Pega Platform '24.2

3 September 2025

©2025 Pegasystems Inc

CONTENTS

Insights and Reporting 7
Insights 10
Explore Data 11
Customizing Explore Data 14
Specifying objects in Explore Data 15
Enabling Explore Data in UI Kit 19
Creating Insights 21
Generating Insights by using Pega GenAI 23
Creating Insights from UI authoring 25
Managing Insights 27
Customizing table-based Insights 28
Customizing chart-based Insights 33
Managing data series in Insights 41
Color-coding the appearance of data series 44
Benchmarking data with reference points 45
Grouping data in Insights 47
Grouping information by dates 49
Grouping Insights information into custom categories 50
Adding custom numeric groups 52
Creating simple value charts 56
Viewing underlying records of Insights 59
Applying promoted filters 61
Managing the visibility of Insights 65
Adding Insights to your application 66
Scheduling Insights 69
Managing scheduled Insights 72
Troubleshooting Insights issues 74
Login prompt appears too often 74
Removing the thousands separator from columns with numbers 75

©2025 Pegasystems Inc

Renaming Explore Data objects. 76
Troubleshooting missing columns in Explore Data. 77
Troubleshooting missing data objects in Explore Data. 78
Troubleshooting Portal issues. 80
Insights and Reporting versus external tools. 82
Dashboards. 84
Creating dashboards. 86
Traditional reports. 89
Report Browser. 90
Report types. 92
Report categories. 94
Creating a report category in Report Browser. 96
Creating a report. 97
Best practices for creating reports. 99
Adding columns to reports. 110
Sorting and grouping columns in list reports. 112
Duplicating existing reports. 114
Running a report. 115
Report Viewer. 116
Reports run on schedule. 118
Scheduling reports. 119
Managing report subscriptions. 122
Subscribing to reports. 123
Testing scheduled reporting tasks. 124
Updating scheduled reporting tasks. 124
Removing scheduled reporting tasks. 125
Editing a report. 126
Report Editor. 130
Data Explorer. 131
Modifying list reports. 133
Modifying summarized reports. 136
Adding filter conditions. 137

PEGA ©2025 Pegasystems Inc

Editing filter conditions. 141
Editing filter logic. 143
Managing charts in Report Editor. 145
Removing charts in Report Editor. 148
Converting a list report to a summarized report. 149
Converting a summarized report to a list report. 150
Creating advanced reports. 150
Report definitions. 152
Managing filter options in Report Viewer. 154
Charts in reports. 156
Defining associations for map charts. 161
Adding maps to reports. 162
Adding charts in the Report Definition rule form. 163
Removing charts in the Report Definition rule form. 165
Enabling users to view detailed chart data. 166
Exposing detailed chart data. 167
Styling charts in the application skin. 167
Associations. 170
Creating simple associations between classes. 171
Creating advanced associations between classes. 172
Referencing association rules in a report. 174
Viewing associations that are available for reporting. 175
Creating Class joins in reports. 175
Using join filters to combine data in reports. 177
Using aggregation to combine data in reports. 180
Reporting landing page. 181
Setting resource limits for reporting. 183
Including report metadata in Excel reports. 184
Excluding items from Report Browser. 185
Creating an advanced report category. 186
Managing report categories. 187
Creating a report shortcut. 188

©2025 Pegasystems Inc

Managing reports shortcuts. 190
Reviewing usage report statistics. 192
Creating usage reports. 193
Customizing usage reports. 193
Specifying a data source for reports. 194
Disabling usage of a reports database. 195
Data extraction. 196
BIX extraction overview. 197
Enabling the BIX Ruleset. 201
Upgrading BIX. 202
Extract Rules overview. 203
Creating and running an Extract rule. 207
Extracting data in batch mode. 212
Extracting data by using secondary database. 223
More about Extract rules. 224
Configuring BIX real-time data extraction. 226
Using the Job Scheduler rule to extract data with BIX. 230
Running an Extract rule by using an agent. 233
Unique run identifier. 235
Overview of BIX extractions in Pega Cloud environments. 236
Procedures for BIX extractions in Pega Cloud environments. 240
Configuring the Pega Cloud SFTP service to access BIX extract files. 241
Configuring an FTP listener to access BIX data extracts in Pega Cloud environments. 242
Configuring an activity to access BIX extract files. 244
Troubleshooting BIX in Pega Cloud environments. 250
Extracting data to custom output repositories. 262
Overview of BIX extractions in on-premises systems. 264
Running a BIX command-line extraction in on-premises systems. 276
Configuring the source database for BIX command-line extractions in on-premises systems. 278

©2025 Pegasystems Inc

Configuring the target database for BIX command-line extractions in on-premises systems. 286
Configuring optional prconfig.xml settings in on-premises systems. 288
Configuring BIX logs for command-line extractions in on-premises systems. 290
Enabling password encryption for BIX command-line extractions in on-premises systems. 291
Running a BIX command-line extraction in on-premises systems. 293
Generating a DDL file for command-line Database Schema extractions 298
Running a BIX command-line extraction using ANT build files in on-premises systems. 299
BIX logging. 302
Viewing and configuring BIX logs. 303
BIX error handling. 304
Optimizing BIX data extraction performance. 306
Extracting large data by using BIX. 309
Business Intelligence Exchange - FAQ. 312
Troubleshooting BIX for on-premises environments. 317

©2025 Pegasystems Inc

Insights and Reporting

Pega Platform™ provides a convenient way to retrieve specific application data and present the information in the form of comprehensive reports or appealing Insights with charts. For example, in a sales application, you can create a report about bestselling products to analyze current trends and identify potential areas for improvements.

The reporting functionality provides statistics that you can analyze to enhance your business and make data-driven decisions. Pega Platform offers the following efficient ways to gather application data for interpretive purposes: Insights and traditional reports.

Insights

Insights offer a wide range of customizable options thanks to which you can, for example, present data as interactive charts that you can then drill down to retrieve more specific information. By using a range of available options, you can create unique Insights that meet specific business requirements, and then conveniently share them with specific groups or individuals.

Insights are accessible in the Explore Data functionality, which can use either Constellation or Theme Cosmos environments, thanks to which you can add Insights to the landing pages of applications. In addition, you can enable Explore Data for UI Kit applications.

For more information, see Insights, Explore Data and Enabling Explore Data in UI Kit.

Pega Platform uses generative artificial intelligence (Pega GenAI™) that helps you create comprehensive Insights that match your advanced business requirements.

The following video shows generating AI-powered Insight in a sample sales application at run time. In this example, the first query is selected from the list of prompts

©2025 Pegasystems Inc

generated by Pega GenAI and shows average sales and profits across states. The second query, is entered manually and showcases 5 bestselling subcategories of products.

Traditional reports

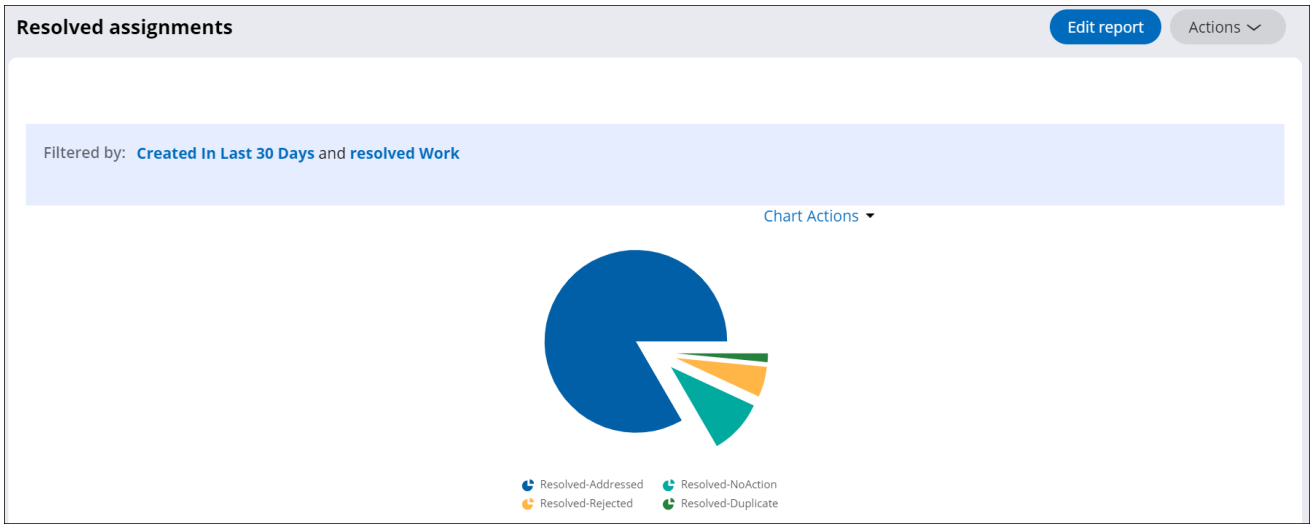
You can present data as extensive list reports that provide detailed information in columns, or as summarized reports that provide statistics. The latter report type is customizable with static charts that you can use to exemplify certain data.

Each report is editable with various options such as sorting, grouping or filtering, so you can adjust the content to the constantly changing business requirements. For example, in a retail application, in a list report about orders, you can filter the results by category, and then sort them from highest to lowest to analyze selling trends. For more information, see Traditional reports .

The following figure presents a sample summarized report about assignments that have been resolved within the last 30 days. The report includes a pie chart to help users conveniently analyze the results.

> **NOTE:** As opposed to Insights, charts in summarized reports are static, so you cannot drill down to subsequent data series.

©2025 Pegasystems Inc



Summarized report with chart

Data extraction

You can integrate data from Pega Platform with data from other systems of records by using Pega Business Intelligence Exchange™. In this way, you create a broad and comprehensive view of your business.

BIX extracts data in a format suitable for use in popular business intelligence applications, such as data warehouses. The extracted data is available in various formats, including XML, CSV, and database schema. With BIX, you can extract large amounts of data from Pega Platform, which your organization can then analyze by using data mining and statistical techniques. Such an approach can help you identify trends, patterns, and areas for improvement.

BIX is a Ruleset that you can run manually from the Ruleform, automatically by using a Job Scheduler, or from the command line (on-premises only).

For more information, see Data extraction.

- **Insights**
- **Traditional reports**
- **Data extraction**

©2025 Pegasystems Inc

Insights

Insights help you conveniently explore and analyze application data, such as company sales, orders, and team assignments. You can use Insights to retrieve specific data, and present the data as a comprehensive table or an appealing chart.

For example, in a sales application, you can visualize data about profits that your company generates in specific regions, and then analyze the distribution of earnings across various categories of products.

Pega Platform uses Insights to transform data queries into tables or visualizations that you can then share between users and embed in landing pages of your applications. Insights offer a wide range of customizable options, thanks to which you can filter, sort, and group multiple data items to achieve the expected results according to your business needs. Additionally, you can visualize the application data to present meaningful information as appealing charts. In this way, you increase legibility and enhance the process of data analysis.

> **NOTE:** Pega Platform includes numerous out-of-the-box Insights that source data from your application, supporting preliminary data analysis. The available options include, for example, the `Number of work items created daily` Insight, which tracks and displays the daily creation rate of work items, helping you monitor workload trends and identify peak activity periods.

The following figure shows a custom chart-based Insight which is tailored to show specific sales information. In this example, products with sales lower than 15,000 are marked in red, and the rest are marked in blue. The chart also features a green reference line that indicates a satisfactory level of sales. By using the available options, you can modify Insights to meet strict business criteria.

©2025 Pegasystems Inc

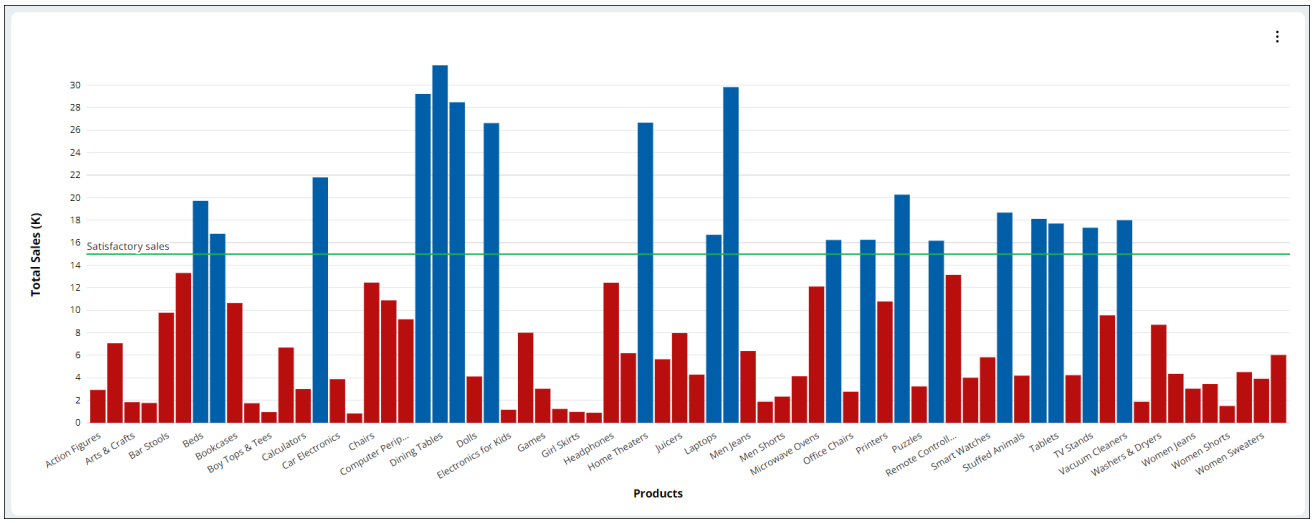


Chart-based Insight for sales

In turn, the following video shows generating AI-powered Insight in a sample sales application at run time. In this example the first query is selected from the list of prompts generated by Pega GenAI™ and shows average sales and profits across states. The second query, is entered manually and showcases 5 bestselling subcategories of products.

- **Explore Data**
- **Creating Insights**
- **Managing Insights**
- **Troubleshooting Insights issues**
- **Insights and Reporting versus external tools**
- **Dashboards**

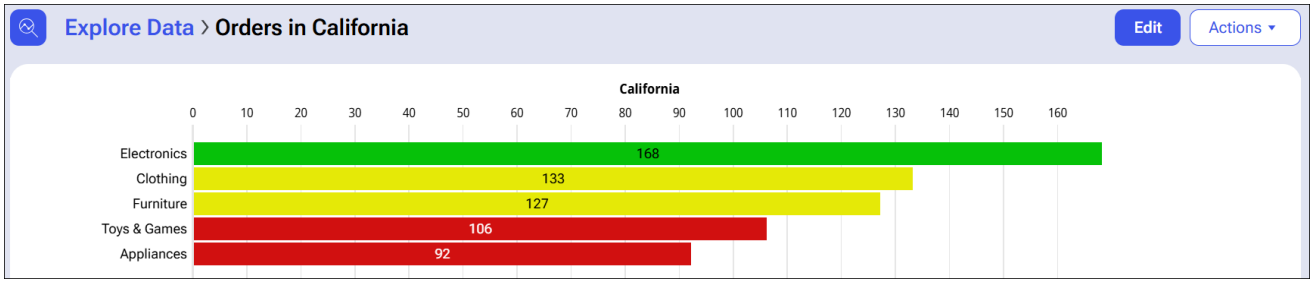
Explore Data

The Explore Data feature provides access to your application data, making it more convenient to present information in a clear and legible way. With Explore Data, you

©2025 Pegasystems Inc

can save time and streamline data analysis by saving the results as appealing chart-based Insights, or comprehensive table-based Insights.

For example, in a sales application, you can use Explore Data to create an Insight with granular information about best-selling articles in California. In this way, you can analyze trends and identify customers' preferences, as shown in the following figure:



Insight with orders placed in California

Obtaining Explore Data

To use Explore Data in your application, perform the following actions:

1. Enable Constellation-based landing pages to use Explore Data.

   For more information, see Constellation in traditional UI and Constellation UI.

2. Add Explore Data to the navigation menu of your application.

   For more information, see Organizing the main navigation for a standard work Portal.

Overview of Explore Data

Explore Data enables you to perform the following actions:

- Gain access to your application data, such as detailed information about assignments, sales, or customer data.
- View and customize all saved Insights, which you can use throughout the application.
- Filter, sort, or group data to view only the specific information that matches your business needs.

©2025 Pegasystems Inc

- Create comprehensive table-based Insights that show your data in legible tables.

  If you want to show properties in Explore Data, ensure that the properties are optimized for reporting and marked as relevant records.

  You can create associations between classes, and add them to your Insights.

  For more information, see Associations.

  > **NOTE:** To enable associations for Explore Data, manage the *pyUseAllAssocationsForDataSet* When Rule, which is turned off by default.
  > The following covers details about associations for run-time Explore Data:
  > - Associations referenced in the Explore Data landing page configuration are used.
  > - If the Explore Data landing page configuration is not used and the *pyUseAllAssocationsForDataSet* When Rule is set to *true*, all available associations (both simple and advanced) are displayed. Otherwise, the system displays simple associations only.
  >
  > The following covers details about associations for design-time Explore Data:
  > - If the *pyUseAllAssocationsForDataSet* When Rule is set to *true*, all available associations (both simple and advanced) are displayed. Otherwise, the system displays simple associations only.

- Create interactive chart-based Insights to conveniently analyze data from your application.

  You can choose from several chart types such as pie, bar, donut, or column.

- Schedule your Insight to run on a specific date, and share it with other users.

The following video shows how Explore Data feature works at run time. In this example, the sales data is grouped by the state of California, then filtered by the Arts & Crafts

©2025 Pegasystems Inc

product category, and finally sorted by profits to retrieve granular information from the application:

- **Customizing Explore Data**
- **Specifying objects in Explore Data**
- **Enabling Explore Data in UI Kit**

Customizing Explore Data

Enhance your user experience with Explore Data by creating custom views, which might feature diverse types of data, allowing you to adapt to the current business needs. In this way, you can save time and streamline the process of data analysis.

For example, in a sales application, you can create a view that groups orders by date to conveniently retrieve the required information from specific time period. In this way, you save time and can effortlessly navigate between various sets of data.

1. In the navigation pane of your application, click **Explore Data**.
2. Customize the appearance of Explore Data according to your needs.

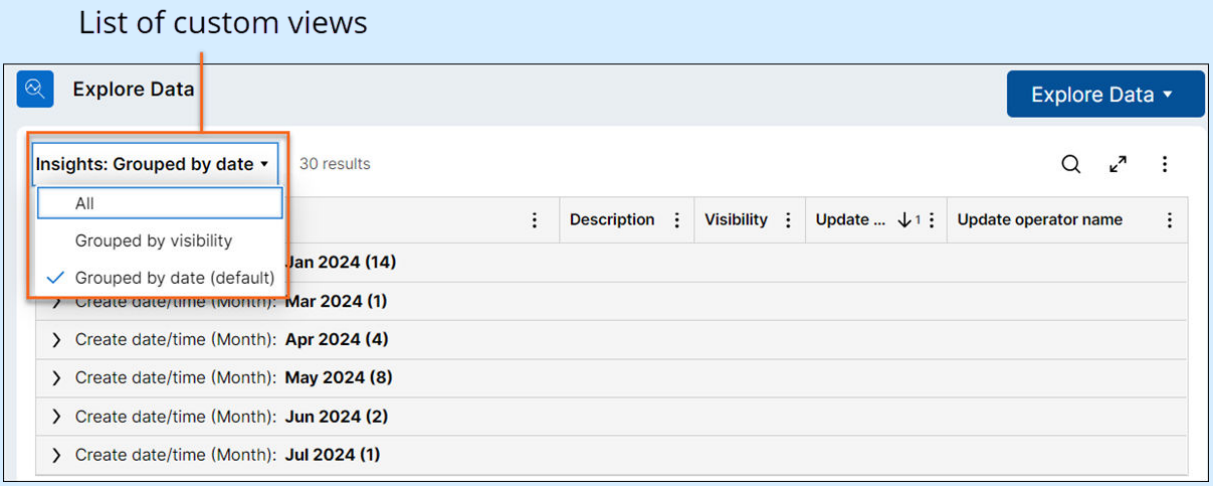
   > **For example:**
   >
   > In the upper-right corner of Explore Data, click **Actions** > **Group**, and then specify how you want to group data in your custom view.

3. Save the view by performing the following actions:
   a. In the upper-right corner of Explore Data, click **Actions** > **Save view as**.
   b. In the **Save view as** dialog box, in the **View name** field, enter a unique name for the view.
   c. **Optional:** To make the view default, select the **Mark as default?** checkbox.
   d. Click **Submit**.

©2025 Pegasystems Inc

Result:

You can now select the new view from the list at the top of Explore Data. The following figure shows Explore Data with two custom views, grouping the data by visibility and date.



Custom views in Explore Data

Specifying objects in Explore Data

Emphasize the importance of specific objects (classes) by creating a custom drop-down list in Explore Data or by controlling which objects are displayed on the Explore Data landing page. You can also specify custom associations that appear during authoring of Insights.

For example, in a sales application, you can customize the drop-down list to feature objects (classes) that include data about orders placed by customers. In this way, you ensure that users focus on relevant objects when using Explore Data.

NOTE: Even if you do not define any associations for a specific object, the system adds the *pyCaseTypeInformation*, *pxCreateOperator*, and *pxUpdateOperator* associations to Insights by default.

©2025 Pegasystems Inc

Before you begin:

Explore Data uses Data Pages to retrieve data from your application. To ensure that your Data Page is valid to use by Explore Data, ensure that it meets the following criteria:

- List structure.
- No parameters.
- Uses report definition as a source.
- Has the **Allow querying any column** checkbox selected.
- Has no response Data Transform.

> **NOTE:** If multiple Data Pages from the same class are valid, then the runtime portal displays only one Data Page per class. The system selects the Data Page that matches the pattern *D_ClassSuffixList* such as *D_EmployeeList* for the class *Org-App-Work-Employee*.

For more information, see Elements of a Data Page.

1. In the navigation pane of App Studio, click **Channels**.
2. In the **Current channel interfaces** section, click the channel that you want to update.
3. On the **Content** tab, click **Landing pages**, and then, in the list of landing pages, select **Explore Data**.
4. In the **Edit landing page** pane, manage the objects (classes):

Choices	Actions
Add objects (classes)	a. In the **Objects** section, click **Add**.

©2025 Pegasystems Inc

Choices	Actions
NOTE: If the Objects list is empty, the Explore Data menu shows all the valid Data Pages.	b. In the list of objects, select the objects (classes) that you want to feature in **Explore Data**. c. Click Add. **NOTE:** The selected objects (classes) appear in the Explore Data drop-down list and on the Explore data landing page (if at least one Insight is created).
Hide objects (classes)	In the **Objects** section, next to the object (Class) that you want to hide, click the Remove icon.
Specify associations	a. In the **Objects** section, next to the objects (classes) for which you want to add associations, click the Configure icon. b. In the **Edit object** dialog box, in the **Associations** section, click Add. c. In the list that appears, select objects (classes) that you want to associate, and then click Add. d. Click Save.

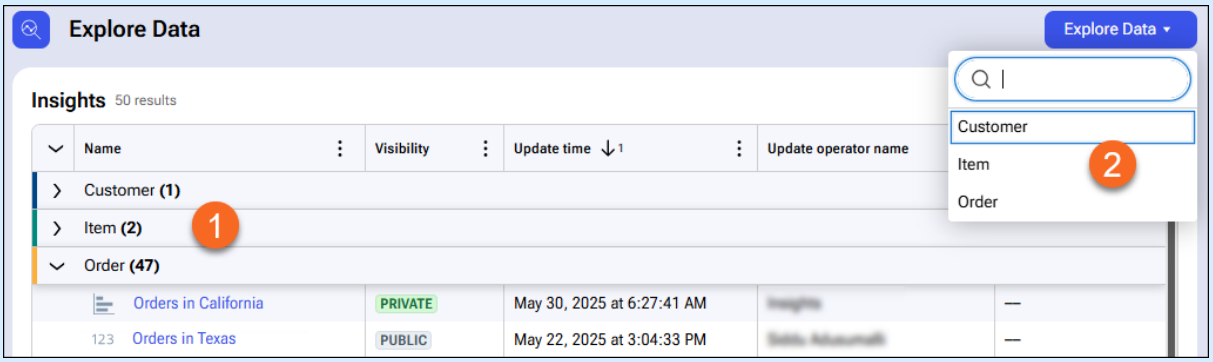
5. In the upper-right corner of the form, click Save.

