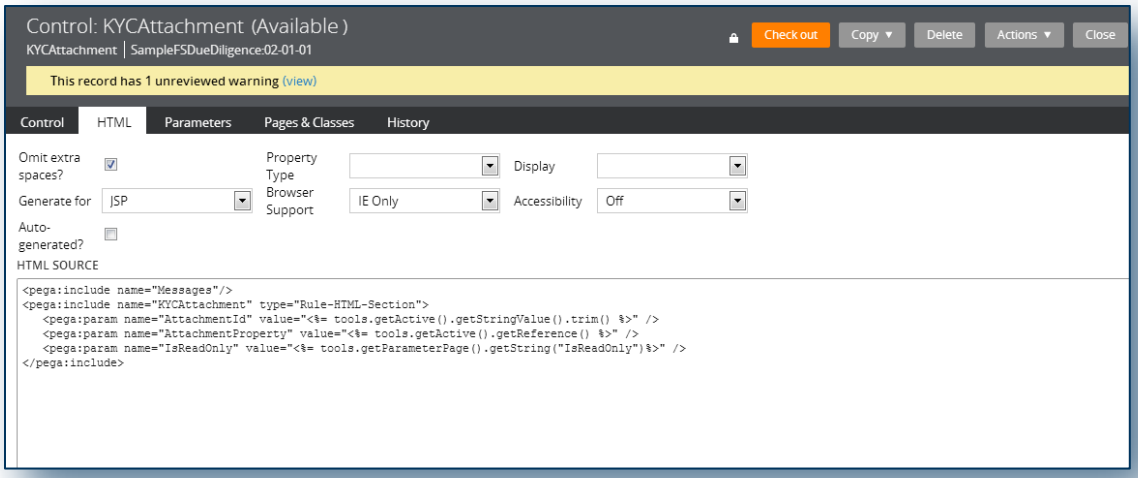
Attaching Documents via Custom Controls

KYC types and items support the capturing (attachment) of documents as KYC item responses. Typically, documents are stored in external document management systems and links to these documents will be captured as item response properties. KYC provides a set of rules for providing functionality to search and attach documents rather than search and reference documents on an external content management system.

KYCAttachment HTML Property

The KYCAttachment HTML property rule needs to be used in item response properties that are expected to hold a link to a **local** document attachment. Associating this with a response property makes the UI for that response property appear as two buttons: **Attach** and **View**.

The following example shows an HTML property implementation where it is calling an HTML section rule that contains the UI with the two buttons.



The HTML section implements the buttons where the **Attach** button calls the SelectAttachment activity and the **View** button calls the ViewAttachment activity.

Functionality provided by these activity rules in KYC makes use of document attachments kept in PRPC.

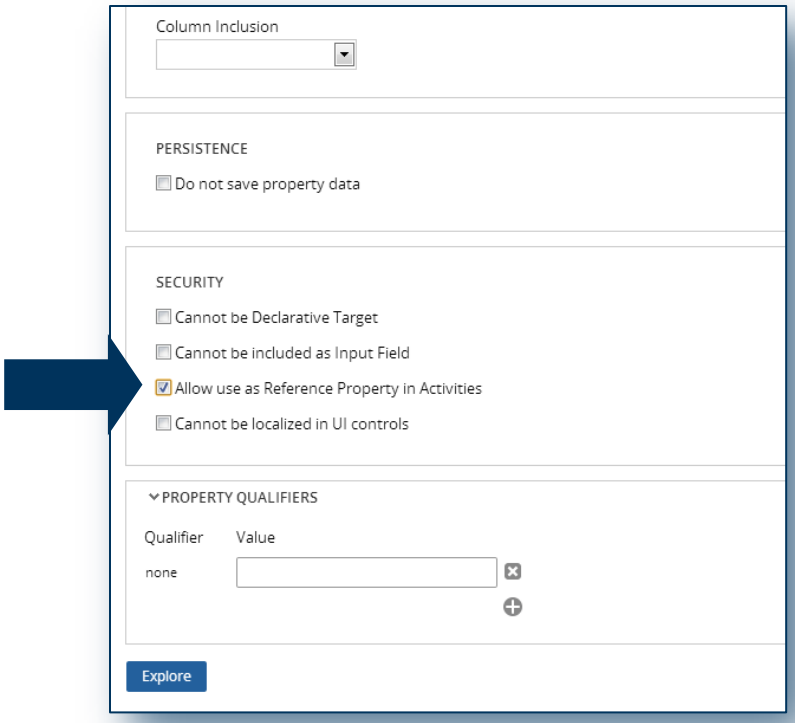
Important! For integration and management of documents in external content management systems, see *Chapter 6: Integrating With a Content Management System*.