©2025 Pegasystems Inc

For example:

The following figure shows:

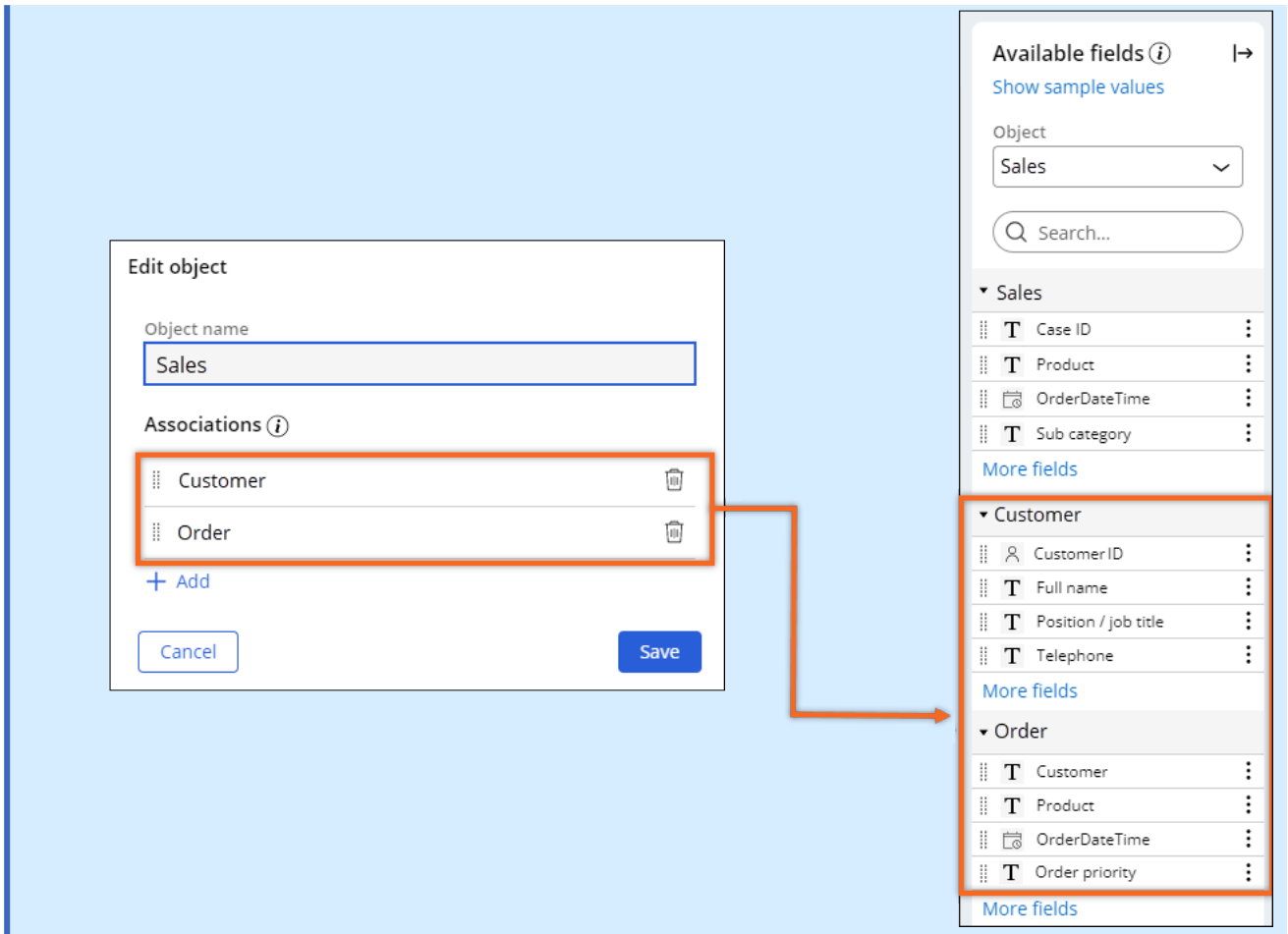
1. Explore Data landing page with custom Data Pages.
2. Explore Data drop-down list with custom Data Pages.



Custom Data Pages in Explore Data

In turn, the following figure shows the customization of the **Sales** object with two specified associations on the left, and the outcome of the customization visible when authoring Insights on the right:

©2025 Pegasystems Inc



Associations in Insights

Enabling Explore Data in UI Kit

Enrich your UI Kit applications with the Explore Data tool, which provides a convenient way to gather and analyze data from your applications. With this enhancement, you can retrieve specific business information and display it in visually appealing charts or comprehensive tables to increase legibility and viewer engagement.

For example, in a sales application, you can use Explore Data to visualize sales data from the United States and help viewers identify top-selling products across the country.

Enabling Constellation UI

©2025 Pegasystems Inc

1. In Dev Studio, enable Constellation UI to render landing pages.
   For more information, see Enabling Constellation components in traditional UI.

Adding the Explore Data tool to your application

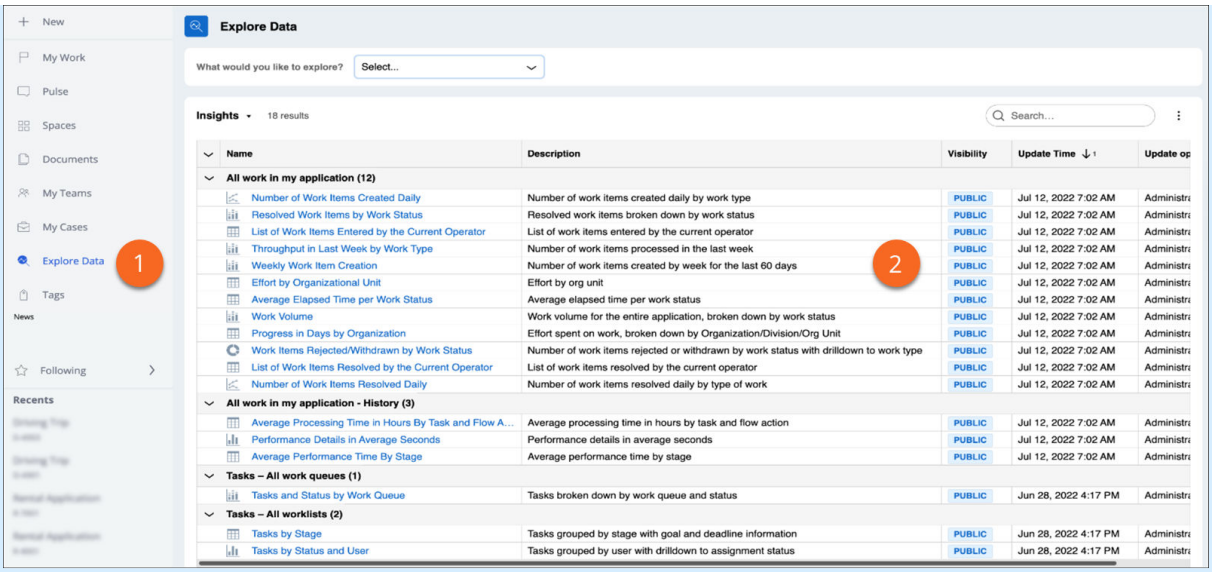
2. In the header of Dev Studio, click the name of the application, and then click Channels and interfaces.
3. In the **Current channel interfaces** section, click the Channel that you want to edit.
4. In the **Navigation** section, add the Explore Data tool:
   a. In the **Navigation menu** section, click Add item > Custom pages > Explore Data
   b. In the upper-right corner of the Rule, click Save.
5. Sign out from your application, and then sign in again for the changes to take effect.

Result:

The following sample figure shows how Explore Data looks at run time for UI Kit applications:
The figure features:

1. Navigation pane with Explore Data.
2. Explore Data with the list of Insights.

©2025 Pegasystems Inc



Explore Data in UI Kit

Creating Insights

Collect specific information from your application, and save the results in a clear visual form to streamline the process of data analysis. In this way, you can showcase meaningful information to your audience in an engaging manner, allowing them to effortlessly understand complex data.

For example, in a sales application, you can create a chart-based Insight with data about orders to present a company's sales results within the last year. In this way, you can identify best-selling products and analyze the revenue.

Before you begin:

Add the Explore Data page to your application. For more information, see Organizing the main navigation for a standard work Portal.

NOTE: You can add the Explore Data page only if your application uses Constellation or a traditional UI architecture hybrid mode.

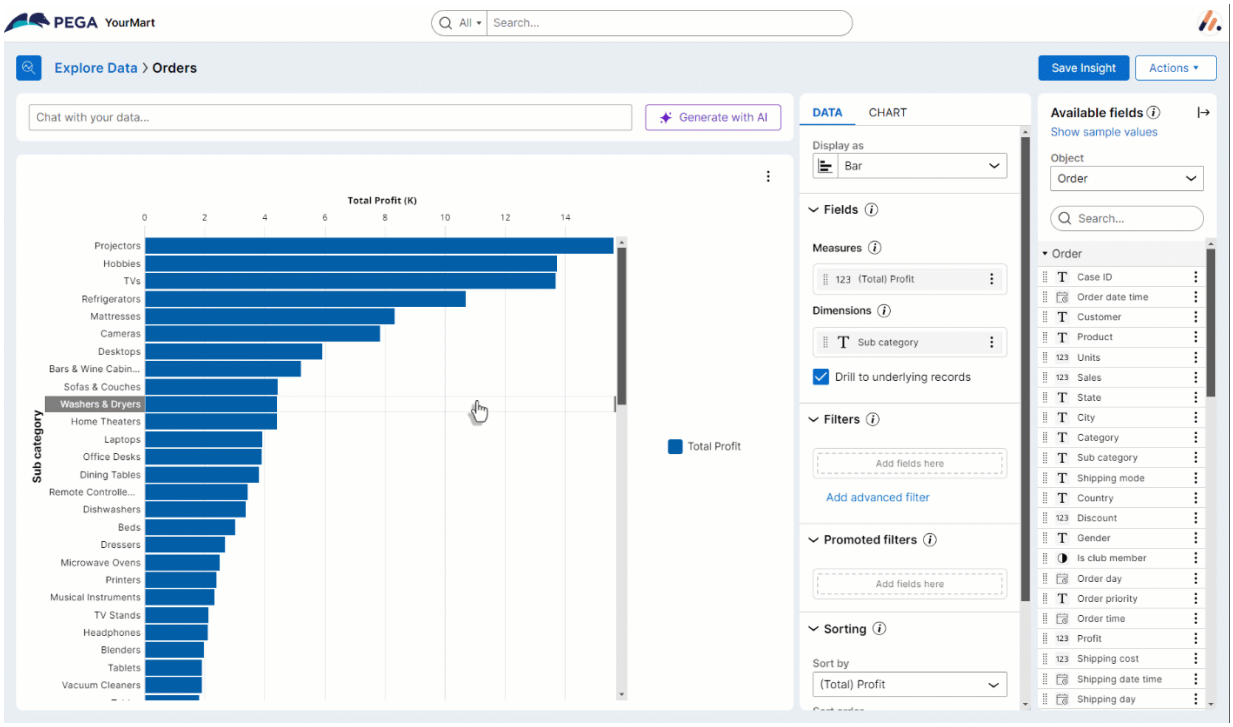
©2025 Pegasystems Inc

1. In the navigation pane of your application, click **Explore Data**.
2. In the upper-right corner of the application, click **Explore Data**, and then select specific type of data for which you want to create an Insight.
3. Customize the Insight based on your business requirements.

> **For example:**
>
> If you customize a chart-based Insight, from the **Available fields** section, drag the selected properties into the **Measures** and **Dimensions** areas to visualize the data.

For more information, see Customizing table-based Insights and Customizing chart-based Insights.

The following sample visualization shows the dynamic changes that occur in the chart-based Insight after adding the *Sales* property and modifying the chart type.



Authoring a chart-based Insight

4. In the upper-right corner of the screen, click **Save as Insight**.
5. In the **Save as Insight** dialog box, provide details for your Insight, and then click Submit.

©2025 Pegasystems Inc

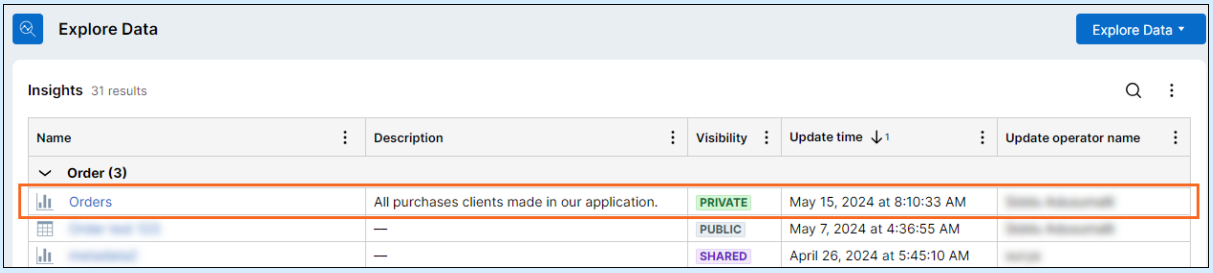
For example:

In the **Visibility** list, select **Shared** to share the Insight with your Access Group. Select the **Show title** checkbox, and then specify more details to give the chart a supplementary title.

For example:

After creating the Insight, you can view and access it in Explore Data.

The following figure shows Explore Data with a custom chart-based Insight for orders:



Insights in Explore Data

- **Generating Insights by using Pega GenAI**
- **Creating Insights from UI authoring**

Generating Insights by using Pega GenAI

Increase productivity by using Pega GenAI to create comprehensive Insights that match your advanced business requirements. In this way, the AI-generated Insight helps you save time on customization, and provides visually appealing data that is ready for analysis.

For example, when authoring Insights in a sales application, enter `top 5 subcategories with highest sales` to showcase bestselling products.

©2025 Pegasystems Inc

Pega GenAI in Pega Platform™ uses natural language processing to make the system read and process your prompts. This functionality simplifies the data browsing experience, and enables you to create comprehensive Insights in a convenient way.

Before you begin:

Enable Pega GenAI in your application. For more information see Connecting to the generative AI provider in Pega Cloud.

1. In the navigation pane of your application, click **Explore Data**.
2. In the upper-right corner of the application, click **Explore Data**, and then select specific type of data for which you want to create an Insight.
3. At the top of your Insight, in the search field, perform one of the following actions:
   - To select a suggestion generated by AI, select the prompt that matches your business requirements.
   - To generate a custom Insight, enter your prompt and then click **Generate with AI**.

   For example:

   In a sales application, enter `display 5 bestselling products as pie chart` to generate a pie chart with a visual distribution of sales among the top-performing products.In a sales application, enter `recent orders in Boston` to generate a table-based Insight with a list of orders from Boston sorted from the newest to the oldest.In a workload management application, enter `show me a list of assignments that I have opened` to generate a list of your assignments that are open at the moment.

4. In the upper-right corner of the Insight, click **Save as Insight**.

©2025 Pegasystems Inc

For example:

The following video shows the process of generating Insights in a sales application by using Pega GenAI. The video covers two effective methods: using AI-powered suggestions and entering custom text prompts.

Creating Insights from UI authoring

Extend the scope of meaningful information that your Cases include by creating custom Insights when UI authoring in App Studio. For example, when authoring an Order Case Type, you can edit the **Details** tab to feature a chart-based Insight with visualized orders of specific products in the last quarter.

Creating Insights when you authoring a UI helps you save time and improves the user experience. For more information, see Working with Views.

NOTE: Only tabs on your full Page Case View can feature Insights. For more information about configuring Full Views, see Configuring a Full Page View.

1. In the navigation pane of App Studio, click **Case Types**.
2. In the list of Case Types, click the Case Type that you want to edit.
3. In the **Case Type** customization View, click the **UX** tab.
4. On the right side of the Case Type, on the **Full Page View** tab, in the **Tabs** section, click the tab to which you want to add an Insight.
5. In the **Edit tab** section, under the region to which you want add an Insight, click **Add**, and then click **Insights** > **Create new Insight**.

©2025 Pegasystems Inc

Result:

The Insight authoring pane appears.

6. Customize the Insight according to your needs.
   For more information, see Customizing table-based Insights and Customizing chart-based Insights.
7. In the upper-right corner of the Insight, click **Save as Insight**.
8. In the **Save as Insight** dialog box, provide details for your Insight, and then click **Submit**.

For example:

In the **Visibility** list, select **Public** to share the Insight with every user. Select the **Show title** checkbox, and then specify more details to give the chart a supplementary title.

Result:

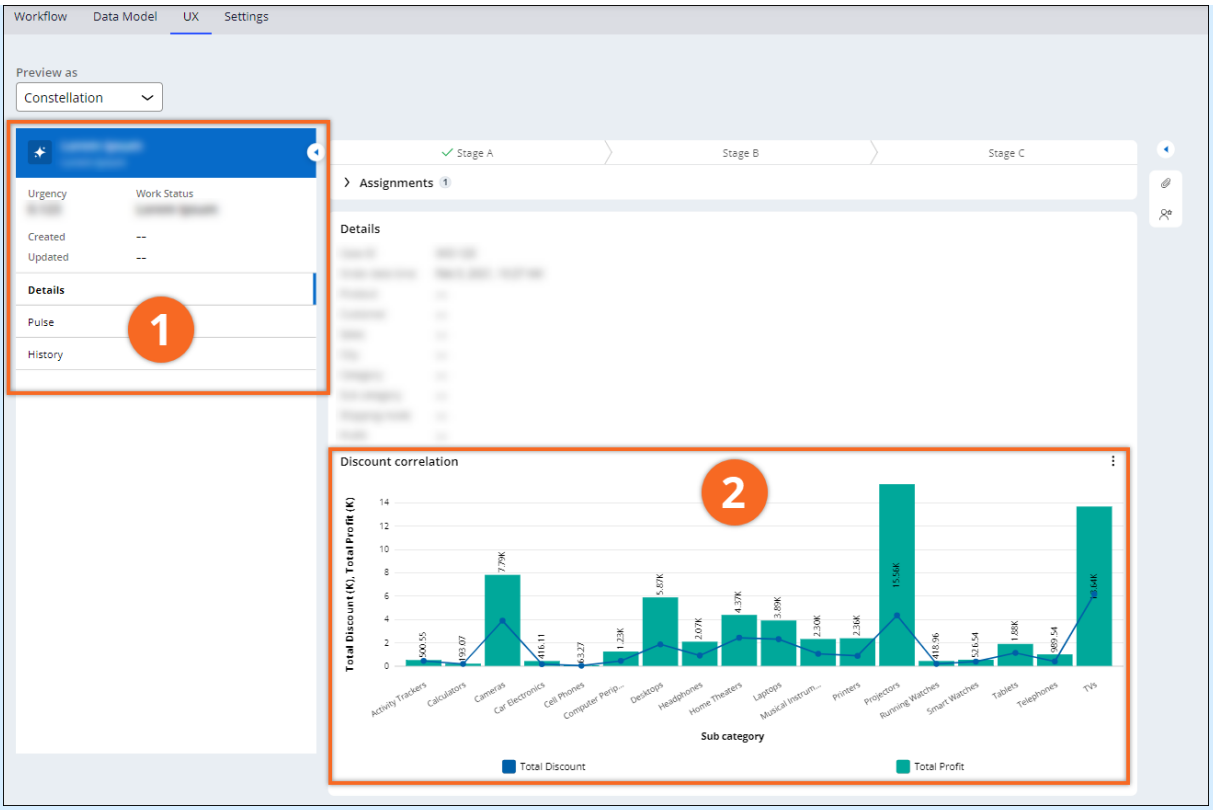
The Insight has been added to the specified region.

For example:

The following figure, which shows a preview of a sample Case Type, includes the following areas:

1. Case Type summary panel with tabs.
2. Sample chart-based Insight displayed in the open tab content area.

©2025 Pegasystems Inc



UI authoring of a Case Type

In this example, the **Details** tab features an Insight with data related to orders.

Managing Insights

You can tailor Insights by using a diverse range of settings to enhance legibility and highlight specific information. The available options help you align Insights with custom business requirements, and ensure that you showcase important data in a visually appealing way.

For example, in a sales application, you can color columns with values surpassing a specified threshold to draw viewers' attention to the meaningful data.

Pega Platform™ enables you to present your application data as either comprehensive table-based Insights or visually appealing chart-based Insights. You can manage their

©2025 Pegasystems Inc

content and appearances by using numerous built-in features, such as grouping, filtering, conditional formatting, or by changing charts.

The following video shows how managing a sample chart-based Insight works at run time:

- **Customizing table-based Insights**
- **Customizing chart-based Insights**
- **Applying promoted filters**
- **Managing the visibility of Insights**
- **Adding Insights to your application**
- **Scheduling Insights**

Customizing table-based Insights

Decide what data to present in table-based Insights by using intuitive tools that help you filter, sort, and group data. In this way, you can tailor the Insights to match specific business preferences, ensuring that only essential data is presented.

For example, in a sales application, you can modify the table-based Insights by organizing sales data by the order date. Then, you can apply custom filters to narrow down the displayed results.

> **Before you begin:**
>
> Add the Explore Data page to your application. For more information, see Organizing the main navigation for a standard work Portal.

©2025 Pegasystems Inc

> **NOTE:** You can add the Explore Data page only if your application uses Constellation or a traditional UI architecture hybrid mode.

1. In the navigation pane of your application, click **Explore Data**.
2. In the list of Insights, click the Insight that you want to modify.
3. In the upper-right corner of the Insight, click **Edit**.
4. In the customization pane of the Insight, on the **Data** tab, manage the content of your Insight:

Choices	Actions
Add visualization	a. Click the **Data** tab. b. In the **Display as** list, select how you want to present data. **For example:** Click **Chart** > **Column** to visualize data with columns.
Add columns	From the **Available fields** section on the right side of the Insight, drag the required columns to the **Columns** area.
Group data	a. From the **Available fields** section, drag the data by which you want to group the Insight into the **Groups** area. b. If you group data by dates, click the **More** icon, and then select certain date part to group data.

©2025 Pegasystems Inc

Choices	Actions
	For example: For the **Create Date/Time** field, select **Months**, to group data by months. **NOTE:** You can group data by multiple properties to drill-down to more specific data.
Apply basic filters	a. From the **Available fields** section, drag the data by which you want to filter the Insight into the **Filters** area. b. In the *Filter name* dialog box, select a relational operator for your filter, and then specify the details. c. If you filter results by dates, in the date parts list, select certain date part, and then specify the time period. **For example:** Select **Date part**, and then in the **Date part** and **Value** lists, select **Year** and **2021** respectively, to display results from the year 2021.

©2025 Pegasystems Inc

Choices	Actions
	d. Click Submit.
Apply advanced filters	a. In the **Filters** section, click Add advanced filter, and then, in the dialog box with filters, click Advanced mode. b. In the Advanced condition field, enter the filter logic, and then click Submit. **For example:** If you apply two filters, enter `1 OR 2` to filter data by either of the conditions.
Apply promoted filters	For more information, see Applying promoted filters.
Sort data	a. From the Available fields section, drag the data by which you want to sort the Insight into the Sorting area. b. Next to the field by which you want to sort data, click Menu > Update. c. In the Category list, select how to sort data, and then click Submit. **For example:** Select Descending to display values from highest to lowest.

©2025 Pegasystems Inc

Choices	Actions
Aggregate data	a. From the **Available fields** section on the right side of the Insight, drag a column with numeric data to the **Columns** area. b. In the table-based Insight, next to the added column, click **More**, and then select the aggregation method. **For example:** Click **Sum** to summarize all values of the added property.

5. On the **Table** tab of the Insight, define table-related settings:

Choices	Actions
Enable specific options related to Insights records	a. Expand the **Editing** section and enable options that meet your requirements. b. **Optional:** To enable users to edit data objects of the Insight that is contained within a Case, select the **Allow edit** checkbox. c. **Optional:** To enable users to perform bulk actions within the Insight, select the **Allow bulk actions** checkbox. d. **Optional:** To enable users to add new records to the Insight, select the **Allow create** checkbox.

©2025 Pegasystems Inc

Choices	Actions
	e. **Optional:** To enable users to reorder records within the Insight, select the Allow row reordering checkbox.
Enable users to create personalized views of Insights	a. Expand the User personalization section. b. Select the Allow saving custom views checkbox.

6. In the upper-right corner of the Insight, click Save insight.

For example:

The following video shows how to customize a table-based Insight in a sample sales application by using either Pega GenAI™ or other convenient built-in features.

Customizing chart-based Insights

Adjust the appearance and content of chart-based Insights to visualize only the information that is relevant to your business needs. With a wide variety of convenient built-in features, you can effectively communicate key metrics, identify trends, and make informed decisions that drive your business forward.

For example, in a sales application, you can customize chart-based Insights with a reference line that indicates satisfactory sales levels. Additionally, you can enhance the chart's visibility by color-coding columns that meet specific conditions, such as having sales levels above 1,000 units.

PEGA ©2025 Pegasystems Inc

Before you begin:

Add the Explore Data page to your application. For more information, see Organizing the main navigation for a standard work Portal.

NOTE: You can add the Explore Data page only if your application uses Constellation or a traditional UI architecture hybrid mode.

1. In the navigation pane of your application, click **Explore Data**.
2. In the list of Insights, click the chart-based Insight that you want to modify.
3. On the right side of the Insight, click **Edit**.
4. In the customization pane of the Insight, on the **Data** tab, manage the content of your Insight:

Choices	Actions
Change visualization	In the **Display as** section, in the types list, select the chart that you want to view.
Visualize specific data	a. From the **Available fields** section on the right side of the Insight, drag numerical data into the **Measures** area. **NOTE:** If you change the Object Type, the system discards all settings of the Insight, such as filter conditions, chart type, and selected data.

©2025 Pegasystems Inc

Choices	Actions
	b. Create a chart by dragging further data into the **Dimensions** area. **For example:** Drag the **Sales** and **Countries** records into the **Measures** and **Dimensions** fields respectively to visualize the sales across different countries.
Change measures	a. In the **Fields** section, in the **Measures** area, next to the record that you want to edit, click **More**, and then select a measure to include in the chart. **For example:** For the **Sales** record, select **Maximum** to distinguish bestselling articles only.
Apply basic filters	a. From the **Available fields** section, drag the data by which you want to filter the chart into the **Filters** area.

©2025 Pegasystems Inc

Choices	Actions
	For example: Drag the **Order date time** record to filter orders by date. b. In the list of relational operators, select an operator for your filter, and then specify the filter details. c. If you filter results by dates, in the date parts list, select certain date part, and then specify the time period. **For example:** Select **Relative date**, and then in the **Time period** and **Date part** lists, select **Current** and **Week(s)** respectively, to display results from the current week only. d. Click **Submit**. e. **Optional:** To apply more filters, drag more records into the **Filters** area.
Apply advanced filters	a. In the **Filters** section, click **Add advanced filter**, and then, in the dialog box with filters, click **Advanced mode**.

©2025 Pegasystems Inc

Choices	Actions
	b. In the **Advanced condition** field, enter the filter logic, and then click **Submit**. **For example:** If you apply two filters, enter `1 OR 2` to filter data by either of the conditions.
Sort data	a. In the **Sorting** section, in the **Sort by** list, select the data by which you want to sort the chart. b. In the **Sort order** list, specify how you want to sort the data. **For example:** Select **Ascending** to display values from lowest to highest. c. In the **Limit results** field, enter the number of results that you want to show in the chart. d. **Optional:** To show results with alike data, select the **Include duplicate results** checkbox. **NOTE:** If you select this checkbox, you might

©2025 Pegasystems Inc

Choices	Actions
	retrieve more results than the specified limit.

5. On the **Chart** tab of the Insight, manage how the system displays supplementary information in the chart:

Choices	Actions
Manage data series	In the **Series** section, next to the data series whose chart you want to update, click **More**, and then select specific option to manage. For more information, see Managing data series in Insights.
Customize the range of axes	a. Expand the **Axis** section. b. Manage the axes of the chart by selecting specific checkboxes. **For example:** Select **Customize range**, and then in the **Minimum** and **Maximum** fields, modify the scale of the chart.
Customize grid lines	a. In the **Grid** section, select the **Show grid lines** checkbox. b. If your Insight include only one data series, in the **Line style** list, select the look of grid lines.

©2025 Pegasystems Inc

Choices	Actions
	c. If your Insight include more than one data series, in the **Apply grid to** list, select the axis to which you want to apply the grid. **NOTE:** Applying grid lines to secondary axis is available after you specify the secondary axis in the **Axis** section. You can apply grid lines for two axis at once when customizing the bubble chart.
Toggle tooltips	In the **Tooltips** section, select the **Show tooltip** checkbox to display supplementary information about the chart columns.
Manage chart legend	a. In the **Legend** section, and then select the **Show legend** checkbox. b. Customize the chart legend by providing specific details.

©2025 Pegasystems Inc

Choices	Actions
	For example: In the **Type** list, select **checkboxes** to create a legend in the form of clickable checkboxes. **NOTE:** The system hides the columns that correspond to the checkboxes that you clear.

6. In the upper-right corner of the Insight, click **Save insight**.

For example:

The following video shows how to customize a chart-based Insight in a sample sales application by using either Pega GenAI™ or other convenient built-in features.

- **Managing data series in Insights**
- **Color-coding the appearance of data series**
- **Benchmarking data with reference points**
- **Grouping data in Insights**

PEGA ©2025 Pegasystems Inc

- Creating simple value charts
- Viewing underlying records of Insights

Managing data series in Insights

Increase the legibility of chart-based Insights by customizing the visualization options of data series. For example, in a sales application, you can include two distinct charts that visually depict the contrasting levels of costs and prices for particular products.

By managing the appearance of data series, you can effectively draw viewers' attention to the meaningful information, enhancing the clarity of the presented data.

1. In the navigation pane of your application, click **Explore Data**.
2. In the **Insights** section, click the Insight that you want to modify.
3. In the upper-right corner of the Insight, click **Edit**.
4. On the **CHART** tab, manage charts for specific data series:

Choices	Actions
Change the chart type	a. In the **Series** section, next to the specific data series whose chart you want to update, click **More** > **Edit chart type**. **NOTE:** This option is available if you specify more than one data series. b. In the **Edit chart type** list, select a chart type for the selected series.

©2025 Pegasystems Inc

Choices	Actions
	For example: Select **Line** to visualize data with points that are connected with a line.
Manage markers of data points **NOTE: Markers are customizable for the Line and Area chart types only.**	a. In the **Series** section, next to the specific data series whose chart you want to update, click **More** > **Edit markers**. b. In the **Edit markers** list, select the shape of markers for the data points. **For example:** Select **Triangle** to visualize data points with triangles. c. In the **Size** list, select the proportions of the markers.
Show labels with values	a. In the **Series** section, next to the specific data series whose chart you want to update, click **More** > **Edit data values**. b. In the **Edit data values labels** dialog box, select the **Show data values** checkbox. c. **Optional:** To select the location of labels in the chart, in the **Label**

©2025 Pegasystems Inc

Choices	Actions
	position list, select the location of the labels. **For example:** Select **Inside** to include the labels within columns. d. **Optional:** To display labels in the chart horizontally, select the **Use horizontal labels** checkbox.

5. If you want to manage the labels of either donut or pie charts, in the **Donut chart options** or **Pie chart options** respectively, perform the following actions:
   a. In the **Value label position** list, select the location of labels with values.
   b. In the **Percentage position** list, select the location of labels that indicate percentage of data distribution.
   c. In the **Category label position** list, select the location of labels with names of data categories.

      For example:

      Select **Inside segment** to embed the label within the chart.

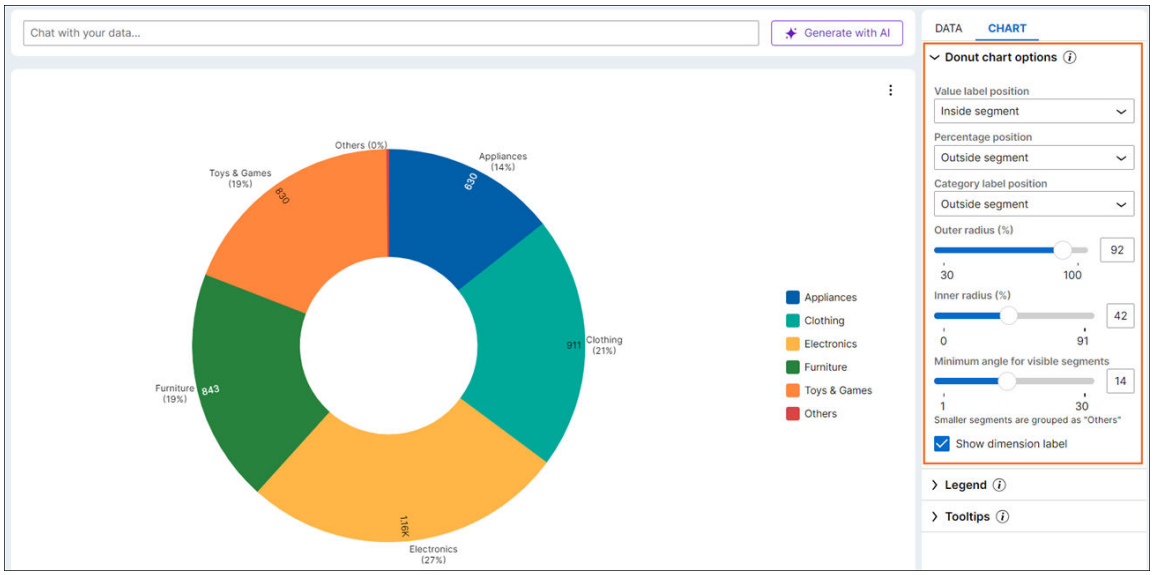
   d. Manage the visual appearance of the chart by moving the horizontal sliders.

      For example:

      Move the **Outer radius** slider to manage the size of the chart.

      The following figure shows a customized donut chart with inner radius slightly smaller than the outer radius.

©2025 Pegasystems Inc



Customization of a donut chart

6. In the upper-right corner of the Insight, save changes by clicking **Save Insight**.

For example:

The following video shows how managing data series works at run time in a sample sales application. In this example, the data series are visualized with distinct charts and are accompanied by value labels, enhancing the legibility of the information.

Color-coding the appearance of data series

Enhance the process of data analysis by applying colors to columns which values meet your specific criteria. In this way, you can bring meaningful data ranges into focus, making it convenient for viewers to effortlessly interpret the data.

For example, in a chart-based Insight that features sales data, you can color columns with values that surpass a predefined threshold to make that data visually stand out.

©2025 Pegasystems Inc

> **NOTE:** If you specify multiple conditions and the specific data series fulfills criteria for more than one condition, then the system prioritizes the condition that is at the top of the list.

1. In the navigation pane of your application, click **Explore Data**.
2. In the list of Insights, click the Insight that you want to modify.
3. In the upper-right corner of the Insight, click **Edit**.
4. On the **Chart** tab, expand the **Conditional formatting** section, and then click **Add condition**.
5. In the **Add condition** dialog box, define the color conditions for specific data series by building a logical expression, and then selecting a color.

   > **For example:**
   >
   > For the `Sales` data series, to color each column whose value surpasses the 50,000 threshold, select the **greater than** operator, and then enter `50000`.

6. Click **Submit**.
7. In the upper-right corner of the Insight, click **Save Insight**.

> **For example:**
>
> The following video shows the process of color-coding data series in a sample chart-based Insight. In this example, custom sales ranges among states have been highlighted by applying two different colors: states with sales up to 25,000 are marked in red, whereas states with sales exceeding 50,000 are marked in green.

Benchmarking data with reference points

Maximize the informative value of your chart-based Insights by including supplementary elements that highlight the most meaningful data.

©2025 Pegasystems Inc

For example, in a sales Insight, you can include a horizontal reference line that indicates the expected revenue to identify departments with subaverage results. You can also color specific regions of the chart to identify a value range that is substantial for your Insight.

1. In the navigation pane of your application, click **Explore Data**.
2. In the list of Insights, select the Insight that you want to modify.
3. In the upper-right corner of the Insight, click **Edit**.
4. On the **Chart** tab, in the **Reference lines and regions** section, manage your chart:

Choices	Actions
Add reference lines	a. In the **Lines** section, click **Add line**. b. In the **Add line** dialog box, customize the reference line. **For example:** In the **Value** field, enter `20000` to locate the line on the axis next to the specified value. c. Click **Submit**.
Mark specific areas in the chart	a. In the **Regions** section, click **Add region**. b. In the **Add region** dialog box, customize the area on the axis that you want to feature. **For example:** In the **Min** and **Max** fields, enter `1000` and `5000` values

©2025 Pegasystems Inc

Choices	Actions
	respectively, to distinguish the area within that range. c. Click **Submit**.

5. In the upper-right corner of the Insight, click **Save Insight**.

For example:

In the following example, the reference line indicates a satisfactory level of sales by state, whereas the region marked in green shows the range of minimum expected sales.

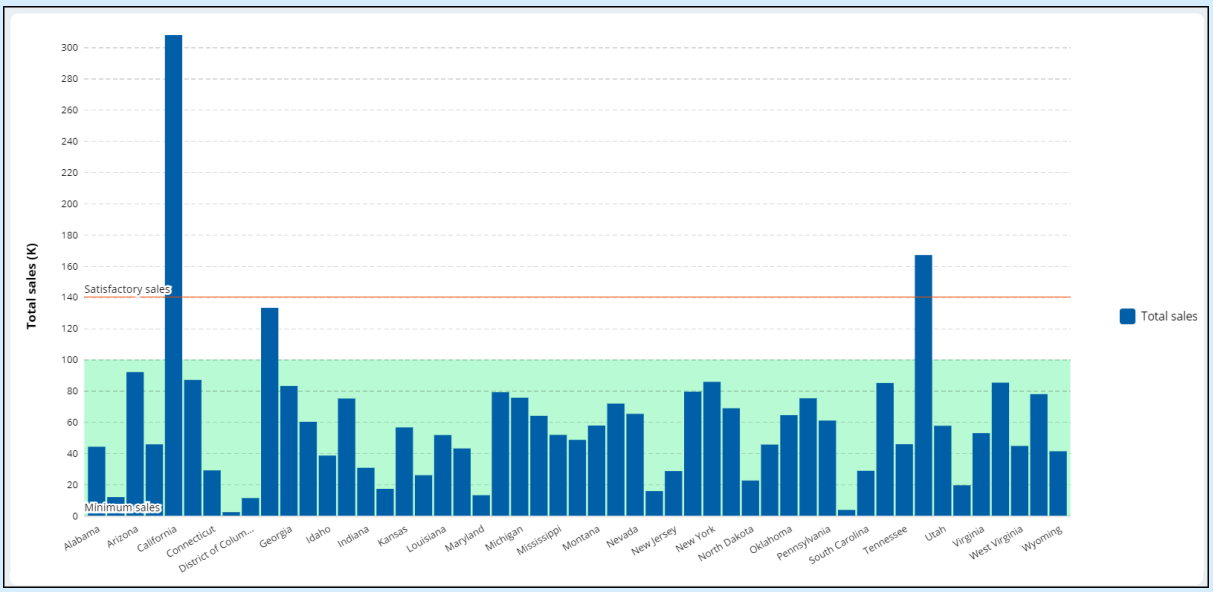


Chart-based Insight with region and reference line

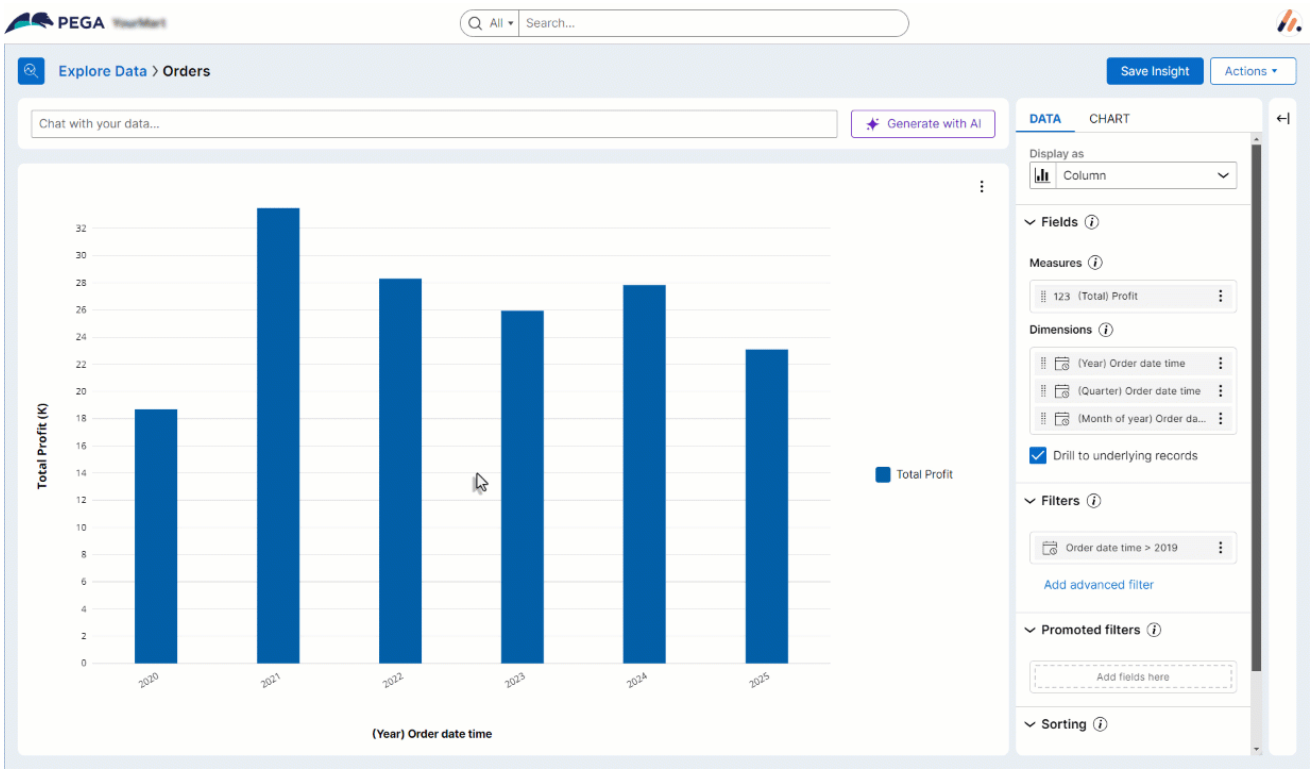
Grouping data in Insights

You can increase the legibility of chart-based Insights by grouping data according to your current business requirements, making data analysis process more convenient. By grouping Insights' data, you can present meaningful information in a clear and organized manner.

©2025 Pegasystems Inc

Pega Platform™ provides various options for organizing data in Insights, such as adding custom numeric groups, grouping information by custom categories, or by dates. With the latter, you can organize data by custom periods of time, making data analysis more convenient. For example, in a sales application, you can group the data by months to identify trends among clients over time.

The following visualization shows a sample chart-based Insight for sales, organized by years, quarters, and months. By customizing Insights that way, you retrieve granular information about orders from specific timeframes.



Grouping data by certain periods

For more information about various types of grouping, see the following articles:

- **Grouping information by dates**
- **Grouping Insights information into custom categories**

©2025 Pegasystems Inc

- **Adding custom numeric groups**

Grouping information by dates

Retrieve data from specific time periods more quickly by grouping the content of chart-based Insights by date. In this way, you enhance clarity and legibility, making data analysis more convenient. For example, in a sales application, you can group orders by months to identify trends among clients throughout the year.

You use date properties, such as *Order date time*, to group chart-based Insights. You can use the grouping option to limit the results, and then drill down to specific data.

> **NOTE:** If you want to retrieve more granular information, add multiple date properties to a single Insight, and then specify date parts such as years, months, or days.

1. In the navigation pane of your application, click **Explore Data**.
2. In the **Insights** section, click the chart-based Insight that you want to modify.
3. On the right side of the Insight, click **Edit**.
4. Group the Insight data by specific period:
   a. From the **Available fields** section on the right side of the Insight, drag the relevant date property into the **Dimensions** area.

      > **For example:**
      >
      > Drag the *Order date time* property.

   b. Next to the date property, click **More** > **Date parts**, and then select the specific period by which you want to group the data.

©2025 Pegasystems Inc

> **For example:**
>
> Select **Month** to display separate results for each month within the specified date range. Select **Month of year** to summarize data of the same months within the specified date range.

5. In the upper-right corner of the Insight, click **Save Insight**.

> **For example:**
>
> The following video shows how to group the contents of Insights by dates in a sample sales application. In this example, data is grouped by years and months to retrieve granular information from extensive data pool.

Grouping Insights information into custom categories

Enhance the legibility of Insights by organizing data into customized groups. With this functionality, you can specify multiple groups with data based on strict criteria that match your current needs.

By using the grouping functionality, you can differentiate specific data from the rest of the data, or even exclude irrelevant data that you do not need at the moment. For example, in a sales application, you can create two custom groups for orders originating from the eastern and western coasts of the US, to visualize the distribution of sales across the country.

> **Before you begin:**
>
> Create an Insight. For more information, see Creating Insights.

1. In the navigation pane of your application, click **Explore Data**.

©2025 Pegasystems Inc

2. In the list of Insights, click the chart-based Insight that you want to modify.
3. On the right side of the Insight, click **Edit**.
4. On the **Data** tab, in the **Dimensions** section, next to the property that you want to group, click **More** > **Add custom groups**.
5. In the **Add custom groups** dialog box, customize the group:
   a. In the **Groups** section, click **Add group**.
   b. In the **Add group** dialog box, specify the group details, and then click **Submit**.

      For example:

      For the sales Insight, in the **Select values** list, select **Washington**, **Oregon**, and **California** to create a group with data from western states of US.

   c. **Optional:** To add more groups, repeat steps 5.a to 5.b.
   d. In the **Remaining values** list, decide what to do with the data that your groups do not include, and then click **Submit**.

      For example:

      Select **Place in "Other group"**, and then provide the name of the group in which to include the remaining values.

6. In the upper-right corner of the Insight, click **Save insight**.

For example:

The following video shows how to group the contents of Insights into custom categories in a sample sales application. In this example, the sales data has been organized into custom categories containing information about sales data from the Southwest and West states of the U.S.

©2025 Pegasystems Inc

Adding custom numeric groups

Increase the legibility of Insights by adding custom numeric groups, which emphasize crucial information, and help you to analyze data more conveniently. For example, in a sales Insight, you can group values by custom ranges to visualize the distribution of products based on their sales levels.

When adding groups, you can choose either to visualize data as intervals or data ranges. With intervals, you can group values into custom categories, while data ranges help you define custom scope for values.

1. In the navigation pane of your application, click Explore Data.
2. In the list of Insights, click the Insight that you want to modify.
3. On the right side of the Insight, click Edit.
4. If you customize a table-based Insight, from the **Available fields** drag the numeric value to the Groups field, and then next to the property click More > Add custom groups.
5. If you customize a chart-based Insight, in the **Dimensions** section, next to the numeric property that you want to group, click More > Add custom groups.

   For example:

   Select the Sales property.

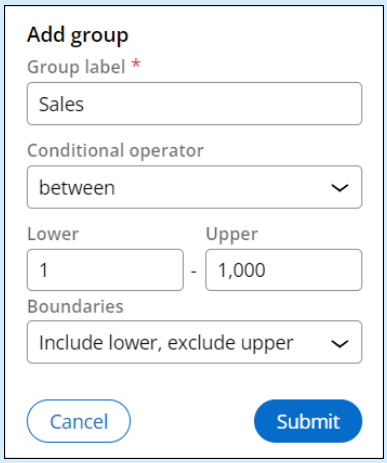
6. In the **Add custom groups** dialog box, customize the group:

Choices	Actions
Add custom ranges	a. In the Mode list, select Range. b. In the **Groups** section, click Add group.

©2025 Pegasystems Inc

Choices	Actions
	c. In the **Add group** dialog box, specify the group details, and then click Submit. **For example:** For the sales Insight, in the Conditional operator list, select between, and specify the values for your Insight to include. **NOTE:** If you select between or not between, in the Boundaries list, you can include or exclude top and lower values. In the following example, the upper value (1000) is

©2025 Pegasystems Inc

Choices	Actions
	excluded from the results.  d. In the **Remaining values** list, decide what to do with the data that your groups do not include, and then click **Submit**. **For example:** Select **Place in "Other group"**, and then provide the name of the group in which to include the remaining values.
Add custom intervals	a. In the **Mode** list, select **Interval**. b. In the **Interval** field, group the results by specific value.

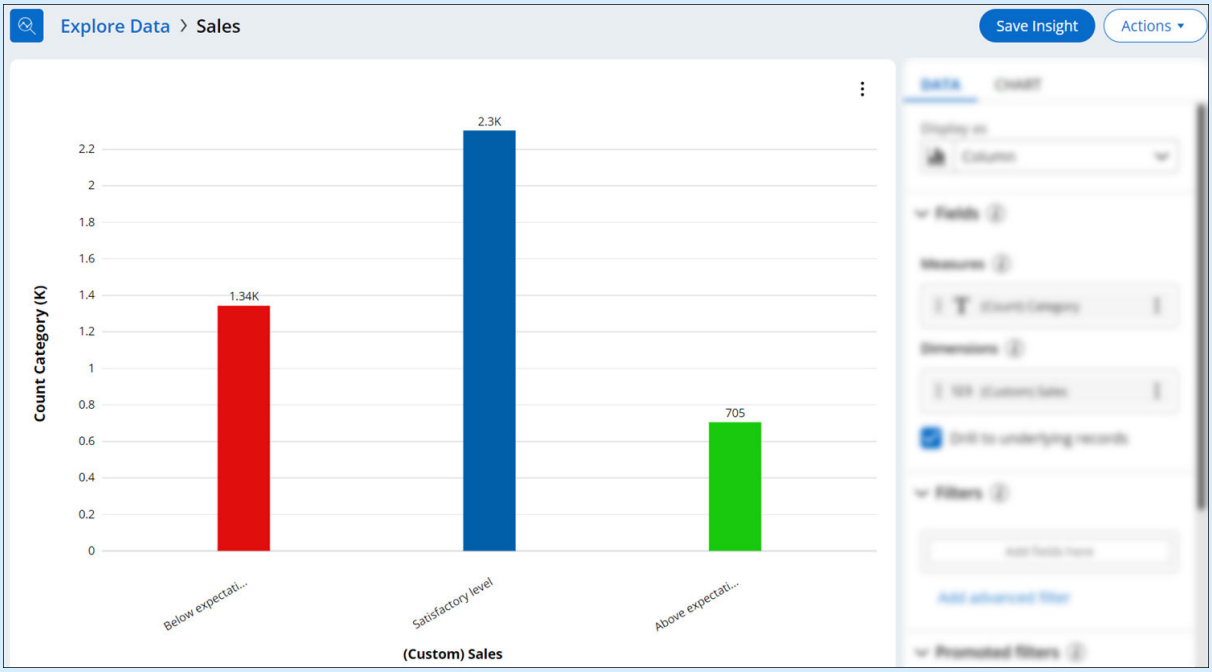
©2025 Pegasystems Inc

Choices	Actions
	For example: In a sales Insight, enter `1000` to group results every thousand sales. c. In the **Boundaries** list, manage top and low values of the intervals. d. Click **Submit**.