Sharing Item Data Values Across KYC Types

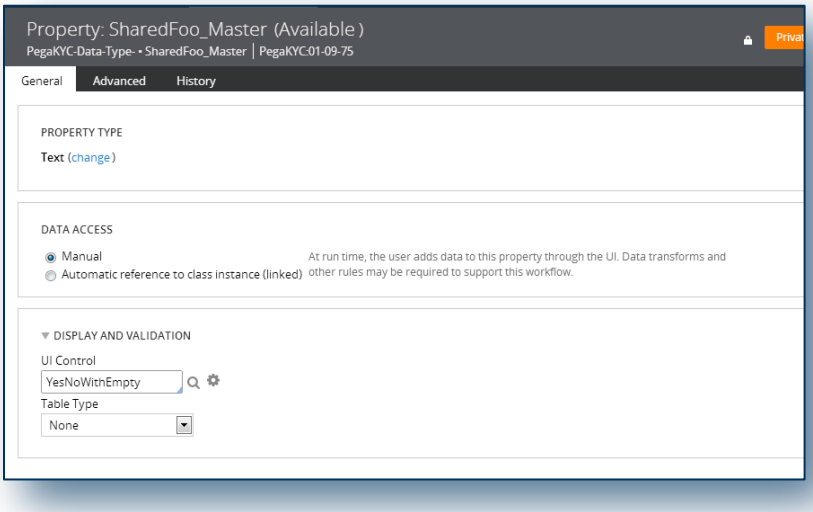
Item response properties and their data values can be shared across KYC types. To configure sharing, perform the following steps:

1. From the Designer Studio, create a property that will be used as an item response property. The property can be created in any class in the MyCo-Data-Type- hierarchy.

2. On the **Advanced** tab of the property, check the Reference Property setting.



3. Create another item property that will serve as the **Master**. Use the same name as the reference property but add the suffix _Master.



4. Add the Master that will serve as the primary item response property to the applicable KYCType rule. You can configure a description using either a Field Value or Paragraph.

5. Add the Item Reference property created in steps 1 and 2 as items to any number of other KYCType rules as required to share the Master items description and data value.

   You can configure a different description using either a Field Value or Paragraph rule but typically, the description will be configured to use the same description used by the master.

> **Important!** Do not re-use it in the same KYCType as the Master.

6. Using the Case Manager portal, create a case. Confirm that the Master appears and edits correctly in the type as configured above. Answer the question and any other required questions for that type.
7. Click on any other types for which you added the Reference property.
8. Confirm that the response value set on the Master appears in the type using the item Reference property.

> **NOTE**: By default, reference properties are read-write which means that the data values can be set in either the master or slave properties and the value will propagate across all of the other properties. If required, you can turn this off by configuring the reference/slave property with a read-only condition for the item to overwrite the read-write default.

Chapter 6: Integrating With a Content Management System

The KYC framework provides relatively seamless integration with enterprise content management systems via the Pega-Content and CMISPlus RuleSets. These RuleSets are included in the KYC application stack. Because KYC can be used with the **CMIS** (Content Management Interoperability Services) standard, any content server that supports CMIS can have plug-and-play integration with the framework.

This chapter provides instructions for integration with the Alfresco content server which supports CMIS.

Alfresco Content Management System configuration

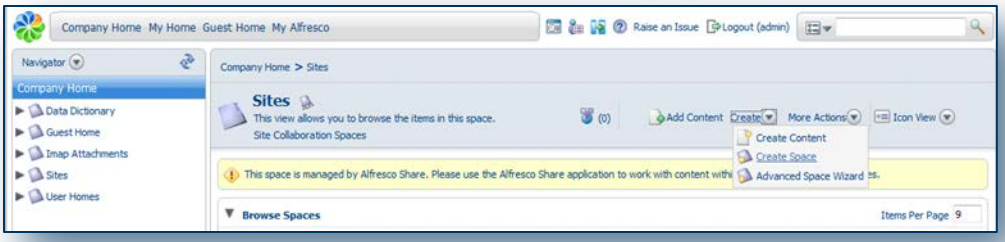
One of the popular, open-source CMIS compatible servers is the **Alfresco** content management system. To demonstrate the KYC framework's CMS integration assets, download the Alfresco software from this link:

http://wiki.alfresco.com/wiki/Download_and_Install_Alfresco

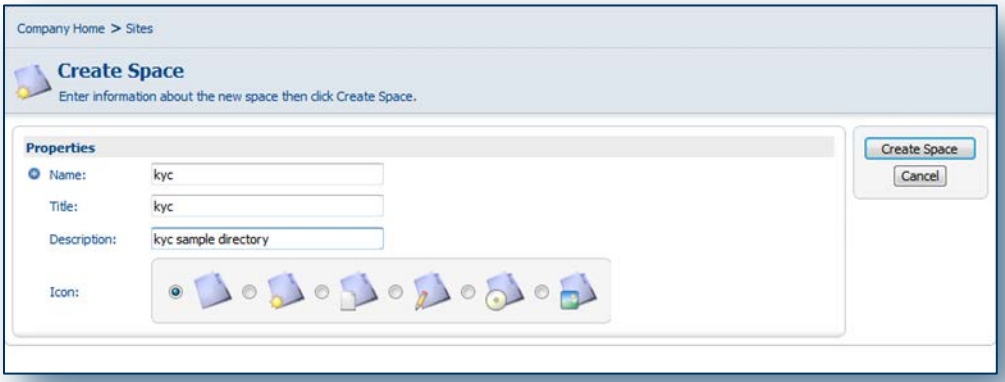
Install the system on your local host or designated server. Then, follow the instructions below.