7. In the upper-right corner of the Insight, click **Save Insight**.

For example:

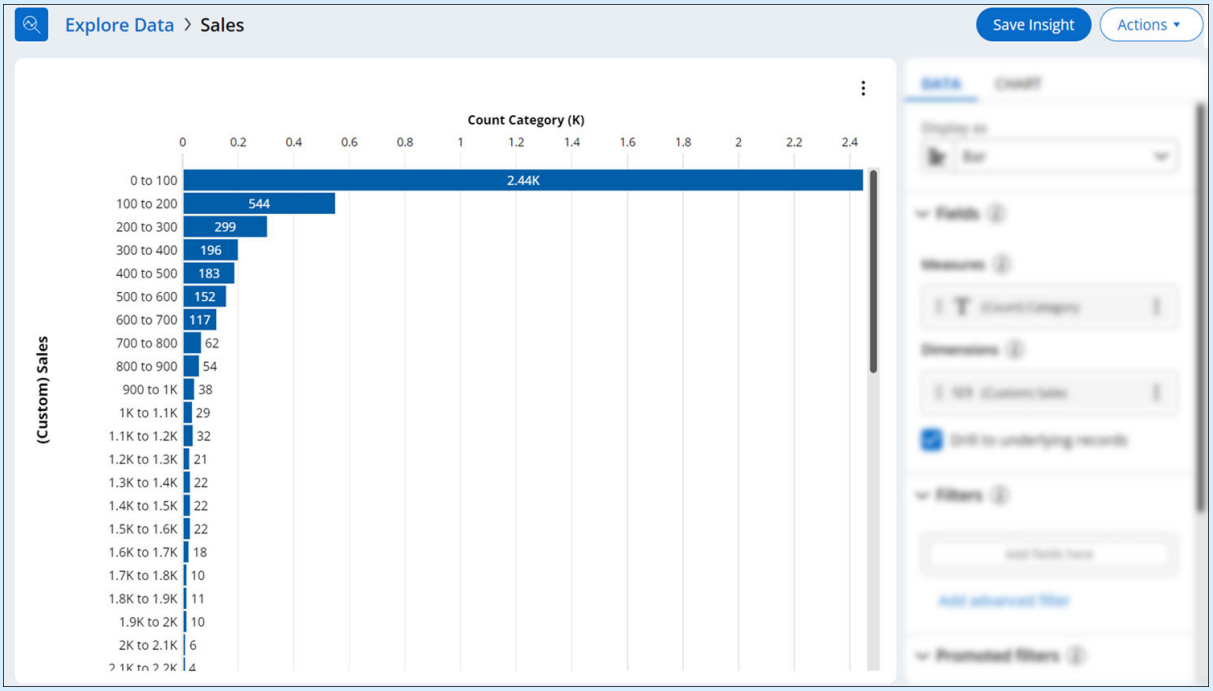
The following Insight consists of three custom ranges that visualize levels of sales in a company. In this example, the sales of 705 products exceeds expectations, whereas over 1300 articles do not reach the anticipated target.



Ranges in a sample chart-based Insight

©2025 Pegasystems Inc

In turn, the following figure shows the same sales data, but with intervals. This example shows that most items are sold for prices up to USD 100.



Intervals in a sample chart-based Insight

Creating simple value charts

Present crucial application data clearly by arranging information in the form of a customized layout. For example, in a retail application, you can create an Insight that highlights data about total sales and profits to help users focus on important information.

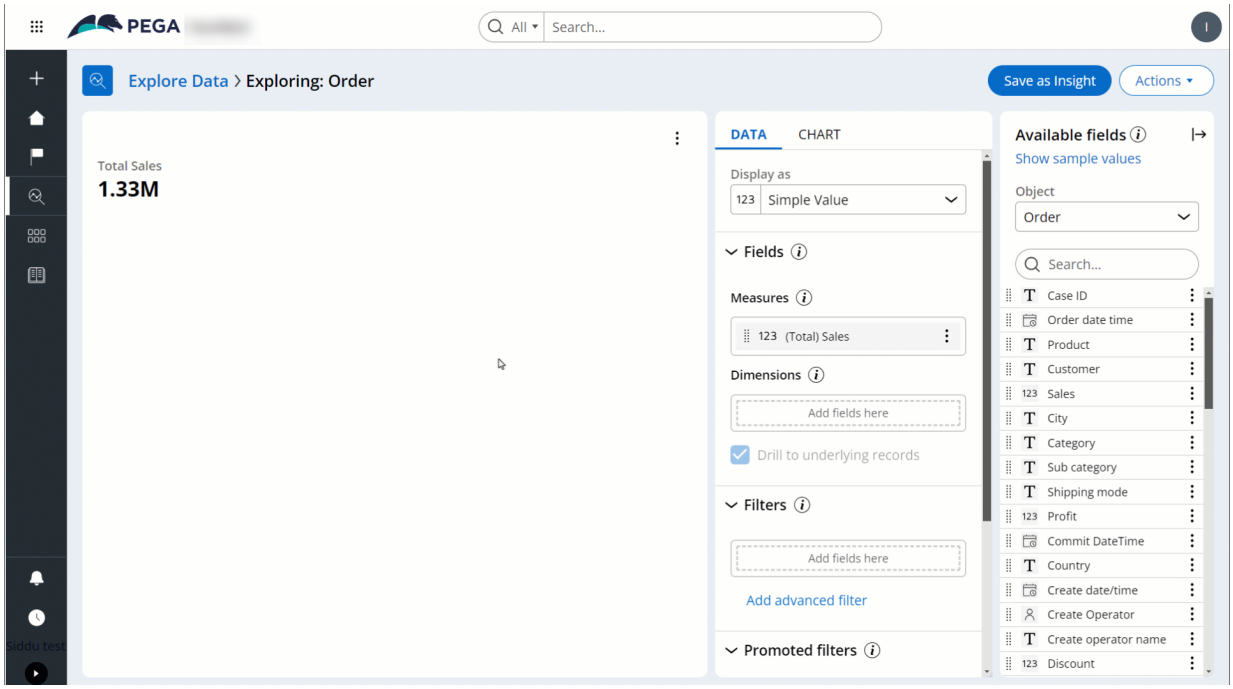
Creating an Insight

1. In the navigation pane of your application, click **Explore Data**.
2. In the upper-right corner of the application, click **Explore Data**, and then select specific type of data for which you want to create an Insight.
3. In the **Data** tab, in the **Display as** list, select **Chart** > **Simple value**.
4. From the **Available fields** section on the right side of the window, drag the required data into the **Measures** area.

©2025 Pegasystems Inc

You can include numerical data and non-numerical data, such as sales quotas and customer categories.

> **NOTE:** The simple value chart updates every time you drag a data object into the **Measures** area as in the following visualization.



Adding data objects to simple value chart

5. **Optional:** To specify calculations for your data, in the **Measures** area, next to the specific data series, click **More**, and then select the required option.

> **For example:**
>
> For the Customers data series, click **Distinct count** to retrieve the number of unique customers.

Customizing the chart's appearance

6. Click the **Chart** tab.

©2025 Pegasystems Inc

7. Expand the **Simple value display** section, and in the **Layout** list, specify the layout of the chart:
   - To feature the values of all data series, select **Highlight**.
   - To feature the value of one data series only, and include other data below that series, select **Highlight + summary**.
   - To create a custom layout, select **Custom highlight + summary**.
8. Depending on the layout that you select, customize specific arrangement settings.

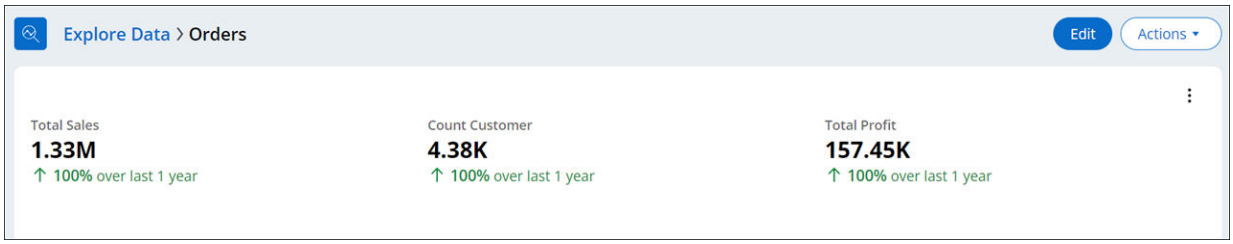
   > **For example:**
   >
   > For the **Custom highlight + summary** layout, in the **Highlights** field, enter 3 to feature three data series.

9. **Optional:** To show how values change over time, expand the **Trending** tab, and perform the following actions:
   a. Next to the data object that you want to edit, click **Menu** > **Add trend**.
   b. In the **Add trend** dialog box, specify details that you want to measure.

      > **For example:**
      >
      > In the **Date field** list, select **Create date/time**, and in the **Time period** and **Date part** lists, select **Last** and **Year(s)** to measure how values changed in the last year.

   c. Click **Submit**.

   The following figure shows last year's trends for sales, number of customers, and profits for a mock company.



Trends in a simple value chart

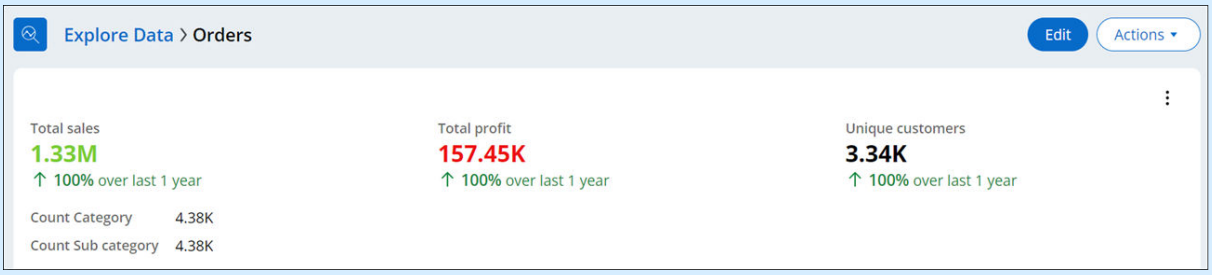
©2025 Pegasystems Inc

10. **Optional:** To apply colors to the data series, in the **Conditional formatting** section, customize the labels.
    For more information, see Color-coding the appearance of data series.
11. In the upper-right corner of the Insight, click **Save insight**.

For example:

The following example presents a **Custom highlight + summary** layout that includes information about sales, profits, number of customers, and product categories. The layout consists of three featured data series (sales, profits, and customers) that are divided into distinct columns, and supplementary data about product categories below the columns. Additionally, the featured data series include information about trends over the last year.

The data series are color-coded to focus on important information.



A sample Insight with a simple value chart

Viewing underlying records of Insights

Improve the process of data analysis by retrieving data that comprise your chart-based Insight. You can view all the underlying records or drill down to the specific information to get more precise data.

For example, in a sales Insight, you can delve into comprehensive data about sales recorded in specific city to identify bestselling articles within that region. In this way, you can conveniently verify current trends among clients to optimize sales strategies.

1. In the navigation pane of your application, click **Explore Data**.

©2025 Pegasystems Inc

2. In the **Insights** section, click the chart-based Insight that you want to modify.
3. On the right side of the Insight, click **Edit**.
4. View the underlying data of your Insight:

Choices	Actions
View all primary fields defined for the Case	a. In the upper-right corner of the Insight, click **More** > **View data**. b. In the modal dialog box that appears, analyze the data.
View records that comprise the specific category	a. On the **Data** tab, in the **Dimensions** section, select the **Drill to underlying records** checkbox. b. Retrieve the list of specific data by drilling-down to the last category of records. **For example:** In a sales Insight, click **Electronics**, and then click**Notebooks** to get information about sales of notebooks.

5. **Optional:** To manage data and appearance of the Insight, perform the following actions:
   a. In the upper-right corner of the Insight, click **More** > **View data**.
   b. In the upper-right corner of the dialog box, click **More**, and then click the required option.

   For example:

   Click **Export to Excel** to save the Insight as Microsoft Excel file.

©2025 Pegasystems Inc

Click **Filter** to apply custom filters to the data that you see in the dialog box.

> **NOTE:** The filter does not apply to the Insight, but to the data preview. To access the customization options, you can also drill down to the last category that is defined in the **Dimensions** section of the Insight.

For example:

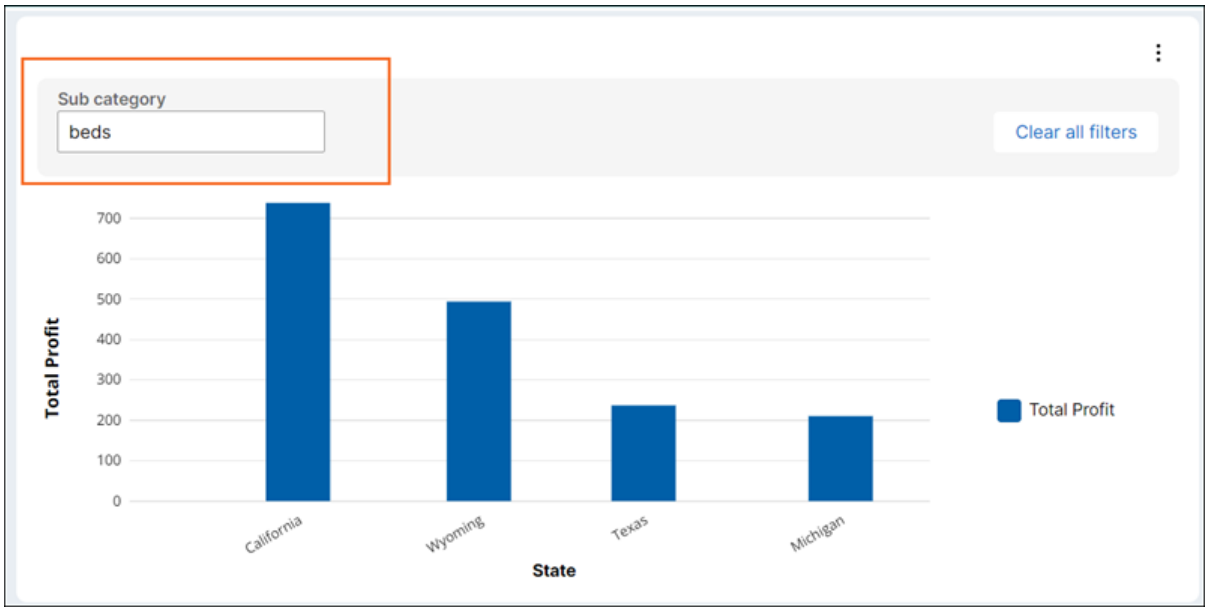
The following video shows how to view underlying records of a sample chart-based Insight. The first method shows drilling down to detailed information about sales recorded in Burbank, California. In turn, the other shows how to view all data that comprise the Insight to get a holistic overview of the dataset.

Applying promoted filters

Enhance the process of data analysis by using promoted filters that help you effortlessly retrieve the required information. The filters appear at the top of Insights to ensure comfortable data customization at run time.

For example, in a sales Insight, you can define the **Sub category** field as a promoted filter, and then enter `beds` to retrieve information about profits generated from selling beds only:

©2025 Pegasystems Inc



Promoted filter in a chart-based Insight

1. In the navigation pane of your application, click **Explore Data**.
2. In the list of Insights, select the Insight that you want to edit.
3. On the right side of the Insight, click **Edit**.
4. From the **Available fields** section, drag the data by which you want to filter the Insight into the **Promoted filters** field.

   > **Result:**
   >
   > The promoted filter field appears at the top of the chart.

5. Filter data of your Insight:

Choices	Actions
Use simple filters	At the top of the chart, in the promoted filters field, enter the value by which you want to filter the Insight.
Use advanced filters	a. On the Data tab, in the **Promoted filters** section, next to the filter that

©2025 Pegasystems Inc

Choices	Actions
	you want to use, click More > Edit filter. b. **Optional:** To customize the name of the filter, in the Filter label field, enter the new name. c. In the Display as list, select Advanced, and then click Submit. d. At the top of the chart, in the list, select a relational operator, and then specify a value. **For example:** For the Shipping cost field, select is equal to, and then enter `400` to present orders whose shipping cost equals 400. e. Click Apply.
Use custom filters	a. On the Data tab, in the **Promoted filters** section, next to the filter that you want to use, click More > Edit filter. b. In the Display as list, select Custom. c. In the **Options** section, click Add options, and then in the **Add filter option** dialog box, specify your custom filter conditions.

©2025 Pegasystems Inc

Choices	Actions
	For example: For the Create date/time field, select before, and then specify date and time to add a custom promoted filter that shows data that match your criteria.
Use picklist	a. On the Data tab, in the **Promoted filters** section, next to the filter that you want to use, click More > Edit filter. b. In the Display as list, select Picklist, and then click Submit. c. At the top of the chart, in the list that appears, select values by which you want to filter the Insight. **For example:** For the *Customer* filter, select Tom Smith and Emily Johnson to showcase all orders made by the customers.

6. In the upper-right corner of the Insight, click Save Insight.

©2025 Pegasystems Inc

For example:

The following video shows how to apply promoted filters in a sample chart-based Insight with sales data. In this example, the data about profits generated within different states is filtered by using two distinct filter types: Default and Picklist.

Managing the visibility of Insights

Control which Insights are accessible to other users of your application by setting specific visibility options. In this way, you ensure that sensitive or confidential information remains restricted to authorized users, maintaining data privacy and security.

For example, as an application administrator, you can make your Insights available for use on landing pages by exposing the Insights as public visibility.

Before you begin:

Add the Explore Data page to your application. For more information, see Organizing the main navigation for a standard work Portal.

NOTE: You can add the Explore Data page only if your application uses a Constellation or Theme Cosmos in hybrid mode.

1. In the navigation pane of your application, click **Explore Data**.
2. In the **Insights** section, click the Insight whose visibility you want to edit.
3. In the upper-right corner of the Insight, click **Edit**.
4. Specify the visibility of the Insight:

©2025 Pegasystems Inc

a. In the upper-right corner of the Insight, click **Actions** > **Insight details**.
b. In the **Insight details** modal dialog box, in the **Visibility** list, select who can see your Insight.

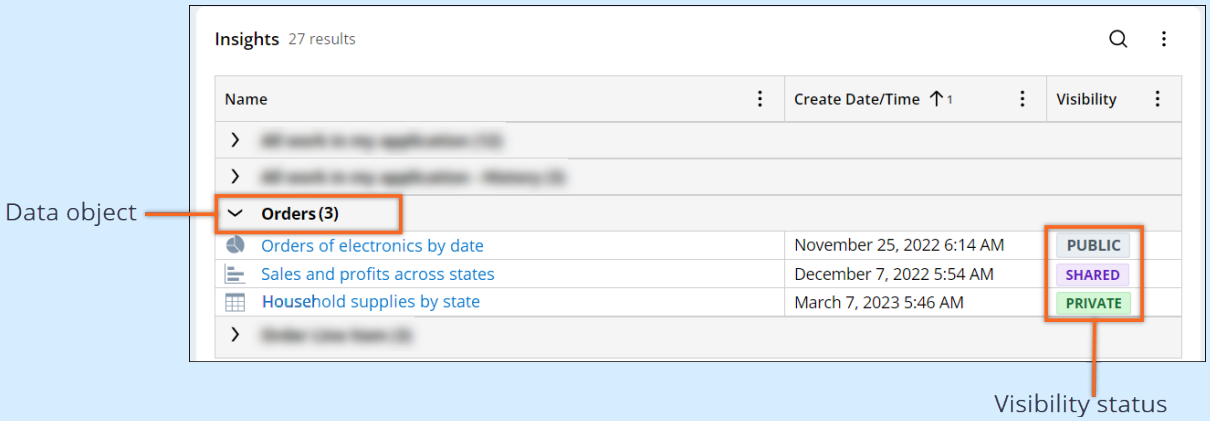
> **For example:**
>
> Click **Shared**, and then in the **Shared** list specify if you want to share the Insight with your access group, everyone, or with specific users.

5. **Optional:** To give the chart a supplementary title, select the **Show title** check box, and customize the title.

> **For example:**
>
> In the **Title alignment** list, select **Center** to display the chart title in the center.

6. Click **Submit**.

> **For example:**
>
> The following figure shows an extract from Explore Data in which you can check the visibility of Insights.



©2025 Pegasystems Inc

Adding Insights to your application

Help users find relevant Insights more quickly by adding appropriate pages to the navigation menu of your application. By using Insights, your users gain quick access to application data in the form of interactive charts or comprehensive lists.

For example, you can include a page with chart-based Insights that illustrate the performance of your sales department, to provide a visual representation of key metrics and trends.

> **NOTE:** You can add an Insight as a page only when the Visibility of the Insight is set to `Public`.
>
> Only application administrators can designate Insights as `Public`. For more information, see Managing the visibility of Insights.

> **Before you begin:**
>
> Enable Constellation-based landing pages. For more information, see Constellation in traditional UI and Constellation UI.

1. Open the portal in which you want to use Insights:
   a. In the navigation pane of App Studio, click **Channels**.
   b. In the **Current channel interfaces** section, click the icon that represents the portal to which you want to add a custom page with Insights.
2. On the portal page, on the **Content** tab, click the **Landing pages** category.
3. In the **Landing pages** pane on the left side of the screen, click **Add**.
4. In the **Add a page** modal dialog box, provide details for the page, and then click **Submit**.

©2025 Pegasystems Inc

For example:

In the **Name** field, enter `Sales Insights` as the title for a page with Insights about the finances of your company. In the **Template** list, select **Two column** to include two chart-based Insights next to each other.

5. Customize the newly created page.
   For more information about managing pages with list view templates, see Configuring a list-based landing page.

   For more information about managing pages with column templates, Configuring a column-based landing page.

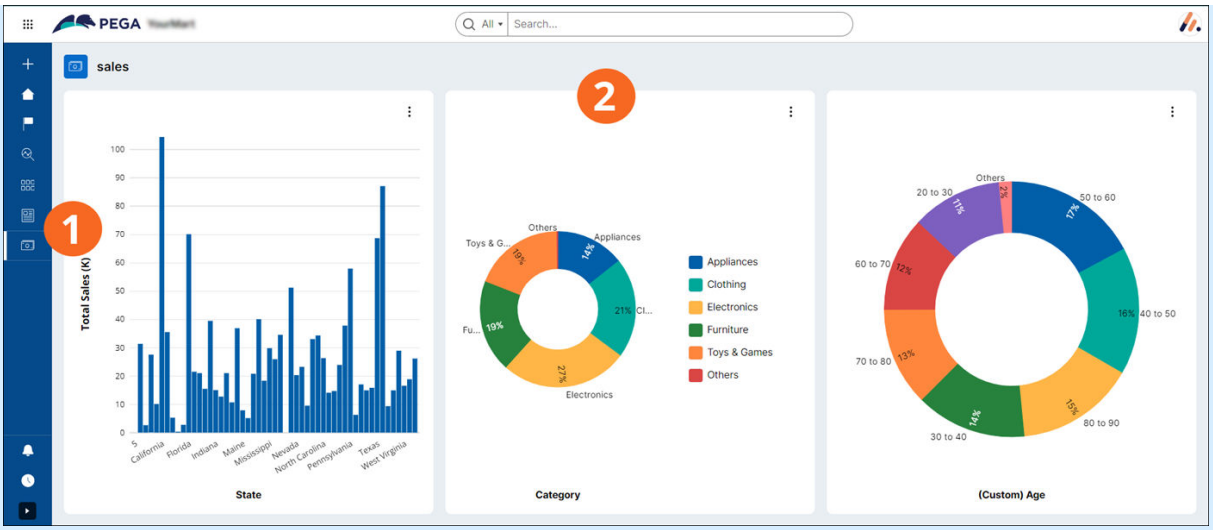
6. Add the page to the navigation menu of your application.
   For more information, see Organizing the main navigation for a standard work Portal.

For example:

The following figure shows a sample landing page that uses a three-column template. The landing page includes chart-based Insights with sales data. The figure includes:

1. Navigation panel with landing pages.
2. Selected landing page with chart-based Insights.

©2025 Pegasystems Inc



Sample landing page with Insights

Scheduling Insights

Distribute crucial information about your application to the email accounts of specific users by scheduling Insights in Explore Data. In this way, you save time and stay informed about the changes that occur in the application.

Pega Platform™ distributes Insights as Excel files, which help recipients conveniently analyze the data. For example, in a sales application, you can schedule an Insight to recur every week, to get detailed information about orders placed by clients. Consequently, you can monitor sales activity and take proactive measures to achieve sales targets or address any potential challenges.

> **NOTE:** If you want to control the number or records that the system exports, manage the *pyExportMaxRecords* Rule on Reporting landing page. For more information, see Setting resource limits for reporting.

1. In the navigation pane of your application, click **Explore Data**.
2. In the list of Insights, select the Insight that you want to schedule.

©2025 Pegasystems Inc

> **NOTE:** You can only schedule table-based Insights.

3. In the upper-right corner of the Insight, click Actions > Schedule.
4. In the **Schedule manager** dialog box, click New schedule.

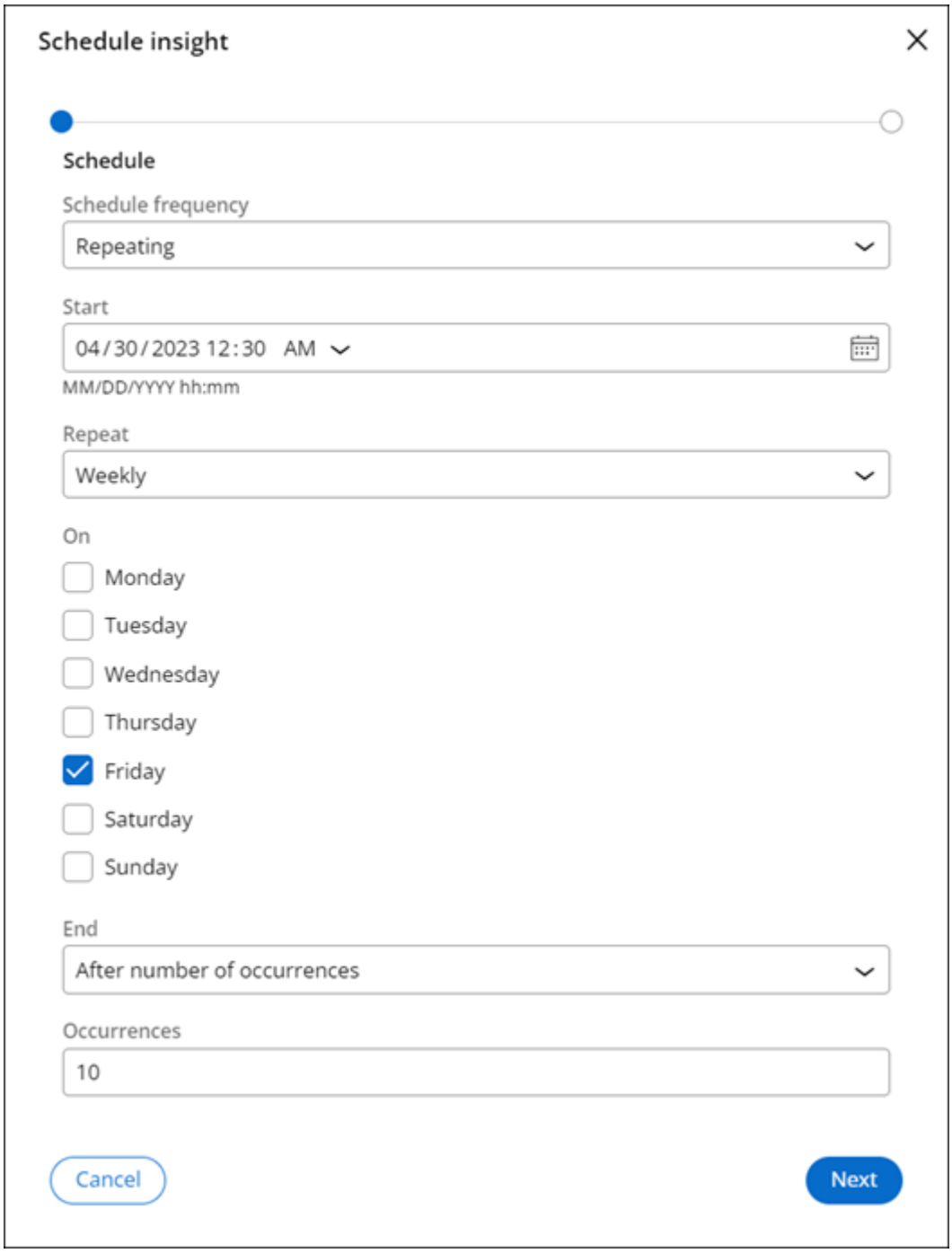
> **NOTE:** If the Insight is already scheduled, the **Schedule manager** dialog box displays information about active schedules, which you can customize.

5. In the **Schedule insight** dialog box, in the **Schedule frequency** list, select how often you want the system to distribute the Insight:

Choices	Actions
Distribute once	a. In the Schedule frequency list, leave the One time option selected, and then decide when to generate the Insight. b. If you click Later, in the **Date** section, specify when you want to generate the Insight. c. Click Next.
Distribute repeatedly	a. In the Schedule frequency list, select Repeating. b. Specify the recurrence settings, and then click Next.

The following figure shows an Insight that is scheduled to recur every Friday for the next 10 weeks:

©2025 Pegasystems Inc



Insight's scheduling

6. Add recipients for the Insight:
   a. In the **Send to** section, click Add.

©2025 Pegasystems Inc

b. In the **New recipient** dialog box, specify the recipient of your Insight, and then click Add.
c. **Optional:** To add more recipients, repeat steps 6.a to 6.b.
d. Click Submit.

Result:

The system displays a prompt with confirmation that the Insight is scheduled.

- **Managing scheduled Insights**

Managing scheduled Insights

Control the distribution of Insights among users of your application by accessing a schedule manager that helps you modify, remove, and create new schedules. In this way, you can conveniently manage the flow of information and ensure that users receive necessary Insights on time.

For example, you can increase the frequency of distribution of an Insight with sales data to keep users informed about recent orders or earnings.

1. In the navigation pane of your application, click Explore Data.
2. In the **Insights** pane, click the Insight for the schedule that you want to manage.
3. In the upper-right corner of the Insight, click Actions > Schedule.
4. In the **Schedule manager** dialog box, manage the scheduled Insights:

Choices	Actions
Modify schedules	a. In the **Active schedules** section, next to the schedule that you want to modify, click the Configure icon. b. Customize the parameters of the schedule, and then click Next.

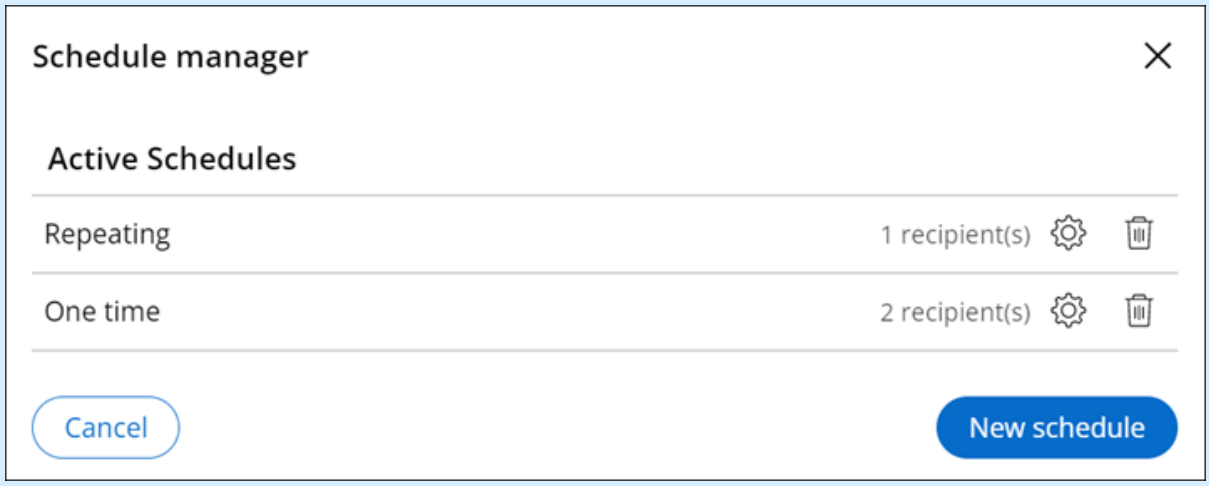
©2025 Pegasystems Inc

Choices	Actions
	For example: In the Schedule frequency list, select Repeating to distribute the Insight multiple times. c. In the **Update schedule** dialog box, manage the recipients of the Insight. **For example:** Click Add, and in the **New recipient** dialog box, specify the user that receives your Insight. d. Click Submit. **Result:** The system displays a prompt with confirmation that the Insight is scheduled.
Delete schedules	In the **Active schedules** section, next to the schedule that you want to delete, click the Remove icon.
Add a new schedule	For more information, see Scheduling Insights.

©2025 Pegasystems Inc

For example:

The following figure shows a schedule manager that enables you to modify active schedules and add new ones:



Sample schedule manager configuration

Troubleshooting Insights issues

Ensure that Insights work as intended by learning about potential issues and their resolutions.

- **Login prompt appears too often**
- **Removing the thousands separator from columns with numbers**
- **Renaming Explore Data objects**
- **Troubleshooting missing columns in Explore Data**
- **Troubleshooting missing data objects in Explore Data**
- **Troubleshooting Portal issues**

©2025 Pegasystems Inc

Login prompt appears too often

If the value of the access token lifetime is inappropriate, the system might frequently prompt you to login. To resolve the problem, change the value according to your requirements.

Managing the access token lifetime

1. In the navigation pane of Dev Studio, click **Records**.
2. Expand the **Security** category, and then click **OAuth 2.0 Client Registration**.
3. In the list of Rules, select the OAuth 2.0 Client Registration Rule generated for your application.

   > **For example:**
   >
   > Select **PegaApp_Ubank**.

4. Manage the access token lifetime:
   a. In the **Token management** section of the Rule, in the **Access token lifetime (in seconds)** field, specify the appropriate token lifetime.

   > **NOTE:** Ensure that the value is high enough so that slower performing data queries do not run longer than the specified limit. Otherwise, the login prompt might display on long-running data queries.

5. In the upper-right corner of the Rule, click **Save**.

Removing the thousands separator from columns with numbers

By default, tables in insights display numbers with separators, which may mislead some users. Modify the user interface controls to remove the separators.

©2025 Pegasystems Inc

Customizing the user interface controls to remove separators

Cloning the `pxInteger` control

1. In the navigation pane of Dev Studio, click **Records**.
2. Expand the **User Interface** category, and then click **Control**.
3. In the list of controls, select **pxInteger**.
4. In the upper-right corner of the control, click **Save as**, and save the control as `NumberWithoutSeparator`.

Customizing the table

5. In the navigation pane of Dev Studio, click **App**.
6. On the **Classes** tab, expand the category for your application.
7. Click **Data Model** > **Property**, and then click the user interface property that you want to edit.
8. In the **Edit property** pane, in the **Display and validation** section, in the **UI control** field, enter `NumberWithoutSeparator`.
9. In the upper-right corner of the property, click **Save**.

Renaming Explore Data objects

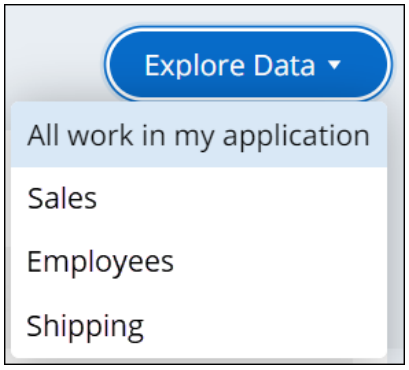
Pega Platform™ sources the names of objects that you can select in the **Explore Data** list from the short description of a certain class. Depending on your business requirements, you might want to change the name that **Explore Data** displays.

Remedy

1. In the navigation pane of Dev Studio, click **Records**.
2. Expand the **SysAdmin** category, and then click **Class**.
3. In the list of classes, select the class that has the label you want to edit.
4. At the top of the Rule, click the class label, and then enter a new name.
5. Click **Submit**.
6. In the upper-right corner of the Rule, click **Save**.

©2025 Pegasystems Inc

The following extract from **Explore Data** shows a list with custom objects (classes) that might be useful in a housing application:



Custom drop-down list in Explore Data

Troubleshooting missing columns in Explore Data

In specific scenarios, the Explore Data functionality may not include all columns with your data.

Cause: Properties not optimized

Issues with columns in the Explore Data functionality might be caused by the lack of optimization of properties.

Solution: Optimizing certain properties

If the specific column is not present in the Show/hide columns list, optimize the backing property that the column uses for reporting.

> **NOTE:** The property requires optimization in the parent classes, and in the specific Class that you view in Explore Data.

For more information, see Optimizing a property from the user interface.

©2025 Pegasystems Inc

Cause: Properties not marked as relevant records

If the specific column is not visible in the Explore Data, the corresponding property may not be marked as a relevant record.

Solution: Marking properties as relevant records

1. In the navigation pane of Dev Studio, click **Records** > **Data Model** > **Property**.
2. In the list of properties, select the property that you want to edit.
3. In the upper-right corner of the rule form, click **Actions** > **Mark as relevant record**.
4. In the upper-right corner of the rule form, click **Save**.

Cause: Cached properties

Web browsers may cache an out-of-date version of a webpage, which might cause new updates to not appear.

Solution: Clearing browser cache

Clear your browser's cache by performing one of the following actions:

- Close all open pages of the application.
- Open the application in an incognito window.
- Manually clear browser's session storage, and then reload the page.

Troubleshooting missing data objects in Explore Data

Provide users with necessary data by troubleshooting issues with missing object types in the Explore Data functionality.

Cause: Improper Data Page settings

The Explore Data drop-down list displays Data Pages that meet the following basic criteria:

- The structure is set to **List**.

©2025 Pegasystems Inc

- Data is sourced from a report definition.
- No parameters set.
- The **Allow querying any column** checkbox is selected.
- No response Data Transform applied.

> **NOTE:** Depending on the Pega Platform™ release, your Data Page might need to meet additional requirements:
>
> - In Pega Platform version 8.7, the system does not display multiple valid Data Pages that belong to the same class. Instead, the system selects just one which matches the pattern *D_ClassSuffixList* such as *D_EmployeeList* for the class *Org-App-Work-Employee*.
> - In Pega Platform version 8.8 and higher, the system follows the same rules as in the version 8.7. Additionally, you can specify data objects in UI.
>
> For more information, see Specifying objects in Explore Data.

If your Data Page is valid, but still does not appear in the dropdown, consider the following options:

- Ensure that the *Index-DPParameters* declare index is populated for your Data Page.
- Ensure that the *Index-DataSources* declare index is populated for your Data Page.
- Ensure that the number of results does not exceed the number specified in the *pyGetDataPagesForInsightSource* report definition on the **Data Access** tab. Otherwise, the system limits the number of results.

Solution: Customizing the Data Page

1. In the navigation pane of Dev Studio, click **Records**.
2. Expand the **Data Model** category, and then click **Data Page**.
3. In the list of Data Pages, select the Data Page that you want to customize.
4. Customize the Data Page to meet the requirements.

©2025 Pegasystems Inc

For more information, see Editing data pages.

Cause: Uncleared cache

The Explore Data page relies on certain caches for performance. To ensure optimal experience, clear the caches.

Solution: Clearing cache

1. In the navigation pane of App Studio, click **Explore Data**.
2. In the upper-right corner of Explore Data, click **Actions** > **Clear cache**.

Cause: Network failure

The system displays HTTP 400 error codes

Solution: Verifying the correct application alias

1. In the header of Dev Studio, click the name of the application, and then click **Definition**.
2. At the bottom of the page, in the **Application URL alias** section, click **Change URL alias**.
3. In the **Change URL alias** modal dialog box, verify that the alias is the same as the one in the application URL.
4. If the alias is different, in the text field, enter the appropriate alias.

> **NOTE:** Do not use special characters such as hyphens or underscores.

5. Click **Submit**.
6. In the upper-right corner of the screen, click **Save**.

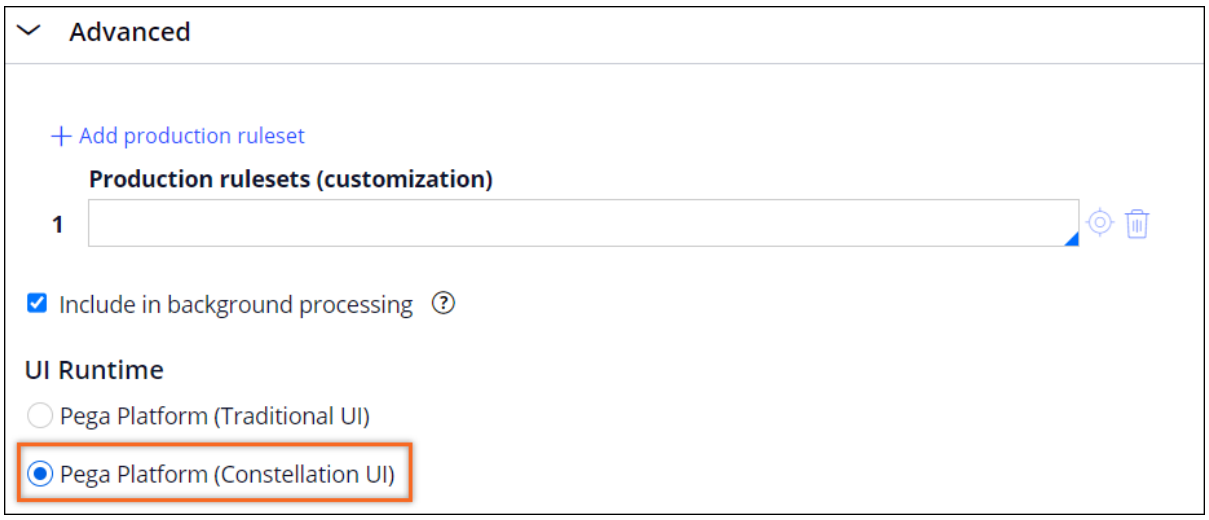
©2025 Pegasystems Inc

Troubleshooting Portal issues

You might encounter issues with the loading of your Constellation Portals or Explore Data page. To solve the issue, ensure that your application customizations are correct.

Solution: Verifying application settings

1. In the header of Dev Studio, click the name of the application, and then click Definition.
2. Expand the Advanced section, and then in the **UI Runtime** section, select the appropriate option:
   - To use server-rendered UI, select Pega Platform (Traditional UI).
   - To use fully React-based UI, select Pega Platform (Constellation UI).
3. Click Generate routing table, and then click Save.



Enabling Constellation *in Dev Studio*

> **Result:**
>
> The Constellation UI environment is now active in your application.

4. In the upper-right corner of the Rule, click Save.

©2025 Pegasystems Inc

Solution: Verifying the correct application alias

1. In the header of Dev Studio, click the name of the application, and then click Definition.
2. In the Application URL alias section, click Change URL alias.
3. In the **Change URL alias** dialog box, verify that the alias is the same as the one in the application URL.
4. If the alias is different, in the text field, enter the appropriate alias.

> **NOTE:** Do not use special characters such as hyphens or underscores.

5. Click Submit.
6. In the upper-right corner of the screen, click Save.

Solution: Setting appropriate authentication types

1. In the navigation pane of Dev Studio, click Records.
2. Expand the Integration-Resources category, and then click Service Package.
3. In the list of packages, select Application.
4. On the Context tab of the Rule, in the Authentication type list, select OAuth 2.0.
5. In the upper-right corner of the Rule, click Save.
6. Repeat steps 3 through 5 for the dev package.

Solution: Verifying the OAuth 2.0 Client Registration Rule

1. In the navigation pane of Dev Studio, click Records.
2. Expand the Security category, and then click OAuth 2.0 Client Registration.
3. In the list of Rules, ensure that the OAuth 2.0 Client Registration Rule exists.

> **NOTE:** The Rule name follows the PegaApp_*ApplicationName* `pattern.`

©2025 Pegasystems Inc

Insights and Reporting versus external tools

Pega Platform™ provides a convenient way to retrieve application data and present the information in the form of comprehensive reports or appealing visualizations. However, some rare scenarios might require you to use external business intelligence (BI) tools for reporting purposes.

The following information might help you decide whether to use Pega Insights and Reporting capabilities or external BI tools for your reporting requirements.

Using Insights and Reporting capabilities

Use Insights and Reporting capabilities when the data that you want to analyze is contained in a Pega application. For operational applications, this approach can handle the majority of reporting requirements.

Insights offer an intuitive and prescriptive user interface that enables users to conveniently define dashboards and create comprehensive ad-hoc reports. Thanks to a wide range of options you can, for example, present data as interactive charts that you can then drill down to retrieve more detailed information. In this way, you enhance data legibility and improve the viewer's engagement. After creating and customizing the Insight, you can then reach a broader audience by sharing the Insight with specific groups and individuals.

The following video shows generating AI-powered Insight in a sample sales application at run time. In this example the first query is selected from the list of prompts generated by Pega GenAI™ and shows average sales and profits across states. The second query, is entered manually and showcases 5 bestselling subcategories of products.

For more information, see Insights and Traditional reports.

©2025 Pegasystems Inc

Using non-Pega reporting tools

In some scenarios, clients might prefer non-Pega reporting tools. This is typically the case when Pega does not provide the primary work system of record, when integration with multiple non-Pega data sources is necessary, or when advanced data visualization or reporting requirements exceed the Pega aplication's current capabilities.

The following list features specific scenarios where using external BI tools might be preferred:

- Large-volume data sets, especially those requiring joins across data from both Pega and non-Pega sources.
- Advanced geo-spatial data analysis.
- Sophisticated visual analysis and querying.

Extracting data from Pega applications

In certain situations, you might need to export Pega operational workflow data to external data stores. The following list includes solutions that Pega supports to facilitate data extraction:

- For extraction of small-volume ad-hoc data, you can use the Insight Export to Excel feature (requires Insights adoption).
- For extraction of mid-scale ad-hoc and programmatic batch data, you can use BIX.
- For extraction of high volume data, you can license new CDC functionality (real-time, basic), or Data Flows (batch or real-time with some transformation and enrichment) to extract data to the target of choice.

For more information, see Data extraction.

Dashboards

Pega Platform™ provides you with a convenient way to create dashboards that include meaningful Insights for your application. You can set a custom dashboard as a landing page for your application to promote Insights to a broader audience.

PEGA ©2025 Pegasystems Inc

For example, to improve the process of data analysis, in a sales application you can create a new dashboard that features all revenue-related Insights in one place.

> **NOTE:** Custom dashboards are contained within the **Dashboards** landing page. For more information, see Organizing the main navigation for a standard work Portal.