1. Login with the username and password configured when Alfresco was installed. The default credentials are admin/admin.

2. In the navigator, go to My Home > Sites.
3. Click **Create** and select Create Space.

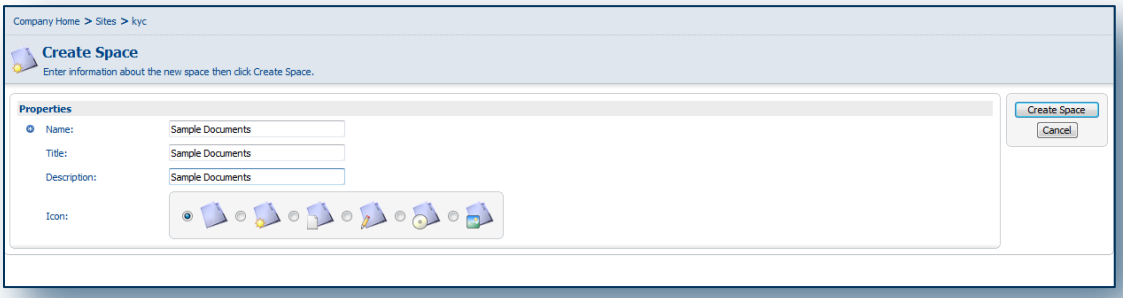


4. Name the space kyc as shown below and click **Create Space.**



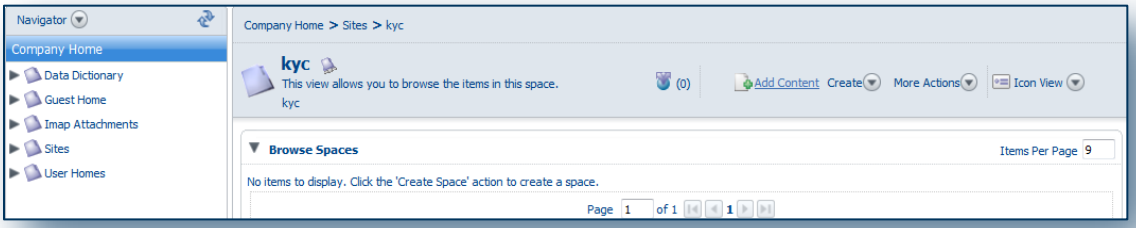
5. Next, create a sub-directory under the kyc space. Click **Create** and select Create Space.

6. Name the sub-space Sample Documents.

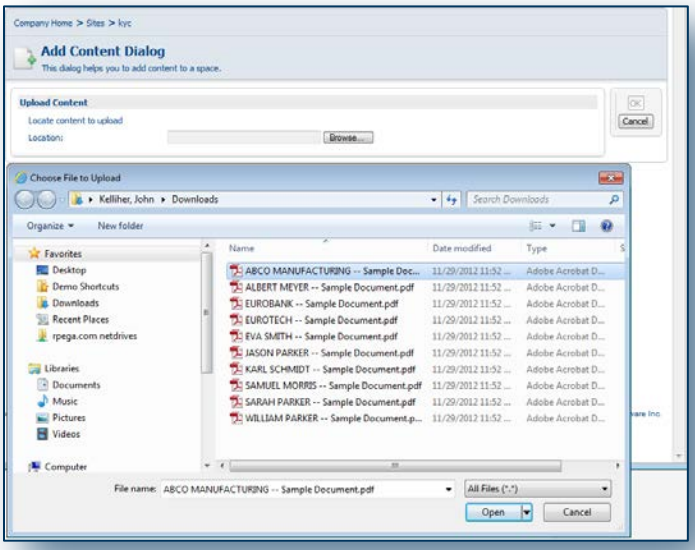


After creating the Sites / kyc / Sample Documents structure, you can upload content to support your test cases or custom demonstration.

NOTE: The KYC media comes with a set of sample documents for each sample customer that can be added as content. When performing the system demonstration using the demonstration script, these documents are used to show out-of–the-box integration assets with a CMIS compatible content server. The sample documents are located in the \Documentation\Sample CMS Documents directory on the media.



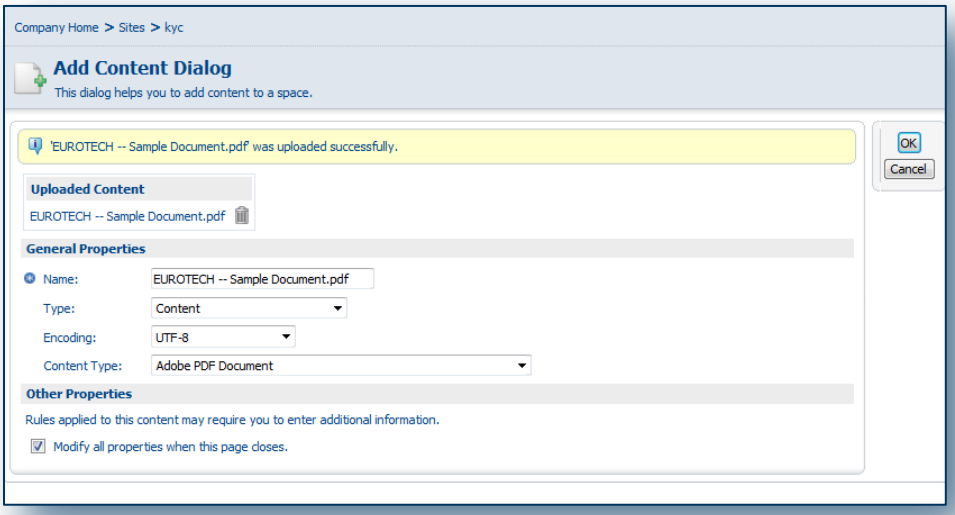
7. To upload content to the Sample Documents space, press Add Content, browse to the sample file, open it and press **OK**.