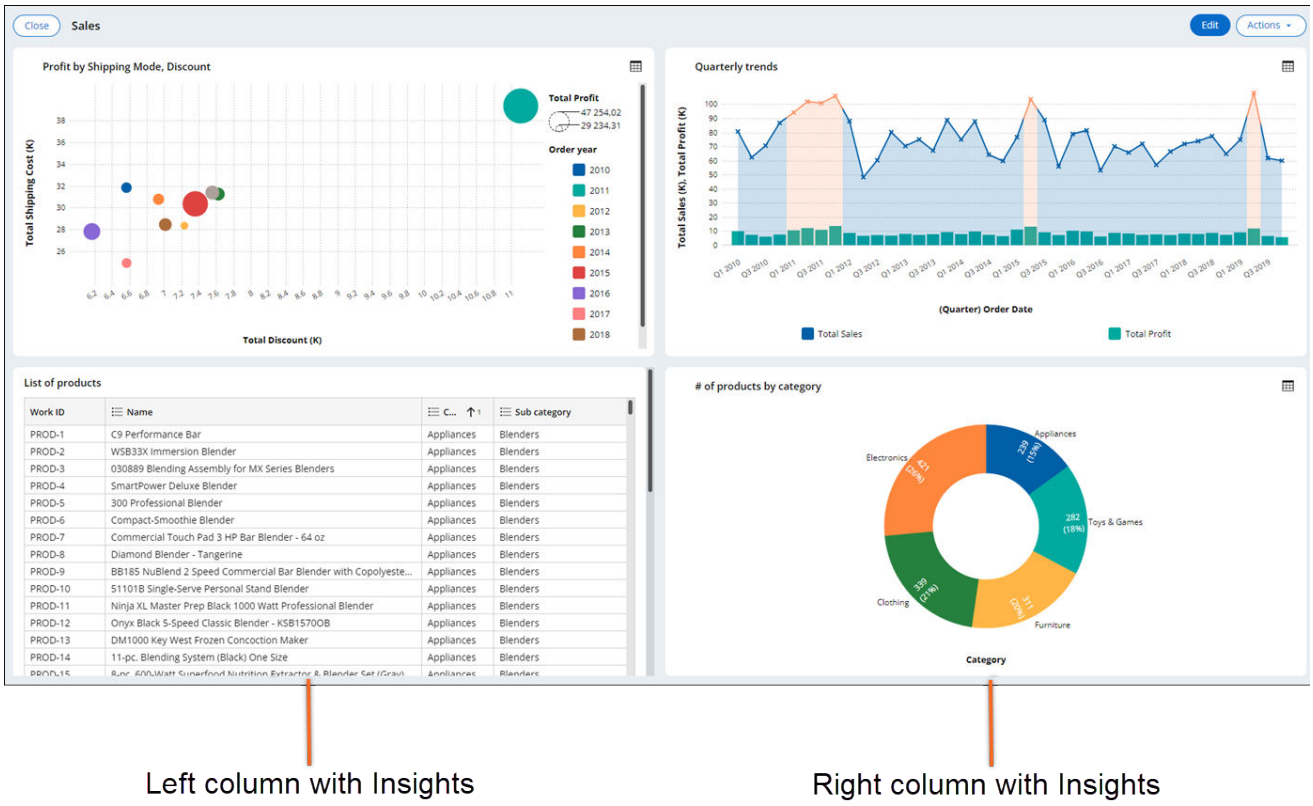
Dashboards include templates that help you organize multiple Insights and create visually compelling content. For example, you can customize a dashboard to feature two columns with Insights to improve the visibility of visualizations. By default, the **Dashboards** landing page shows a table with all dashboards that are available to you. In this way, you can efortlessly access the specific dashboard you need at the moment.

However, you can set one dashboard as the default for convenient access to the required information. For example, you can set the custom *Sales* dashboard to retrieve data about orders right after clicking **Dashboards** in the navigation pane.

For more information, see Creating dashboards.

The following figure shows a sample *Sales* dashboard that uses a two column template:

©2025 Pegasystems Inc



Dashboard with Insights

- **Creating dashboards**

Creating dashboards

Present relevant application data more conveniently by organizing Insights into readable dashboards. In this way, you ensure that users can effortlessly access information that is meaningful to them.

For example, in a sales application, you can create a dashboard that features Insights with data about discounts, orders and customers to streamline data analysis.

©2025 Pegasystems Inc

Before you begin:

Ensure that operators who access your application have the **Allow rule check out** checkbox selected to enable them to save dashboards. For more information, see Defining security information for an operator.

1. In the navigation pane of your application, click **Dashboards**.
2. In the upper-right corner of the dashboards landing page, click **Create dashboard**.
3. Select a template for your dashboard:
   a. In the **Dashboard** section, in the **Template** pane, click **Edit**.
   b. In the **Edit template** dialog box, select the template that you want to use, and then click **Submit**.

   For example:

   Select **Three column**.

4. In the **Visibility** list, select who can see your dashboard.

   For example:

   Select **Shared** to share the dashboard with your Access Group.

5. Add Insights to your dashboard:
   a. In the **Region A** section, click **Add**.
   b. In the list of Insights, select the Insights that you want to feature in your dashboard, and then click **Add**.
   c. Repeat 5.a to 5.b for all regions (columns) of your template.
6. **Optional:** To apply promoted filters, perform the following actions:
   a. In the **Filter position** list, select the location for the filters.
   b. In the **Filters** section, click **Add**, select the filters, and then click **Add**.

©2025 Pegasystems Inc

Result:

The promoted filters appear in the selected location.

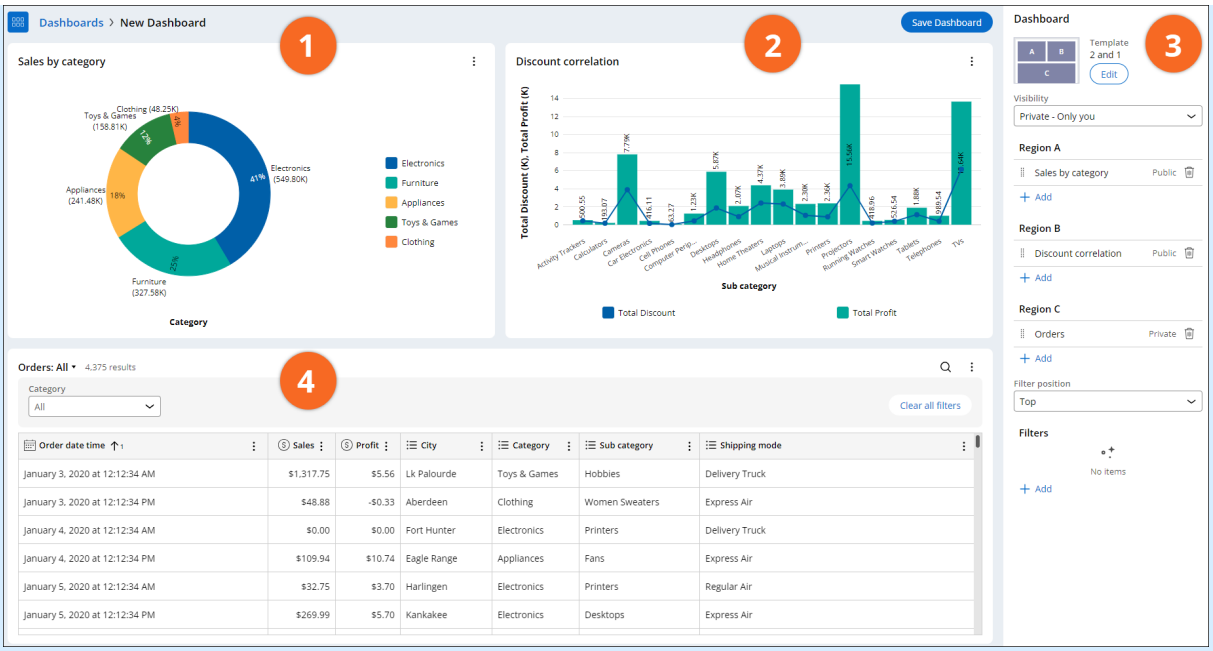
7. At the top of the landing page, save your dashboard:
   a. Click **Save dashboard**.
   b. In the **Save dashboard** dialog box, enter a name and description of the dashboard.
   c. **Optional:** To show the dashboard after you click **dashboards** in the navigation pane, select the **Set as default dashboard** checkbox.
   d. Click **Submit**.

For example:

The following figure shows customization of a sample dashboard at run time, which consists of the following areas:

1. Chart-based Insight with sales data.
2. Chart-based Insight with discount data.
3. Customization panel.
4. Table-based Insight with orders.

©2025 Pegasystems Inc



Custom dashboard

Traditional reports

Pega Platform™ helps you present application data in the form of comprehensive reports to make data analysis more convenient. You can show meaningful information as extensive list reports that feature data in columns or as summarized reports that provide statistics with charts.

For example, in a retail application, you can create a detailed financial report to analyze a company's revenues and costs. As a result, you remain up-to-date with crucial data about the financial situation of the company.

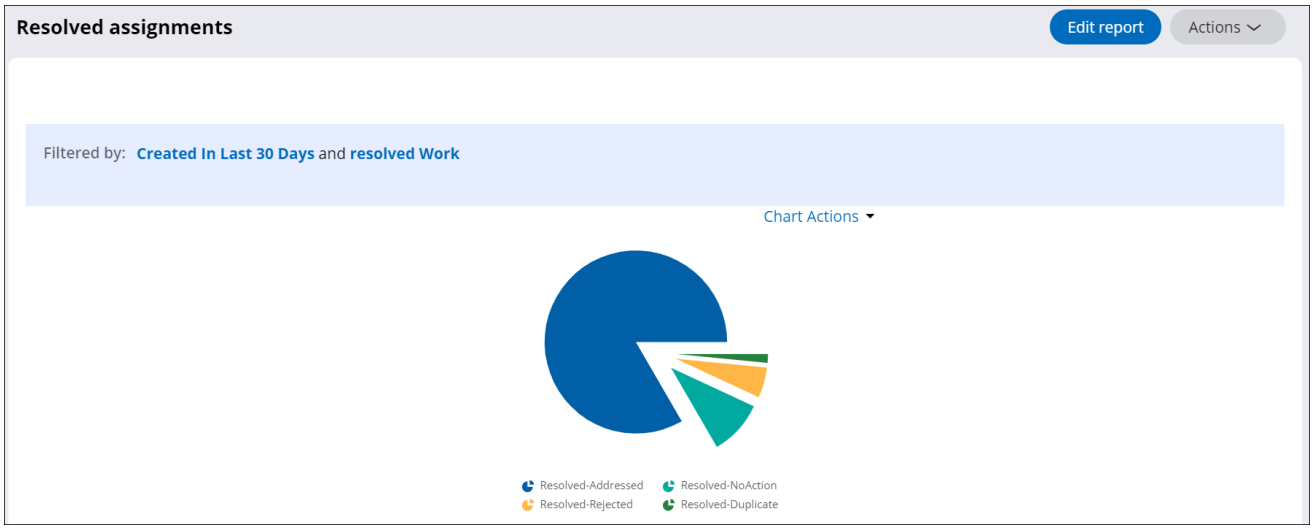
In Pega Platform, you can customize each report with various options such as sorting, grouping, or filtering. This way, you can effortlessly rebuild a report in seconds to adjust its content to constantly changing business requirements.

> **NOTE:** Traditional reports are not supported by Constellation.

©2025 Pegasystems Inc

The following figure shows a sample summarized report about assignments that have been resolved within the last 30 days. The report includes a pie chart to help users conveniently analyze the results.

> **NOTE:** Unlike Insights, charts in summarized reports are static, and you cannot drill down to subsequent data series.



Summarized report with chart

To learn more about creating and managing traditional reports, see the following articles:

- **Report Browser**
- **Creating a report**
- **Running a report**
- **Editing a report**
- **Creating advanced reports**

©2025 Pegasystems Inc

Report Browser

Report Browser is a tool that provides you with quick access to all reports that are available to you. For example, when searching for financial reports, you can save time by selecting a dedicated category that stores such reports.

Report Browser includes report shortcuts that act as links to specific reports. Report shortcuts are grouped into report categories to ensure effective report management, and to maintain readability.

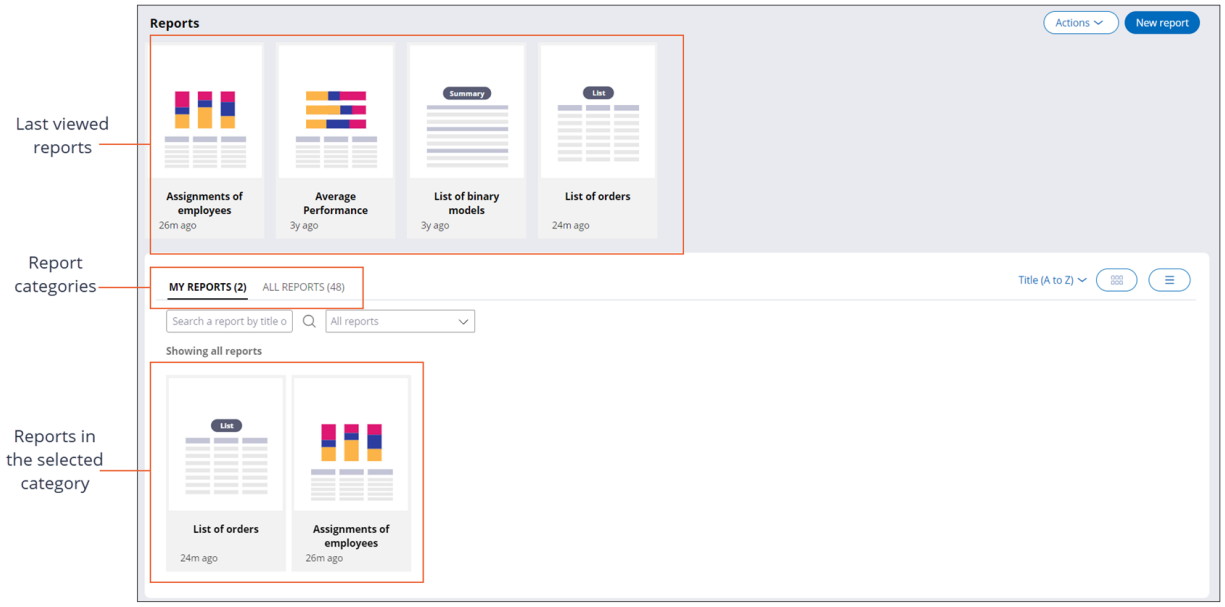
> **NOTE:** Open your Access Group rule to verify if it supports Report Browser. Otherwise, add the portal to your Access Group to make the Report Browser available in your application.
>
> For more information, see Granting Portal access to an Access Group.

Report Browser features

By accessing Report Browser, you can perform the following actions:

- Run reports.
- Create custom reports and modify copies of the existing reports.
- Edit report details.
- Move reports to different categories.
- Delete reports from the system.
- Search, filter, and sort reports.
- Switch between your private reports and all other available reports.
- Share reports with others.
- Schedule reports to run at a certain time, or on a recurring basis at regular intervals.
- Subscribe to notifications with the results of scheduled reports.

PEGA ©2025 Pegasystems Inc



Report Browser tool

- **Report types**
- **Report categories**
- Running a report
- Creating a report
- Editing a report
- Scheduling reports
- Managing report subscriptions
- Report Viewer

Report types

Pega Platform™ provides you with the option to analyze data from different perspectives depending on the report type that you use. You can monitor detailed lists of events that occur in your application, or view summarized numbers of the occurrences.For example, you can create a report that lists the number of cases that users resolve within the last seven days, and analyze the report to assess user performance.

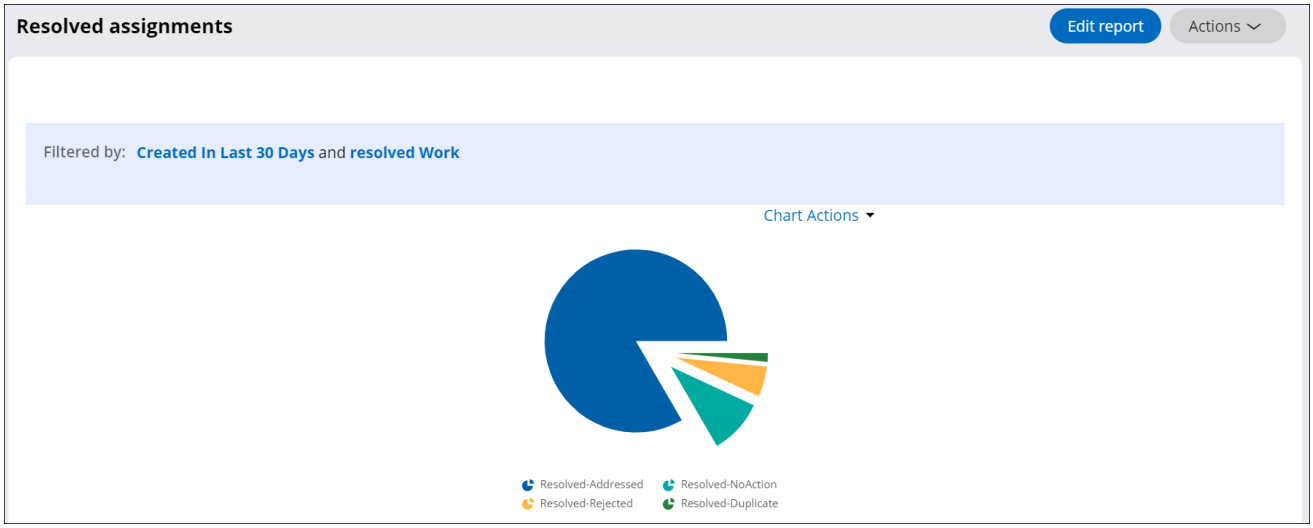
©2025 Pegasystems Inc

You can create two types of reports: summarized reports, and list reports.

Summarized reports

Summarized reports help you discover the number of specific events that occur in your application. For example, you can view the total number of days that a team required to complete a task.

That type of reports provides you with several built-in functions that present data as numbers. That type of reports can include, for example, counts of cases, total amount of orders, or average elapsed time between the start and the resolution of a work item. For increased clarity, you can also include charts that graphically represent data.



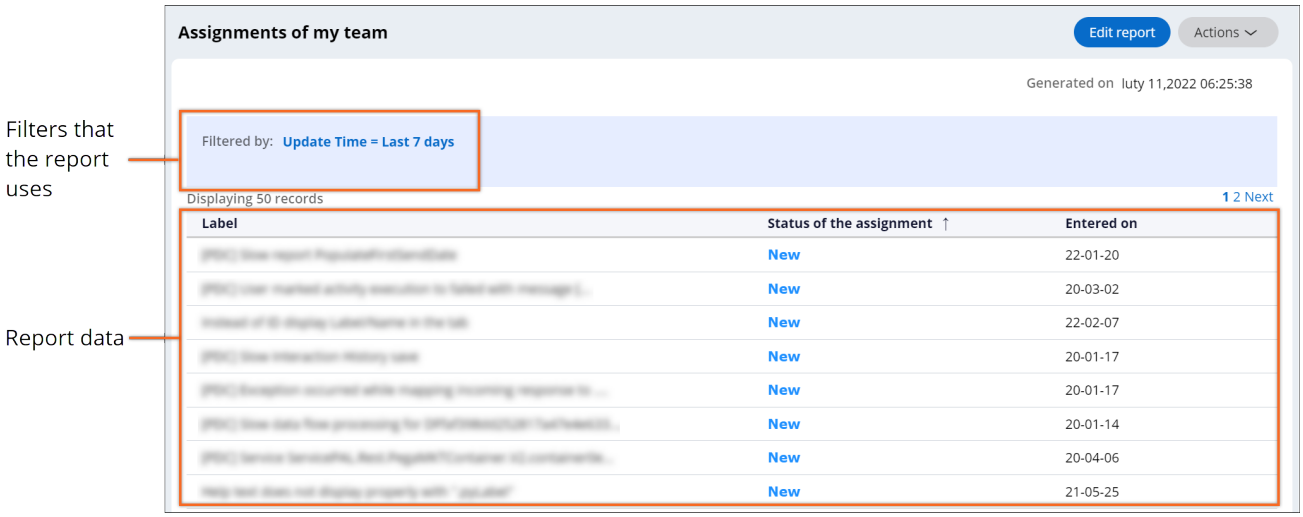
Summarized report with chart

List reports

List reports help you analyze extensive data in a clear and compact way. For example, you can create a list of all open cases to verify the remaining workload.

The reports provide detailed and comprehensive information about application data. For example, you can create a report on resolved cases that includes a column with descriptions of each case. You can then use the report for easier analysis of user performance.

©2025 Pegasystems Inc



List report

- Converting a summarized report to a list report
- Converting a list report to a summarized report

Report categories

You can distinguish reports from each other by organizing them into categories.Pega Platform provides several report categories, but you can create custom report categories. For more information, see Creating a report category in Report Browser.

Based on the purpose, you can assign reports to one of the available categories:

Analyze Performance

Provides information about resolved cases in an application at the level of each individual step, or at the level of individual actions, within a business process. The reports analyze completed work to determine whether business processes are efficient and effective. An example of the Analyze Performance report is a report that tracks processing time in hours by task and action.

Analyze Quality

Provides information about resolved cases in an application. Similar to reports that analyze performance, quality reports analyze completed work to determine if business processes are effective. An example of the Analyze Quality report is a

©2025 Pegasystems Inc

report that displays the average time a work item is assigned a certain status, like Open or New.

Case Metrics

Provides information about the number of cases that have been created each day for the last seven days, and the time per stage for resolved cases.

Monitor Assignments

Provides information about assignments for open or unresolved cases in an application. The report tracks the work based on the assigned user. An example of the Monitor Assignments report is a report that measures timeliness by task.

Monitor Processes

Provides information about assignments for open or unresolved cases. In comparison to the Monitor Assignments reports, the report focuses on the work rather than on the users. An example of the Monitor Processes report is a report that measures throughput in the past week by work type.

Open Cases

Provides information on the timeliness of a case from the time a case is created to the time the case is resolved.

Predictive Model Monitoring

Monitors the performance of the predictive models to detect when they stop making accurate predictions, and to re-create or adjust the models for better business results, such as higher accept rates or decreased customer churn.

Service Level Performance

Provides information about service-level-agreement (SLA) status that is grouped by assignment or by operator.

Step Performance

Provides information about service-level-agreement (SLA) status that is grouped by assignment, and a historical view of the time the user spends to complete a task.

- **Creating a report category in Report Browser**

©2025 Pegasystems Inc

Creating a report category in Report Browser

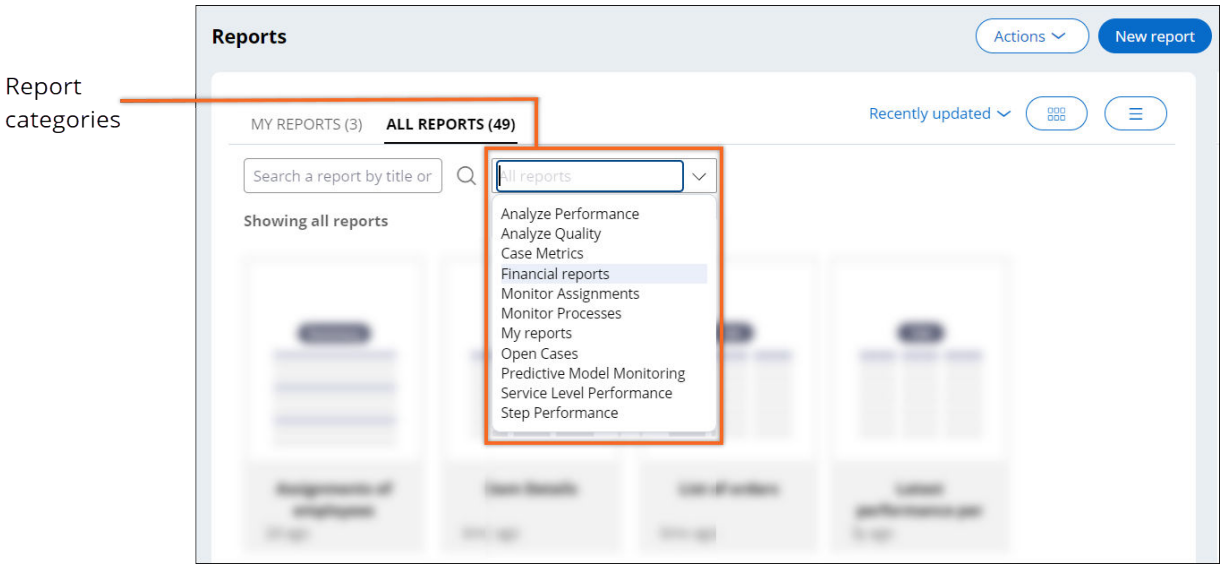
Save time on searching for certain reports by organizing them into custom categories. For example, you can create a category for financial reports to store all important data in one place.

1. If App Studio is not in Preview mode, in the header, click **Preview application**.
2. In the header of App Studio, select the portal in which you manage reports.
3. In the left navigation pane, open the Report Browser by clicking **Reports**.
4. Click **Actions** > **Manage Categories**.
5. Click **New Category**.
6. In the **Create new category** modal dialog box, enter the category details:
   a. In the **Category name** field, enter the report category name.

   For example:

   Enter `Financial reports`.

   a. In the **Category description** field, enter the report category description.
   b. In the **Category type** list, select whether the report is private or public.
   c. Click **Submit**.



©2025 Pegasystems Inc

Report categories in Report Browser

- Managing reports shortcuts

Creating a report

Achieve greater precision and flexibility in data retrieval and analysis by creating custom reports. For example, you can create a comprehensive list report that includes detailed information about company sales within specific period to analyze the performance of salesmen.

1. In the header of App Studio, select the portal in which you manage reports.
2. In the navigation pane, open Report Browser by clicking **Reports**.
3. In the upper-right corner of Report Browser, click **New report**.
4. In the **Create new report** modal dialog box, specify the report details:
   a. In the **Case type** list, select the primary case type or data type on which you want to report.
   b. In the **Report type** list, select the type of report.
   c. Click **Submit**.

   > **Result:**
   >
   > The system opens Report Editor where you can manage the report.

5. In Report Editor, modify the report to meet your business needs.

   > **For example:**
   >
   > From the **Data Explorer** section, drag specific records to your report to include them in the report.

6. In the upper right corner of Report Editor, click **Done editing**.
7. In the **Save report as** modal dialog box, enter the report details:
   a. In the **Title** field, enter the report title.