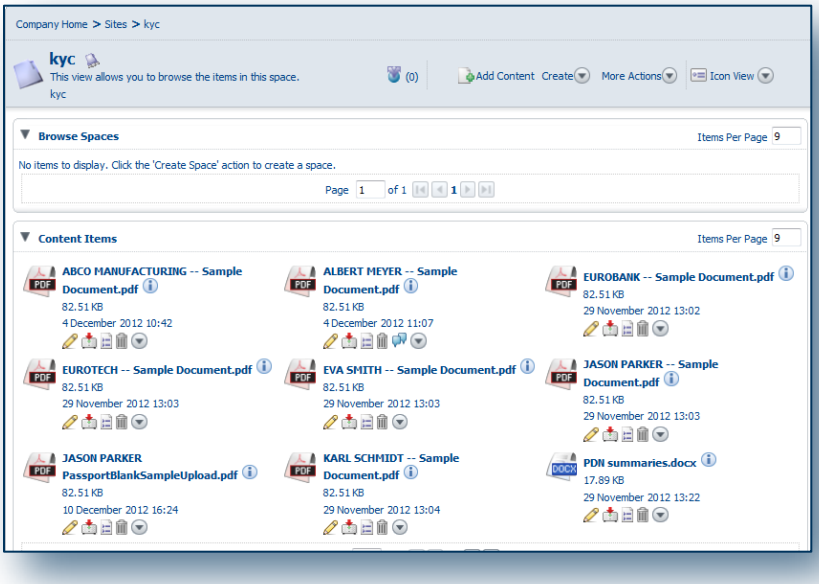
Important! If you need to upload your own custom sample files, then be sure to include the name of the party involved in the PRPC KYC case in the file name.

For example, if you are creating a case for Customer Eurotech, then you should have a file with the naming convention of EUROTECH – Sample Document.pdf or ACME – Sample Document.pdf etc. using **UPPERCASE** characters for the customer name.

8. Accept the default **General Properties** settings and click **OK.**



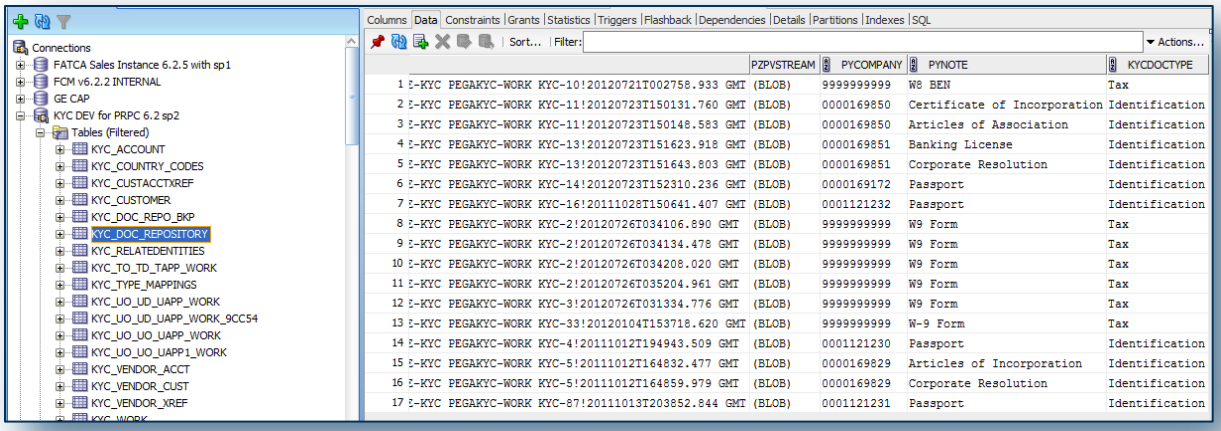
When the sample documents are uploaded to Alfresco, they will be visible in the kyc / Sample Documents directory (space).



Enabling CMS for demonstration and implementation

The KYC application default configuration sources the sample documents from the KYC_DOC_REPOSITORY table. The sample document instances are shipped with the product rule and instantiated during installation of the PegaKYC application.

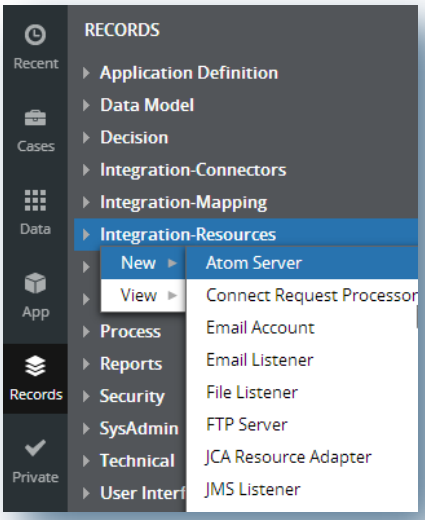
Important! Sample instances in the KYC_DOC_REPOSITORY are included in the event that a demo user does not have the time or resources to configure the Alfresco content server or they cannot access the internet to connect to an external content server.