©2025 Pegasystems Inc

b. In the **Description** field, enter the report description.
c. In the **Category** list, select the report category.

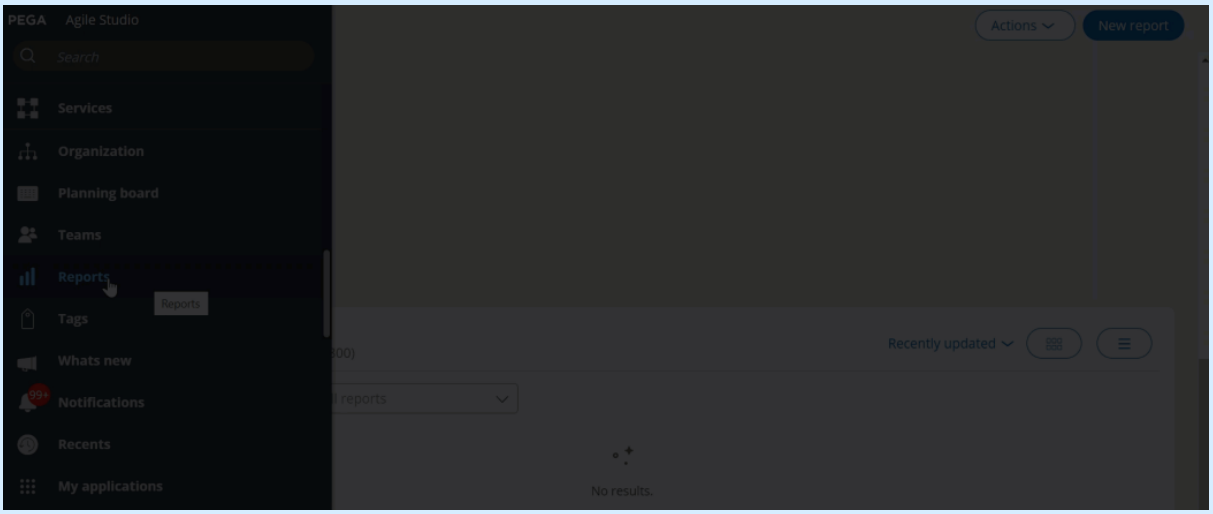
8. Click **Submit**.

Result:

The report opens in Report Viewer, and shortcut of the report becomes available in the selected category.

For example:

The following visualization presents how to create and manage reports in App Studio.



Creating custom reports

- **Best practices for creating reports**
- **Adding columns to reports**
- **Duplicating existing reports**
- Running a report
- Editing a report
- Adding columns to reports

©2025 Pegasystems Inc

- Managing charts in Report Editor

Best practices for creating reports

When creating reports in your application, follow best practices to avoid system failures that result from retrieving excessive data in a report. Ensure that your reports limit the number of unnecessary records that might significantly slow down your system or lead to errors.Ensure an effective and smooth user experience by, for example, filtering records by date to limit the amount of data. In this way, you control reports, and improve the performance of your system.
Pega Platform™ does not support partitions out of the box. However, you can boost the database query time by following best practices for creating reports:

- Specify the number of records that your reports retrieve from the database.
- Enable paging in reports.
- Filter data that your reports display.
- Optimize the database.

> **NOTE:** As an application developer, in Dev Studio, you can manage filtering options in the *pyDefaultReport* that users use as a template for custom reports. Ensure that you create the default report with the proper filters in the base Class record of the client. Otherwise, each custom report based on the default report includes inappropriate settings, which results in retrieval of an enormous number of records that slow down the system, and can eventually cause system errors.

Managing report settings

Improve the performance of your reports by customizing report settings. For example, you can boost your reports by limiting the number of records that the report retrieves from the database.

©2025 Pegasystems Inc

1. In the navigation pane of Dev Studio, click **Records**, and then click **Reports** > **Report definition**.
2. In the list of report definitions, click the report definition that you want to edit.
3. Enable paging in the report:
   a. Click the **Report viewer** tab.
   b. In the **User actions** pane, select the **Enable paging** checkbox.
4. Specify the maximum number of records that the report retrieves:
   a. Click the **Data access** tab.
   b. In the **General data access settings** section, in the **Maximum number of rows to retrieve** field, set the value to a number that is lower than 10000.

   > **IMPORTANT:** Values higher than 10000 may lead to performance issues.

5. In the upper-right corner of the report, click **Save**.

Filtering report data

Limit data that your report displays by specifying custom filters. In this way, you ensure that the report runs faster, and that it includes only the data that you require.As a best practice, add date and time filters to retrieve data from only the time periods that you need to view. For example, instead of running a report for all the data in the database, restrict the data to records from the last month.

1. Run the report that you want to update.
   For more information, see Running a report.
2. In the upper-right corner of the screen, click **Edit report**.
3. In the **Data Explorer** section, add a filter condition:

©2025 Pegasystems Inc

Choices	Actions
Add a function to the filter condition	a. From the Calculations tab, drag a specific function to the **Drop column to add filter** section. b. In the **Calculation builder** dialog box, in the Function input field, press the Down arrow key, and then select a property to calculate. **NOTE:** Some functions require more than one input property, so more Function input fields are available. c. Click Submit. d. In the **Edit filter** section, in the *Function name* list of operators, select a relational operator for the filter condition. e. In the *Function name* box, press the Down arrow key, and then select a property to compare with your function. For example, you can use the Older than n days function to filter the report to display cases that remain open for a specific period of time.

©2025 Pegasystems Inc

Choices	Actions
Add a property to the filter condition	a. From the Best Bets or All Matches tabs, drag a specific property to the **Drop column to add filter** section. For more information on Data Explorer tabs, see Data Explorer. b. In the **Edit filter** section, in the *Filter condition name* list of operators, select a relational operator for the filter condition. c. In the *Filter condition name* box, enter specific value, or press the Down arrow key, and then select a property to compare with your property. d. **Optional:** To add a function with which the system compares your property, click Calculation Builder, and specify the function. For example, you can use the Day of the week function to filter the report to display all resolved cases on a particular day of the week.

4. **Optional:** To change the name of the filter condition that is visible in the Report Viewer and the Report Editor, in the Filter Caption field, enter a new name.
5. **Optional:** To use a null comparison value if a comparison value is not specified, select the Use null if empty check box.
6. **Optional:** To apply a case insensitive comparison, select the Ignore case check box.

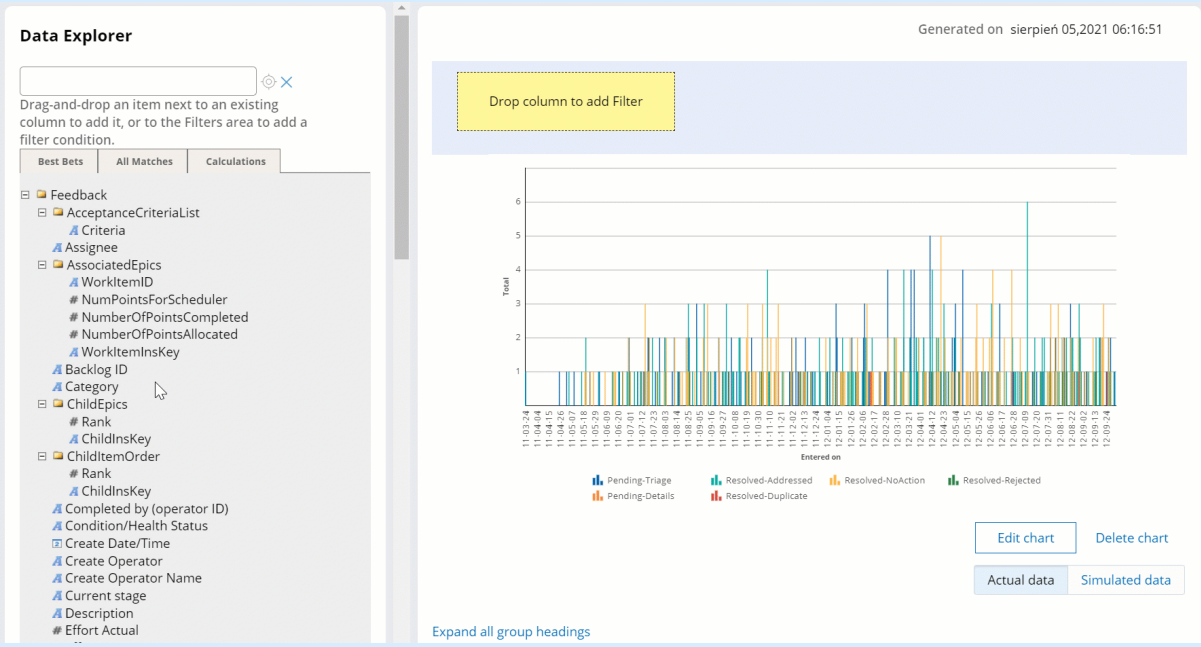
©2025 Pegasystems Inc

> **IMPORTANT:** Selecting the **Ignore case** checkbox results in adding the `UPPER` function to the `WHERE` clause of your query. As a result, it might affect the system performance.

7. Add the filter condition by clicking **Apply changes**.

For example:

The following visualization presents how to apply two filters that restrict the results to include resolved assignments that are created date prior to 2011-11-19.



Applying filters in summarized report

Optimizing database

Improve your system performance by customizing database settings. For example, you can free up database space by removing tables that are no longer in use.For more information, see Optimizing schemas.

©2025 Pegasystems Inc

1. In the header of Dev Studio, click **Configure** > **System** > **Database** > **Optimize Schema**.
2. Select the database for which you want to view optimization suggestions.
3. Select the type of suggestions you want to view.
   - **Index suggestions** – To view an application other than the current application, select it from the Filter by Application list. Click **Create** to create an index for a column.
   - **Unused tables** – Click Instance Count to view the classes that were mapped to the associated table and their individual instance counts.
   - **Unused columns** – Select whether to drop these columns from the table.
   - **Invalid Class mappings** – To view an application other than the current application, select it from the Filter by Application list.
4. Click **Proceed with Changes** to apply the selected optimizations.
5. Click **Submit**.

Efficient use of reporting

Reporting is a powerful functionality with numerous tools that you can use to retrieve specific application data. Learn how to manage report definition rules to improve system performance and use the functionality efficiently.

Choosing correct database index columns

As a best practice, create database table indexes that match as many queries in your workload as possible, because that approach reduces the overall amount of indexes and improves system performance. Efficient reporting depends on the uniqueness of the combination (cardinality) of the columns' values involved in the query, `WHERE` clause, or filters in report definition.

The following is an example of the query:

```
SELECT firstname, lastname FROM students WHERE street='XXX' AND city='XXX'
```

©2025 Pegasystems Inc

In the above example, the recommended choice is a multi-column index From the above example multi-column index should be `street AND city`, but not `city AND street`.
The following list includes other important recommendations regarding the index columns:

- When choosing the indexes, consider the effect on insert, delete and update operations. More indexes means slower operations, because indexes require reorganization for each operation we perform on the table.
- Choose B-tree indexing for <,>,< >,=,!=,>=, and <= operations.
- Choose hash-based index only if != and = operations are involved, because the hash-based index does not help with other comparison operations.
- Use Query analyzer / Explain statements to verify if the created indices are in use. If you require assistance, contact your Database Administrators for better understanding.

Choosing appropriate columns for filters

Improving the performance of reporting also depends on choosing the right column as a filter. For example, you can create a report definition that includes all resolved cases in your application. The following list includes two possible approaches to achieve the goal:

- Filter results by `pyStatusWork` column whose data starts with `Resolve`.
- Filter results by `pyResolvedTimeStamp` column and set `Is not null`.

In the case of first approach with the `pyStatusWork` column, the database might not take full advantage of the index even if it is present on `pyStatusWork`. As a result, the report definition might run slowly.

In turn, in the case of the filter with `pyResolvedTimeStamp` column, the database query runs faster than in the first approach. Index unique scan occurs, if `null` is considered as part of the index.

©2025 Pegasystems Inc

Enabling pagination

Report definition provides the pagination option that allows the server to improve performance and memory use when loading the results. You can use pagination by accessing your Data Page, and checking the **Enable pagination** check box. Selecting this option improves Data Page memory use and performance for data pages that are used as the source of a grid. The pagination option propagates the settings from grid to Data Page, and then to report definition.

> **NOTE:** The **Enable pagination** check box appears when you set the following parameters on the **Definition** tab of the Data Page rule form:
>
> - Structure – List
> - Mode – Read-only
> - Scope – Thread or Requestor
> - Source – Report definition

Ensure that the report definition is in the same Class as the Data Page.

Using bulk processing

Avoid processing of large sets of results which uses a lot of system resources causing performance issues by using bulk processing. Iterative approach helps to achieve same goal without impacting system resources.Implementing iterative approach with Report definitions and Obj-Browse method improves query performance and reduces memory footprint. For example, an application user might want to update 5000 000 eligible cases with One time utility. Retrieving great number of records might degrade the application performance.

1. Define hasMore parameter and set to true before calling Obj-Browse or Report Definition.
2. Start looping by using loop option in activities, and configure precondition for that step.

©2025 Pegasystems Inc

3. Obj-browse or Report Definition should use any unique numeric property that is sorted by the ascending order.
4. You can do business logic after this step on the results set. Once business logic is done at the end, set values Param.hasMore based on pxMore from Resultset page and set last processed id.

> **NOTE:** The pxMore is set by Pega to true if there are more records left for your query. That is indication for system to continue the loop. When there are no records left, loop precondition fails, and the step is exited.

Best practices for dashboards

Pega Platform provides you with a convenient tool to create custom dashboards that present your data in a visually appealing form. In this way, you can select meaningful application data and promote it to a broader audience.

To create a valuable and informative dashboard, consider the following best practices:

Knowing your audience

Familiarize yourself with the type of audience, their intent, and finally, the context under which they consume the dashboard. Your audience might be focused on strategic areas or interested in operational aspects. A dashboard should address these needs differently. For example, a vice-president of sales might be interested in products or sales trends in a certain territory, whereas a customer service organizational leader might be interested in agent productivity. Similarly, an airline executive might look for the status and position of aircraft at any given time versus trends in the utilization of aircraft in the past few quarters.

In addition to insights, your audience might be looking for suggested actions to achieve or correct business outcomes. Different audiences operate under different contexts, hence, your dashboard might be consumed in variety of environments – ranging from a large screen in an executive boardroom to a factory floor. It might even cater to mobile

PEGA ©2025 Pegasystems Inc

audiences on the small screen of a cell phone or tablet. Most importantly, one must understand consumption patterns – whether it is presented in meetings, sent over email, or glanced at during travel. Such synchronous or asynchronous consumption may lead to various interpretations by your audiences.

Preparing a story

There is no one correct way to prepare your dashboard stories. Less is more when it comes to conveying a crisp message, hence, you should stick to the purpose of the dashboard and keep it relevant. Most dashboards generate questions like “so what” or “why” in your audience’s mind, so anticipate and address such questions in your dashboards. This is where data becomes insights. Finally, your dashboard should smoothly and appropriately present exceptions, trends, and recent effects.

Focusing on visuals

Visual representation of your story is equally important as your story. Consider the following best practices:

1. Ensure relevancy. Currency, decimals, and text are presented appropriately per context.
2. Validate accessibility. This is very important if you have a varied audience consuming a dashboard under different contexts. Size and placement of widgets, color combinations, and scrolling should ensure consistency and readability.
3. Place an emphasis on aesthetics. Selection of your widgets and color pattern should attract the attention of your audience and not distract them from the main story.

The type of visualization you chose impacts all three aspects of visual design. Bar charts give an absolute impression of quantity dimension versus line charts that implicitly depict trends over prior data points. Bubble charts give you deeper control by letting you choose the position, size, and color of the bubble. Maps make clear sense as a backdrop when you want to plot data against spatial and geographic dimensions.

PEGA ©2025 Pegasystems Inc

Consider technical design factors

The correct technical implementation ensures the smooth functioning of your dashboard. Product owners and engineers (mainly database designers) are accountable for simple and clean design. While there are a lot of factors for technical design, the key aspects to watch out for are captured below:

- **Performance**: Interactive dashboards should respond to every click in less than 3 seconds in various environments (for example, data volume or network considerations). Appropriate schema design (for example, star schema) and denormalized data structure helps reduce query load and improve performance.
- **Agility**: New demands to see data in various ways will raise the curiosity of your audience. This necessitates the need for agility in maintaining your dashboard. You should be able to add or alter new visualizations with minimum engineering or deployment effort. Your dashboard should truly be a no-code solution.
- **Data latency**: Your transactional data schema might be different from a reporting data schema (or Datamart schema). Data transformations are inevitable while copying transactional data to reporting schema. This increases the burden of data refresh timings and creates data latency. While this might not be a concern in strategic dashboards, it is certainly a showstopper in operational dashboards. Incremental event-driven refreshes, with optimized data schema, may help resolve this to a certain extent. One can choose either ETL (extract-transform-load) or ELT (extract-load-transform) to get data from transactional to mart. ELT is a modern alternative because storage and compute resources are separate.
- **Integrated experience**: Dashboard's integrated visual experience is very important in a larger enterprise environment. Your dashboards should be developed as loosely coupled components that can be seamlessly embedded into any enterprise system.

Tools selection

A lot of data visualization tools are available in market today. You should ensure that you opt for ones that support no-code dashboard creation and maintenance and offer connectivity with varied data sources.

PEGA ©2025 Pegasystems Inc

Adding columns to reports

Extend the scope of your report by adding properties or calculations to that report. For example, you can include a calculation that displays the difference in hours, days, or months between two dates of your choice.

1. Run the report that you want to update.
   For more information, see Running a report.
2. In the upper-right corner of the report, click **Edit report**, and then add columns:

Choices	Actions
Search for properties to add	a. In the **Data Explorer** section, in the search box, enter a property name, and then press Enter. b. Drag the property next to an existing report column.
Display the most suitable properties	a. In the **Data Explorer** section, click the **Best Bets** tab. b. Drag the property next to an existing report column.
Display all available properties	a. In the **Data Explorer** section, click the **All Matches** tab. b. Drag the property next to an existing report column.
Display SQL functions to use	a. In the **Data Explorer** section, click the **Calculations** tab. b. Drag the calculation next to an existing report column. c. In the **Calculation builder** dialog box, in the **Function input** field,

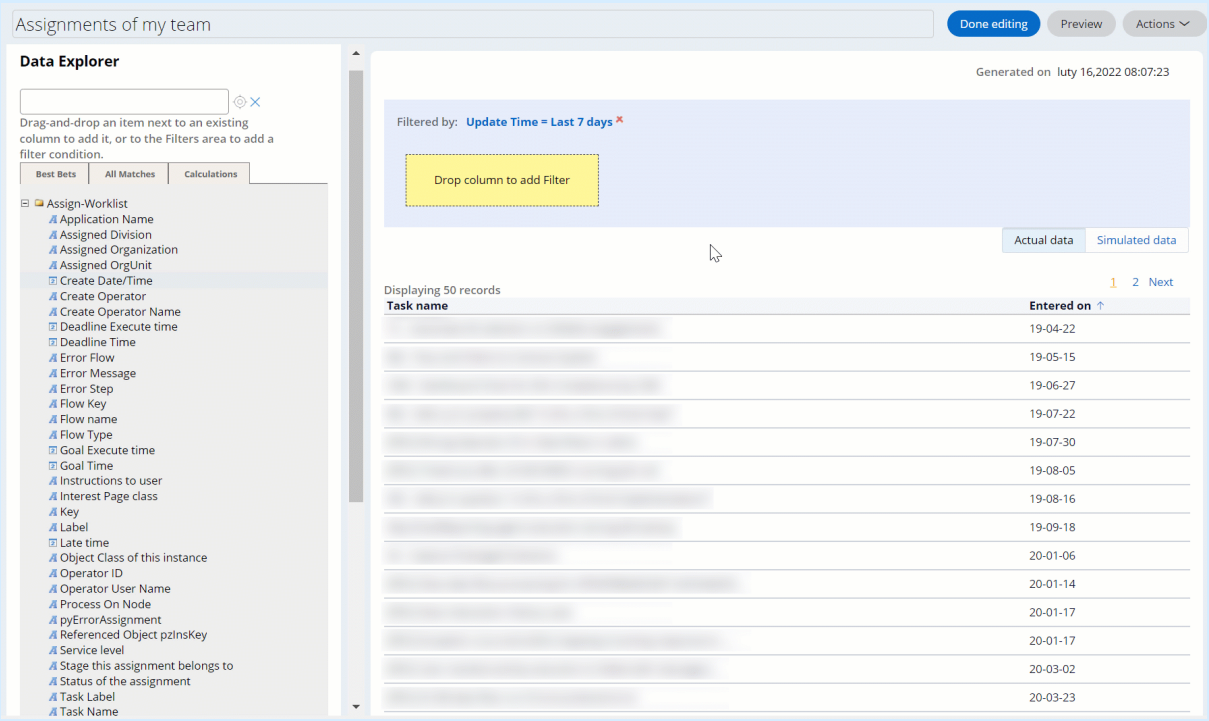
©2025 Pegasystems Inc

Choices	Actions
	press the Down arrow key, and then select the property to calculate. d. Click **Submit**.

3. In the upper-right corner of the screen, click **Done editing**.

For example:

The following visualization presents how to add custom columns to the report in Report Editor.



Adding columns to report

- **Sorting and grouping columns in list reports**
- Managing reports shortcuts
- Creating a report

©2025 Pegasystems Inc

- Editing a report

Sorting and grouping columns in list reports

Make your report more readable and organized by grouping columns and sorting the column values. For example, you can sort and group the report results to present the cases that the user accessed most recently.

1. If App Studio is not in Preview mode, in the header, click **Preview application**.
2. In the header of App Studio, select the Portal in which you manage reports.
3. In the navigation pane, open the Report Browser by clicking **Reports**.
4. Select the reports that you want to view:
   - To view your custom reports, click the **My Reports** tab.
   - To view all available reports, click the **All reports** tab.
5. From the list of reports, select the report that you want to edit.
6. In the upper-right corner of the screen, click **Edit report**.
7. In the upper-right corner of the report, click **Actions** > **Sort**.
8. In the **Sort and Group** section, manage the report columns:

Choices	Actions
Sort the column values	Next to the column name, in the **Sort** list, select a sorting method for the column. **NOTE:** You can apply different a sorting method for each column.
Group the results	a. Select the **Group results** check box. b. In the numeric list, select the number of columns to display as the group heading.

©2025 Pegasystems Inc

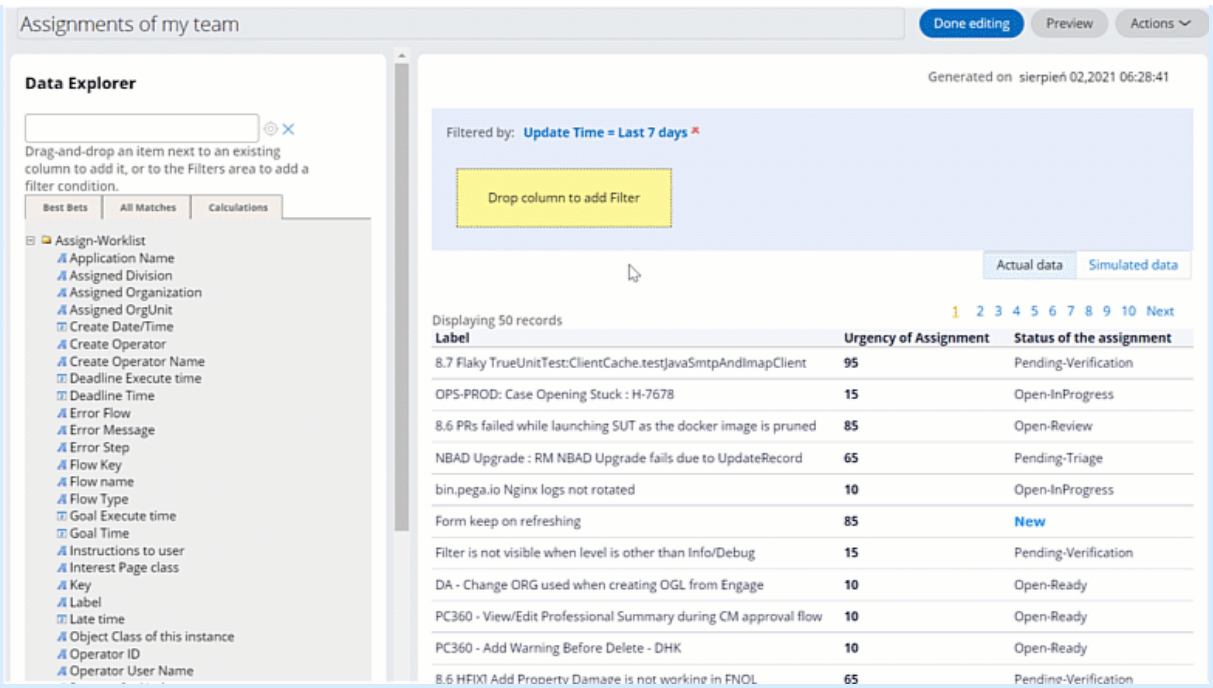
Choices	Actions
	c. **Optional:** To remove the rows that share the same values in all columns, select the **Remove duplicate rows** check box.
Limit the number of displayed rows in the report	a. In the **Top/Bottom Rank** section, in the **Display** list, select how you want to display the rows. For example, if you want to display the rows with the highest values, select **Top Ranked**. b. In the box next to the **Display** list, enter the maximum number of rows to include in the report. c. In the **Rows** list, select where you want to limit the rows. For example, if you want to limit the overall number of rows in the entire report, select **Overall**. d. In the **based on** field, press the Down arrow key, and then select a property to use.