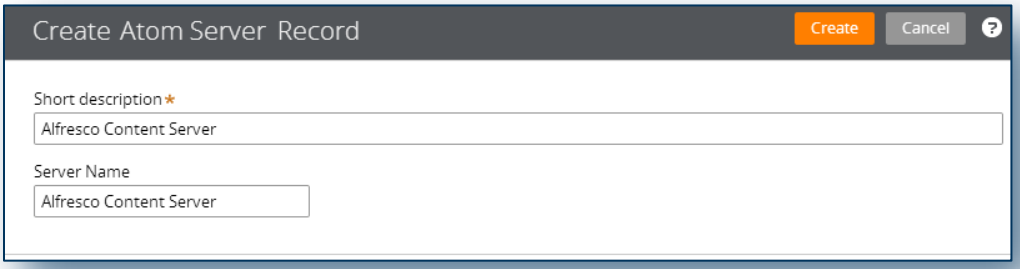
When the user installs and configures the Alfresco content server, the user must perform several steps to activate the CMIS assets.

Using these steps, users can **turn off** sourcing the out-of-the box document instances and **turn on** integration to the content server.

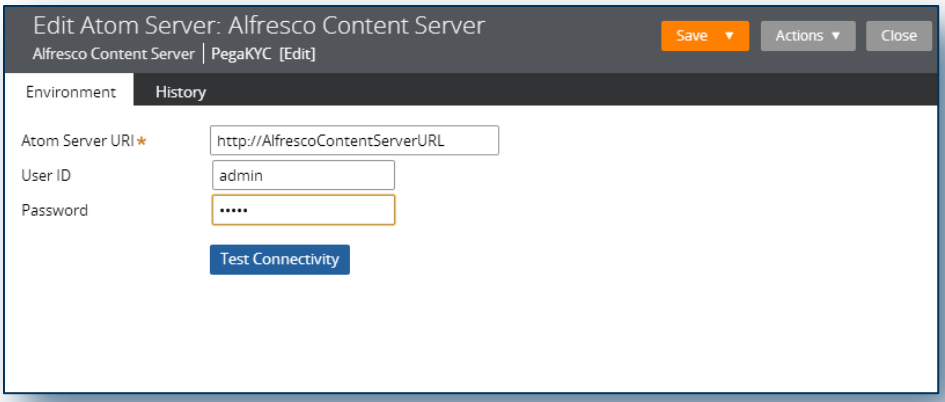
1. **Create an Atom Server rule** that references your content management server. To do this, go to Records > Integration-Resources > Right-click > New > Atom Server.



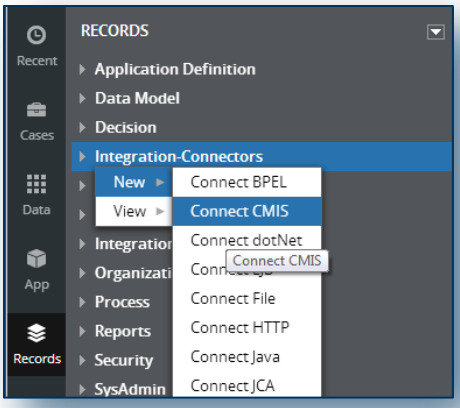
2. Enter a short description and server name and click **Create**



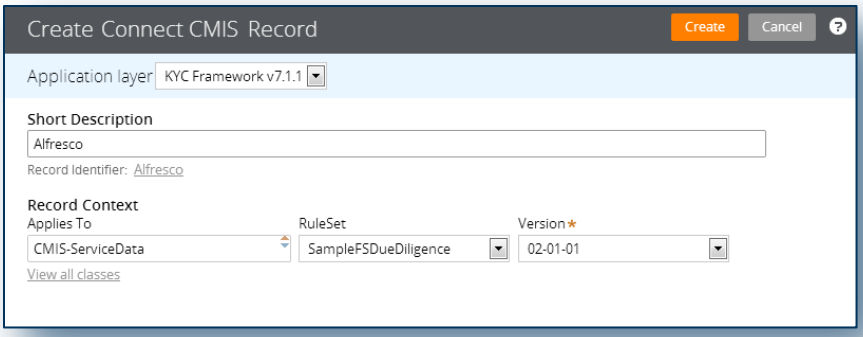
3. In the **Atom Server URL** field, enter the name of the content server. Also enter the **User ID** and **Password** for the server. Click **Save**



4. Next, **create a CMIS Connector rule** with an Endpoint matching the Atom Server rule you just created. To do this, go to Records > Integration-Connectors > Right-click > New > Connect CMIS.



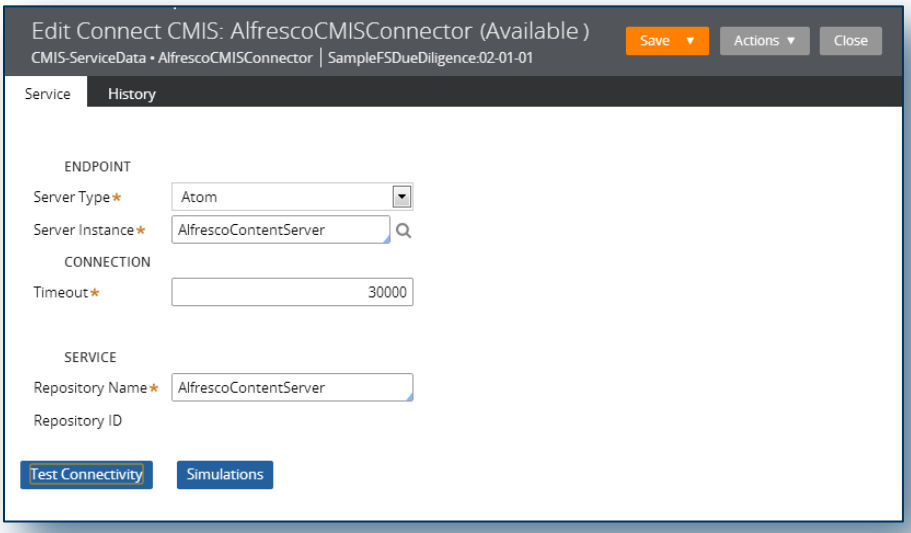
5. Enter a short description, set the Applies To class to CMIS-ServiceData and save the rule to your specialized application's RuleSet. Click **Create**



6. Set the **Server Type** to Atom and **Server Instance** to the appropriate Atom Server URL.

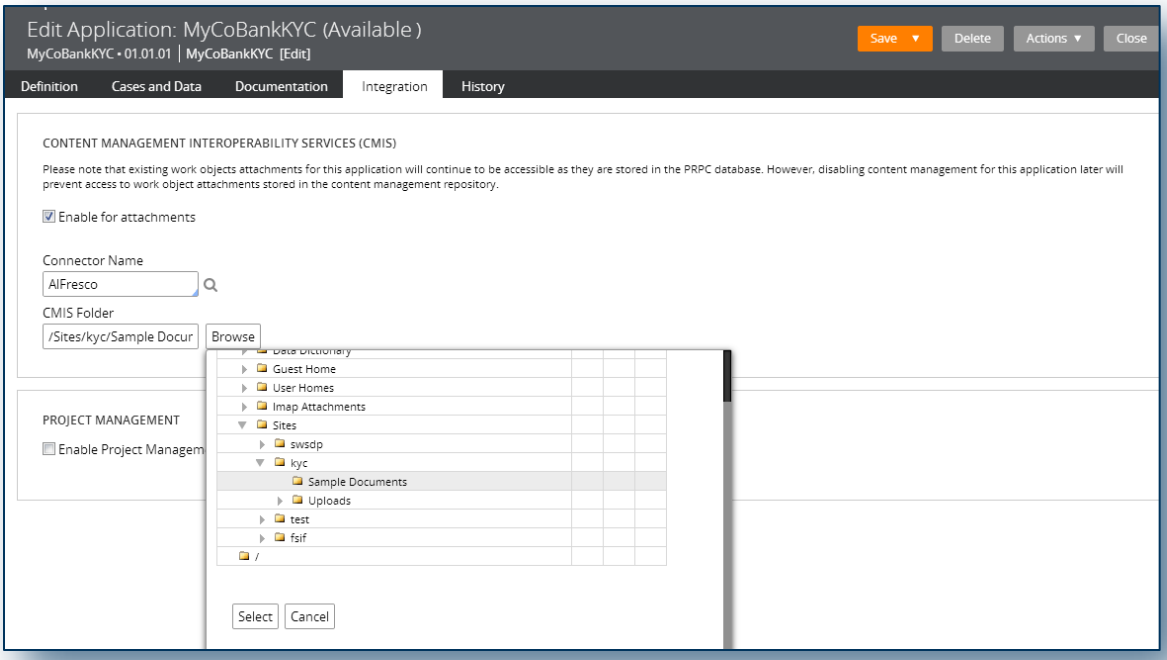
   After setting these values, the **Repository Name** (Atom Server rule name) should be available in the smart prompt. After selecting this, the **Repository ID** value will populate.

   Click **Save.**



7. Open the custom application rule —for example: MyCoBankKYC — that is built on the KYC application.

8. Select the **Integration** tab.



9. Configure the **Content Management Interoperability Services (CMIS)** sections.

   a. Check the **Enable for attachments** box.

   b. Enter the **Connector Name** rule created in the previous step

   c. Browse to the **CMIS Folder** created earlier in this document — /Sites/kyc/Sample Documents — and save the application rule.

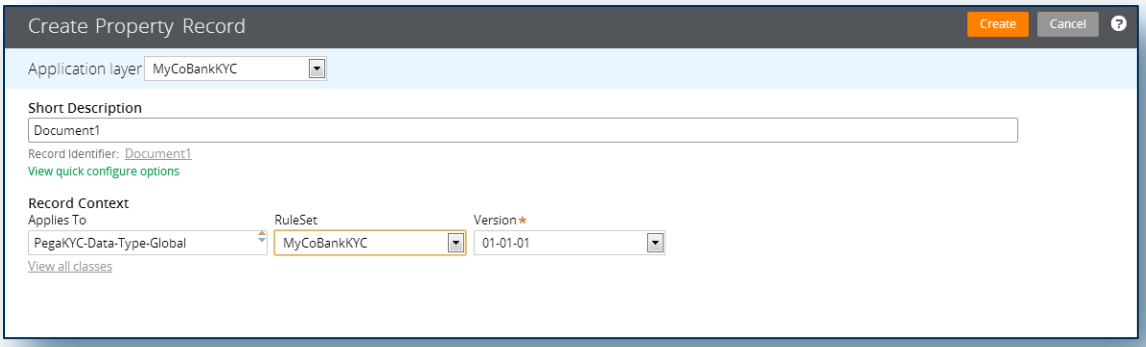
      The CMIS Folder is the default location on the Content Management instance, where attached documents will be uploaded and also the default location for searching documents already uploaded.

Important! If you change the Server Instance or the URL for the Server changes, you must reset the Server instance settings so that the Repository Name (Globally Unique Identifier) is refreshed.

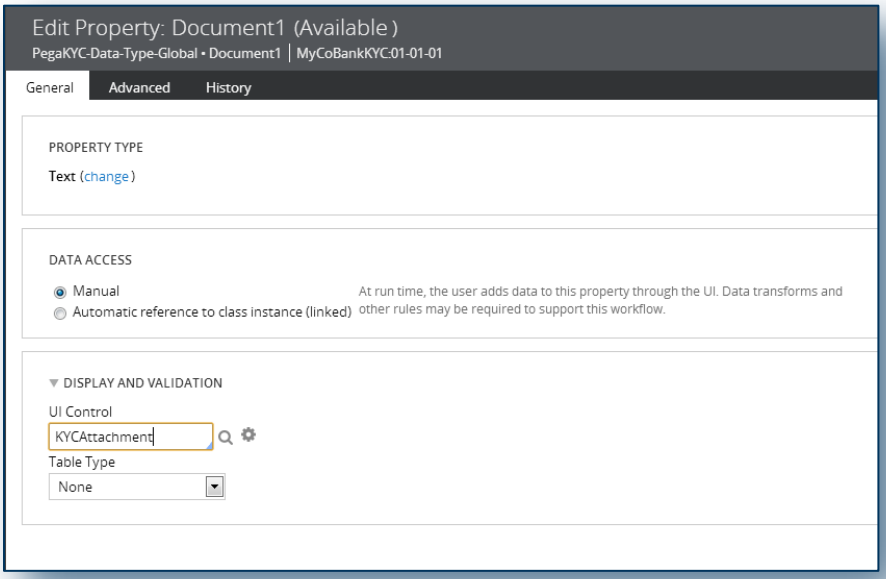
Testing Alfresco CMS Integration

After the connector rules and Alfresco environment are configured, you need to associate the KYCAttachment control with a KYC Item response property to initiate the connection to the content server. To do this, perform the following steps:

1. Create a new Item property or modify an existing Item property within the appropriate Type data class.

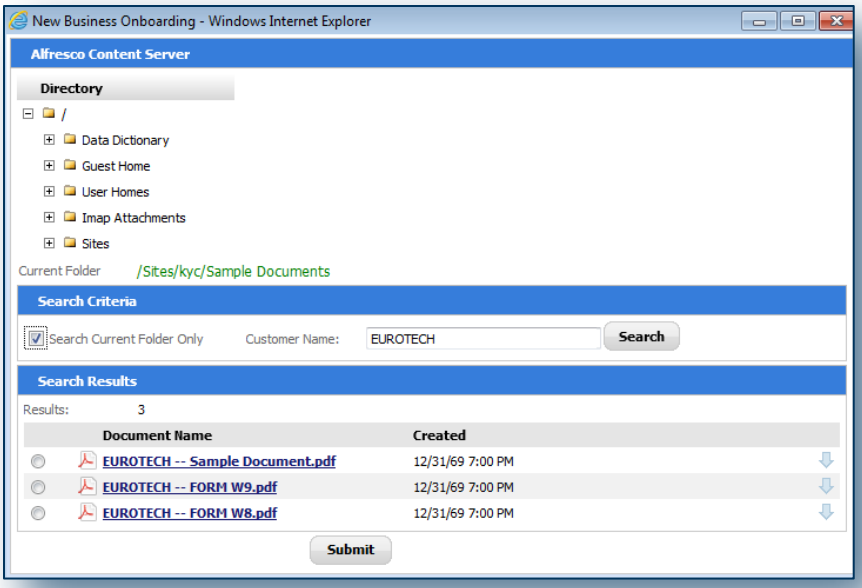


2. In the **DISPLAY AND VALIDATION** settings, reference the KYCAttachment UI control. **Save** the property.



3. To test the connection to the Alfresco server, create a KYC Case that applies the KYC Type containing the newly created Item property with the KYCAttachment control.

4. At runtime, the Item will display a **Search** button and when clicked, the system will connect to the Alfresco server and display the following pop-up dialog window



a. Note the default content directory is set to Sites/kyc/Sample Documents as configured in the Application rule.

b. By default, the system passes the customer's name to the content server to search the directory for files beginning with that name (EUROTECH).

5. Select a file from the list and click **Submit.**

6. Return to the case and note that a **View** button appears. Clicking the button prompts the user to open the document from the case or save a local copy.

Appendix A: Known Issues and Limitations

As with any software product, features occasionally do not behave as intended when the product is released. Pegasystems is committed to a high standard of quality and has implemented procedures and programs to detect and correct such issues in the product. The following is a list of issues that have not been resolved in this release that are of most interest and likely to have the most impact on the KYC user and developer community.

KYC Type Expiration – Limitation 1

Scenario

Expiry during KYC Case review/approval process

A case has been completed and is routed for approval/review. While awaiting approval, a response property item designated with an expiration property expires. For example, this could be a document. When the item expires, the KYCType that contains the item also expires. When the reviewer views the case and KYCTypes in read-only mode - outside of the Work-.Action section - the KYCType will appear as incomplete and the reviewer cannot edit the item. The reviewer would have to return the case to the submitter for re-evaluation so the item can be marked as complete. Though highly unlikely, there is a chance of this happening if the approval process gets delayed for a substantial amount of time and therefore, increasing the chances that a Type's item property may expire while awaiting review.

Solution

Configure processing so that already approved and unexpired KYCTypes at the time of submitting are not considered for expiration if pending cases exist that contain the same item and KYCType. That is, if a case is in Pending status, and a KYC Type contained within it is completed and approved, then the same KYC Type at the folder level cannot be evaluated by the Agent for expiration.

Status

An issue has been added to KYC product backlog and will be addressed in a future release of KYC. If the issue is deemed mission critical then it can be addressed during implementation at the client site and under the guidance of the product team.

KYC Type Expiration – Limitation 2

Scenario

Expired KYC Type requires completion even if a case to handle expiry exists.

a. Case with an already approved, applicable KYC Type is created for a new account/policy being opened.

b. KYC Type that is applicable for the scenario expires (which sets the expiration at the Folder level) during the evaluation and data collection process but the KYC Type of the case being evaluated will already be marked as complete.

c. Currently, the System Agent that monitors expiration of KYCTypes within the Master Folder and automatically creates another case for re-evaluation of the expired KYC Type.

d. Although a new KYC case got created for the expired KYC Type, the initial case's KYC Type cannot be considered complete until responses for the expired KYC Type are provided (i.e. user will not be able to send the first case for review without providing responses for the expired KYC Type).

Solution

Configure the Agent to search for open cases containing the same KYC Type that has expired. If a case is found, then the Agent should avoid creating a new case for re-evaluation of the expired KYC Type. The open case(s) with the expired KYC Type should be marked as incomplete and force the user to correct the item of the already open case. Alternately, when a case is submitted for approval, a check can be run against the Master Folder to see if any of the KYC Types have been marked as invalid. If yes, then the user cannot submit the case until the expired item(s) are addressed. Bottom line – the system should not automatically create cases for expired items if there are already open cases containing the same, expired KYC Type and items.

Status

An issue has been added to KYC product backlog and will be addressed in a future release of KYC. If the issue is deemed mission critical then it can be addressed during implementation at the client site and under the guidance of the product team.

KYC Type Expiration – Limitation 3

Scenario

The Expiry Agent will not re-evaluate a case's expired KYC Type if the type has been declined and therefore, not synched with Master Folder.

a. Expiry processing agent automatically creates a case for handling expired KYC Type(s).

b. Case is completed and sent for approval.

c. Case gets declined and is resolved The KYCType is not synchronized with the Master Folder.

d. Agent will not create another case since it was already processed.

Solution

Introduce a new indicator to the case for handling expired cases. The indicator would be used to reset the Expiry Processing indicator in the Master Folder when a case is resolved as rejected / declined so that the agent could create a new case as necessary.

Status

An issue has been added to KYC product backlog and will be addressed in a future release of KYC. If the issue is deemed mission critical then it can be addressed during implementation at the client site and under the guidance of the product team.

KYC Type already approved via different case

Scenario

a. Simultaneously created cases 1 and 2 both contain KYC Type A.

b. Case 20 is approved first while case 22 is still in the data capture process.

c. At the time of submitting case 22, KYC Type A is already considered approved but the new case will have KYC Type A go through approval process again.

Solution

In order to avoid this, the system will need to include synchronization with the Master Folder using a pre-processing utility within the CompleteProcess flow action. The system should be able to retrieve applicable and approved KYC Types from the Master Folder so that it would not require re-approval. That is, it should automatically be marked as **Complete** when it gets to the review level.

Status

An issue has been added to KYC product backlog and will be addressed in a future release of KYC. If the issue is deemed mission critical then it can be addressed during implementation at the client site and under the guidance of the product team.

KYC Type Rule modifications

Scenario

a. KYC Type B rule is included in a case currently in KYC data capture assignment.

b. KYC Type B rule is modified; item is added or removed.

c. At the time of submitting the case with KYC Type B, it does not conform to the most current KYC Type specification.

Solutions

1. In order to avoid this include sync with Master Folder in pre-processing of the flow action to detect KYC Type rules being updated (considering KYC Item / property rules will further increase complexity) and allow the user refresh KYC Type via a local action.

2. Use date circumstancing for modified KYCType rules. Modification to KYC type rules must be strictly controlled and audited according to change management procedures.

3. Enforce completion of all KYC cases containing the Type rule before any changes can be made to the items

Status

An issue has been added to KYC product backlog and will be addressed in a future release of KYC. If the issue is deemed mission critical then it can be addressed during implementation at the client site and under the guidance of the product team.