9. Click **Apply changes**.

For example:

The following visualization presents how to limit the number of rows in the table by specifying certain conditions. The system retrieves five rows with the highest values in the **Urgency of assignment** property.

©2025 Pegasystems Inc



Sorting columns in list reports

- Editing a report
- Creating a report

Duplicating existing reports

Safely modify report data without affecting the source report by duplicating that report. For example, you can use the data from the original report, but filter or organize the information differently.

1. If App Studio is not in Preview mode, in the header, click **Preview application**.
2. In the header of App Studio, select the portal in which you manage reports.
3. In the header of Dev Studio, select the portal in which you manage reports.
4. In the left navigation pane, open the Report Browser by clicking **Reports**.
5. Run the report that you want to duplicate.
6. In the upper-right corner, click **Actions** > **Save as**.
7. In the **Save report as** modal dialog box, enter details for the duplicated report:
   a. In the **Title** field, enter the report title.

©2025 Pegasystems Inc

You can specify the same title for multiple reports that belong to different categories.

b. **Optional:** To give the duplicated report a description, in the **Description** field, enter a summary of the report.
c. In the **Category** list, select a category for the report.
d. Click **Submit**.

> **Result:**
>
> The duplicated report opens in the Report Editor, and the report shortcut becomes available in the Report Browser.

- Running a report
- Editing a report

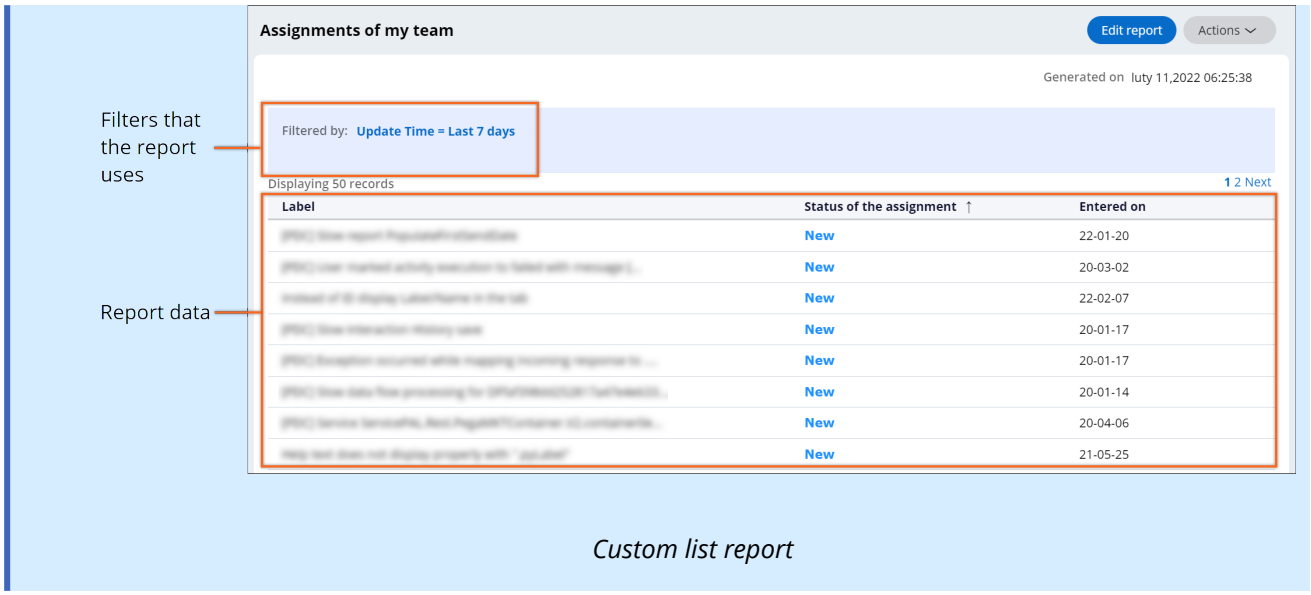
Running a report

Retrieve valuable information about your application by running comprehensive reports. You can analyze data from the reports to gain knowledge about your business.For example, you can run a financial report from previous quarter to analyze credit rating of your company.

1. In the header of App Studio, select the Portal in which you manage reports.
2. In the navigation pane, open the Report Browser by clicking **Reports**.
3. Select the reports that you want to view:
   - To view your custom reports, click the **My Reports** tab.
   - To view all available reports, click the **All reports** tab.
4. Click a report that you want to view.

> **Result:**
>
> The report opens in Report Viewer.

©2025 Pegasystems Inc



Custom list report

- **Report Viewer**
- **Reports run on schedule**
- **Scheduling reports**

- Scheduling reports
- Insights and Reporting
- Duplicating existing reports

Report Viewer

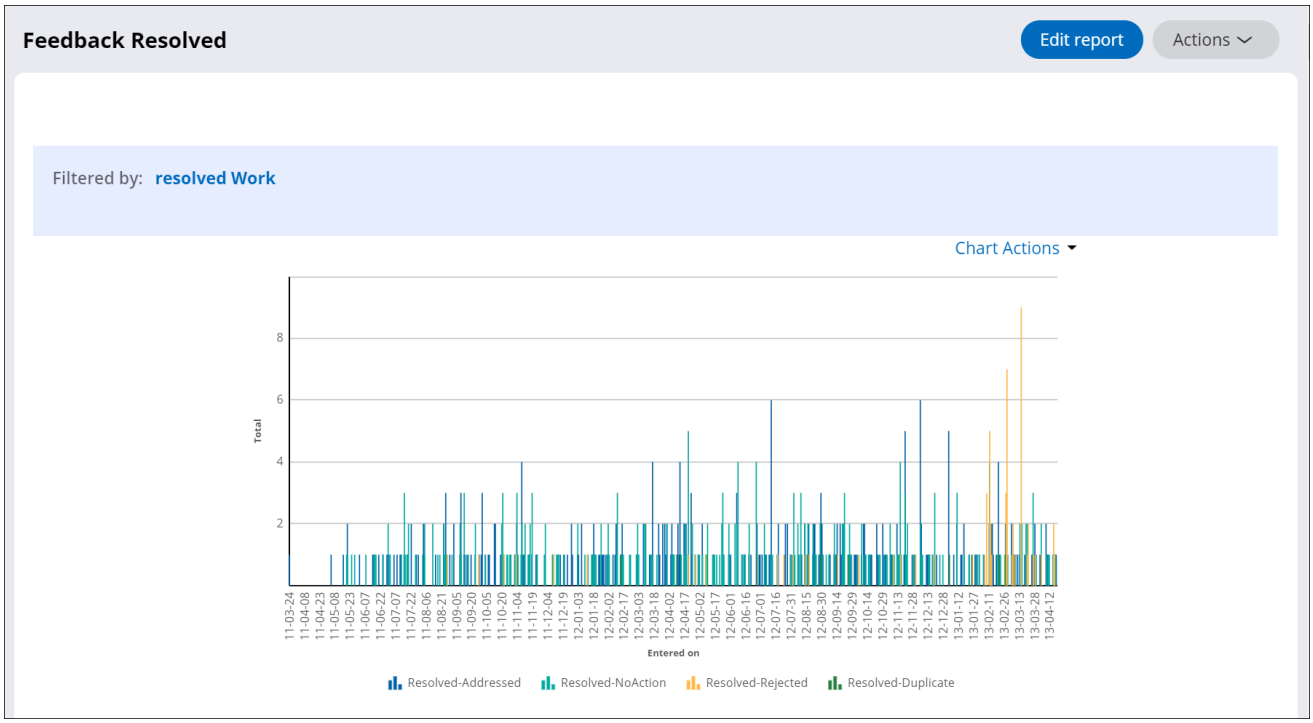
Report Viewer displays report data in a clear and comprehensive way, and helps you to analyze that data. For example, you can access a report that lists bugs in your application, and then analyze which areas of the application require improvement.The Report Viewer opens when you run a report in the Report Browser. Depending on the report type, the Report Viewer displays data in the form of detailed lists of the events that occur in your application, or as summarized numbers of occurrences.

Report Viewer features

By accessing the Report Viewer, you can perform the following actions:

©2025 Pegasystems Inc

- View the report results.
- Sort the report results.
- Access the Report Editor to manage the report results.
- View and manage filters.
- Convert the report to a summarized or list report.
- Export the report to a PDF or Excel file.
- Save a copy of the report.
- View and interact with chart data.
- Summarized reports only: drill down to view detailed information about a row or cell.
- List reports only: filter the report results.



Report Viewer tool that displays an exemplary report

- Report types
- Report Browser
- Report Editor
- Editing a report
- Converting a list report to a summarized report

©2025 Pegasystems Inc

- Converting a summarized report to a list report

Reports run on schedule

Pega Platform™ provides a functionality to schedule a report to run on a specific date and time. For example, you can retrieve data about bugs resolved within a specific time slot to analyze productivity of the team.You can schedule a report to run by completing a schedule task form that consists of the following sections:

Task Definition

Task Name

Label of the scheduled task. Supports only letters, numbers, and the underscore character.

Task Description

Short summary of the report.

Task Scheduling

Time of Day

Select what time the report runs. The time corresponds to the time zone set for the server that runs the Pega Platform.

Recurrence – One-Time only

Run the report once. Specify the date in the **on** field.

Recurrence – Recurring task

Run the report at specific intervals. In the **Recurrency information** section, select when the report run starts, and how often the report run occurs.

End date

Select when to finish the report run.

You can run the report indefinitely, specify the exact number of runs, or specify the end date.

©2025 Pegasystems Inc

On error, retry

Rerun the report if errors occur. You can set the interval and the maximum number of attempts.

Task Output Processing

Output File Format

Specify the report file format. You can choose between PDF file format, and Microsoft Excel file format.

> **NOTE:** Reports in the PDF format do not include charts.

Enable authorized users to subscribe to task notifications

Allow other users to subscribe to the report.

Send Notifications to the following users

Specify a list of users that receive a copy of the report.

Attach output to notification

Attach the report output file to the notification e-mail that subscribed users receive when the report runs.

Embed link to output in notification

Embed a link to the report output file in the notification e-mail that subscribed users receive when the report runs.

Configure Notification Message

Specify the content of the notification that the subscribed users receive when the report runs.

- Scheduling reports

©2025 Pegasystems Inc

Scheduling reports

Analyze how data changes over time by gathering data from scheduled reports. For example, you can schedule a weekly report about resolved cases, and compare the report data to analyze the productivity of your team.

> **NOTE:** By scheduling a report, you automatically subscribe to that report.

You can set a specific time, interval, and frequency of a report run.

1. In the header of App Studio, select the Portal in which you manage reports.
2. In the navigation pane, open the Report Browser by clicking **Reports**.
3. Select the reports that you want to view:
   - To view your custom reports, click the **My Reports** tab.
   - To view all available reports, click the **All reports** tab.
4. Hover over the report shortcut, click the **Actions** icon, and then click **Schedule**.

   > **Result:**
   >
   > Report scheduling task window appears.

5. **Optional:** To provide the task details, in the **Perform** report scheduling task window that appears, configure the following parameters:
   a. In the **Task Definition** section, in the **Task Name** field, edit the scheduling task label.
      The name supports only letters, numbers, and the underscore character.
   b. In the **Task Description** field, enter a task summary.
6. In the report scheduling task window, provide the scheduling details:
   a. In the **Task Scheduling** section, in the **Time of Day** list, select when to run the report.

©2025 Pegasystems Inc

> **NOTE:** The time corresponds to the time zone set for the server that runs the Pega Platform

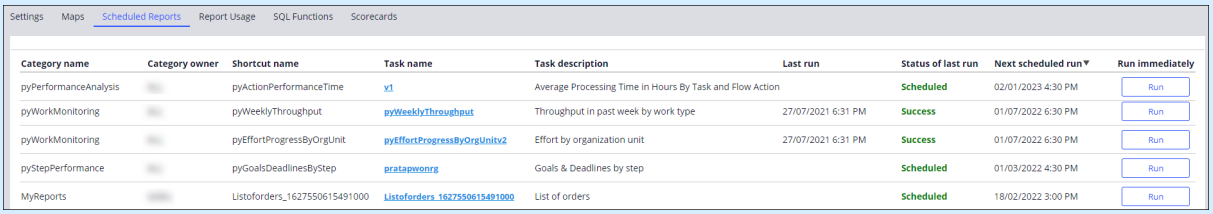
b. In the **Recurrence** section, select how often you want to repeat the task. For example, if you want to run the report once, click **One-Time only**.

c. **Optional:** To rerun the task if errors occur, select the **On error, retry** check box, and then specify the interval and the maximum number of attempts.

7. In the **Task Output Processing** section, provide the report processing details: For more information, see Managing report subscriptions.
8. Schedule the report by clicking **Submit**.

For example:

You can view and manage all scheduled reports in Dev Studio. The following figure presents a detailed list of reporting tasks with current status and the scheduled run date.



Category name	Category owner	Shortcut name	Task name	Task description	Last run	Status of last run	Next scheduled run	Run immediately
pyPerformanceAnalysis		pyActionPerformanceTime	v1	Average Processing Time in Hours By Task and Flow Action		Scheduled	02/01/2023 4:30 PM	Run
pyWorkMonitoring		pyWeeklyThroughput	pyWeeklyThroughput	Throughput in past week by work type	27/07/2021 6:31 PM	Success	01/07/2022 6:30 PM	Run
pyWorkMonitoring		pyEffortProgressByOrgUnit	pyEffortProgressByOrgUnity2	Effort by organization unit	27/07/2021 6:31 PM	Success	01/07/2022 6:30 PM	Run
pyStepPerformance		pyGoalsDeadlinesByStep	pratapwonrg	Goals & Deadlines by step		Scheduled	01/03/2022 4:30 PM	Run
MyReports		Listoforders_1627550615491000	Listoforders_1627550615491000	List of orders		Scheduled	18/02/2022 3:00 PM	Run

List of scheduled reporting tasks

What to do next:

You can manage the scheduled reports. For more information,see Updating scheduled reporting tasks.

- **Managing report subscriptions**
- **Subscribing to reports**

©2025 Pegasystems Inc

- **Testing scheduled reporting tasks**
- **Updating scheduled reporting tasks**
- **Removing scheduled reporting tasks**
- Running a report

Managing report subscriptions

Keep users up-to-date about certain scheduled reports by managing report subscriptions. For example, you can extend the list of users that receive the report.

> **NOTE:** Users without the required privileges do not receive scheduled reports.

1. If App Studio is not in Preview mode, in the header, click **Preview application**.
2. In the header of App Studio, select the Portal in which you manage reports.
3. In the header of Dev Studio, select the Portal in which you manage reports.
4. In the navigation pane, open the Report Browser by clicking **Reports**.
5. Select the reports that you want to view:
   - To view your custom reports, click the **My Reports** tab.
   - To view all available reports, click the **All reports** tab.
6. Hover over the report shortcut, and click **Actions** > **Schedule**.
7. In the **Perform** dialog box that appears, in the **Task Output Processing** section, manage the report subscriptions:
   a. In the **Output File Format** list, select the report file format.

      > **NOTE:** Reports in the PDF format do not include charts.

   b. **Optional:** To allow other users to subscribe to the report, select the **Enable authorized users to subscribe to task notifications** check box.

©2025 Pegasystems Inc

c. Select the **Send notifications to the following users** check box, click **Add a row**, and then enter the user Operator ID.
d. Select how you want to attach the report to the notification e-mail.

> **For example:**
>
> Click **Embed link to output in notification** to include a link to the report.

8. Click **Configure Notification Message**, and then complete the **Email Parameters** form.
9. Save your changes by clicking **Submit**.

- Scheduling reports

Subscribing to reports

Keep up-to-date with all system changes, and receive copies of scheduled reports by subscribing to these reports. For example, you can monitor finances of your company by subscribing to the monthly report that retrieves data about revenues.

> **Before you begin:**
>
> Ensure that the report to which you want to subscribe runs on a schedule. For more information, see Scheduling reports.

1. If App Studio is not in Preview mode, in the header, click **Preview application**.
2. In the header of App Studio, select the Portal in which you manage reports.
3. In the header of Dev Studio, select the Portal in which you manage reports.
4. In the navigation pane, open the Report Browser by clicking **Reports**.
5. Select the reports that you want to view:
   - To view your custom reports, click the **My Reports** tab.
   - To view all available reports, click the **All reports** tab.

©2025 Pegasystems Inc

6. Hover over the shortcut for the report to which you want to subscribe, and click Actions > Subscribe.

   Result:

   The system displays the **Perform** task scheduling window.

7. In the **Task Output Processing** section, subscribe to the report:
   a. Select the Send Notifications to the following users check box.
   b. Click Add a row.
   c. In the field that appears, enter your operator ID.
8. Click Submit.

- Managing report subscriptions

Testing scheduled reporting tasks

Verify whether your scheduled reporting tasks is completed correctly by performing a test run. For example, you can send the subscribers an immediate email to check whether a notification message is generated.Test runs do not impact the **Last run** and **Status of last run** columns of the report, because they do not register as regular report runs.

1. In the header of Dev Studio, click Configure > Reporting > Scheduled Reports.
2. In the **Run immediately** column for your report, click Run.
3. In the **Run immediately** dialog box, confirm the run by clicking Submit.

Result:

The system delivers the notification message to the subscriber.

©2025 Pegasystems Inc

Updating scheduled reporting tasks

Dynamically adapt your scheduled reports to your evolving business needs by updating the scheduled reporting tasks. For example, if you want to display charts in the report, you can change the output format to Microsoft Excel Workbook that supports charts.

1. In the header of Dev Studio, click **Configure** > **Reporting** > **Scheduled Reports**.
2. In the **Task name** column, click the label of the scheduled task that you want to update.

   > **NOTE:** Each task is accessible for one user at a time.

3. At the bottom of the PX-ST-*Task Number* tab, click **Update**.
4. Update the task scheduling form.

   > **For example:**
   >
   > You can change the date when the report runs.

   For more information about the form, see Scheduling reports.

5. At the bottom of the tab, click **Submit**.

Removing scheduled reporting tasks

Control all scheduled reporting tasks in your application to ensure that the system runs only the reports that are required. For example, if you do not need a report anymore, you can save processing capacity by removing a scheduled reporting task.

1. In the header of Dev Studio, click **Configure** > **Reporting** > **Scheduled Reports**.
2. In the **Task name** column, click the label of the scheduled task that you want to remove.
3. At the bottom of the PX-ST-*Task Number* tab, click **Update**.

©2025 Pegasystems Inc

4. At the bottom of the tab, click **Delete**.
5. If the task is still in the list, in the upper-right corner of the tab, load the latest changes by clicking **Refresh**.

Editing a report

Adapt to dynamic needs and continuously improve your reporting capabilities by modifying your reports. You can change the way your report displays information or filters data for different audiences.For example, in a financial report, you can filter the data to retrieve information about the income of your company generated within certain period.

1. If App Studio is not in Preview mode, in the header, click **Preview application**.
2. In the header of App Studio, select the Portal in which you manage reports.
3. In the navigation pane, open the Report Browser by clicking **Reports**.
4. Select the reports that you want to view:
   - To view your custom reports, click the **My Reports** tab.
   - To view all available reports, click the **All reports** tab.
5. Click a report to edit.
6. In the upper-right corner, click **Edit report**, and edit the report:

Choices	Actions
Change the report title	At the top of the Report Editor, in the text field, enter a new report title.
Add a column to the report	From the **Data Explorer** section, drag the column next to an existing report column.
Remove a column from the report	a. Right-click the report column, and then click **Delete**. b. In the **Delete Column Confirmation** modal dialog box, click **Submit**.

©2025 Pegasystems Inc

Choices	Actions
Change the column heading	a. Right-click the report column, and then click Heading. b. In the Column heading field, enter the column name, and then click Apply changes.
Format the column values	a. Right-click the report column, and then click Format. b. In the Format values field, enter appropriate value, and then click Apply changes.
Sort the column values	a. Right-click the report column, and then click Sort. b. If you want to display values in ascending order, click Lowest to highest. c. If you want to display values in descending order, click Highest to lowest.
Add a chart to the Summary or Chart report type	a. Click Add chart. b. Click All chart types. c. In the **Select Chart Type** modal dialog box, click a chart type that you want to include in the report. d. Click Submit. e. From the Available columns section, drag the columns into the chart. f. Click Apply changes.

©2025 Pegasystems Inc

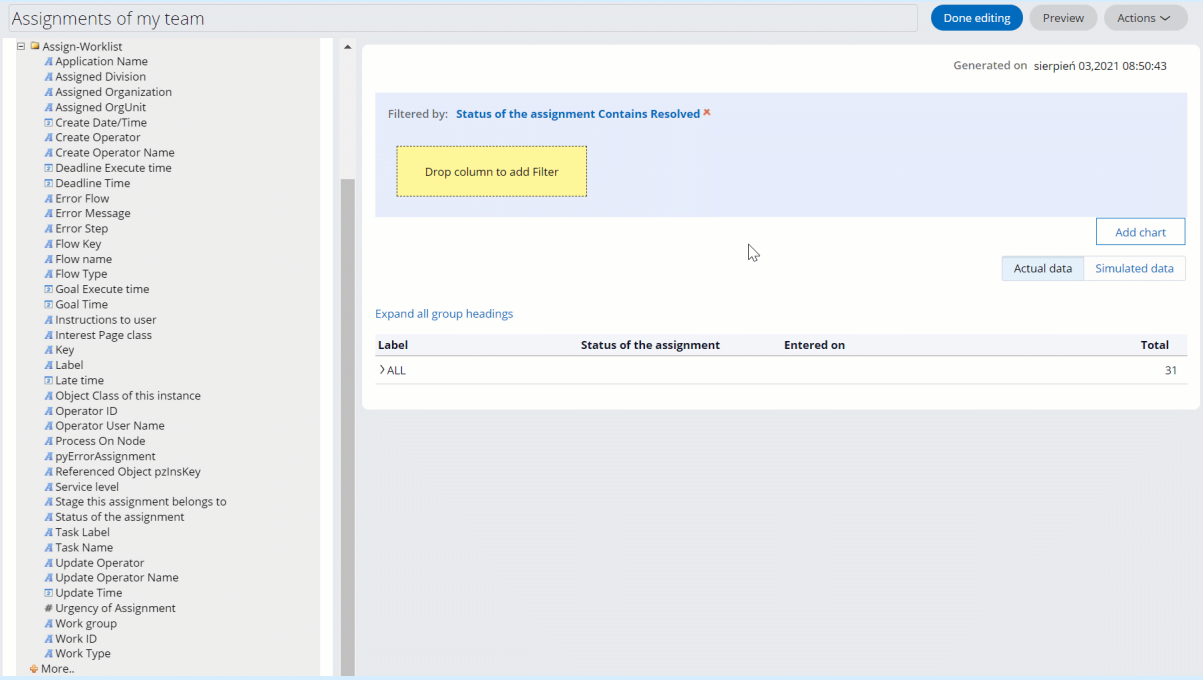
Choices	Actions
Edit the report chart	a. If a chart already exists, click Edit chart. b. Click Title to change the chart title. c. Click Color settings to customize the chart colors. d. Click General settings to set general chart settings. e. Click Apply changes.
Filter the columns content	a. From the Data Explorer section, drag the filter into the **Drop column to add Filter** box. b. In the **Edit filter** section, set filter options, and then click Apply changes. For more information about filter conditions, see Editing filter conditions.
Switch between simulated and actual data	Depending on your business need, click Actual data or Simulated data. **NOTE:** Designing a report with simulated data avoids repeated database queries while you edit the report, allowing for faster changes.

7. Click Done editing.

©2025 Pegasystems Inc

For example:

The following visualization presents how to add a chart to the report.



Adding chart to the report

- **Report Editor**
- **Modifying list reports**
- **Modifying summarized reports**
- **Adding filter conditions**
- **Managing charts in Report Editor**
- **Removing charts in Report Editor**
- **Converting a list report to a summarized report**
- **Converting a summarized report to a list report**
- Editing filter logic

©2025 Pegasystems Inc

Report Editor

Report Editor provides options for managing the content of reports. You can update reports with specific columns to meet your business needs. For example, in a report that lists resolved cases, you can insert a column that contains information about exceeded deadlines to identify areas that require improvement.You can access Report Editor from Report Viewer when running reports.

Report Editor features

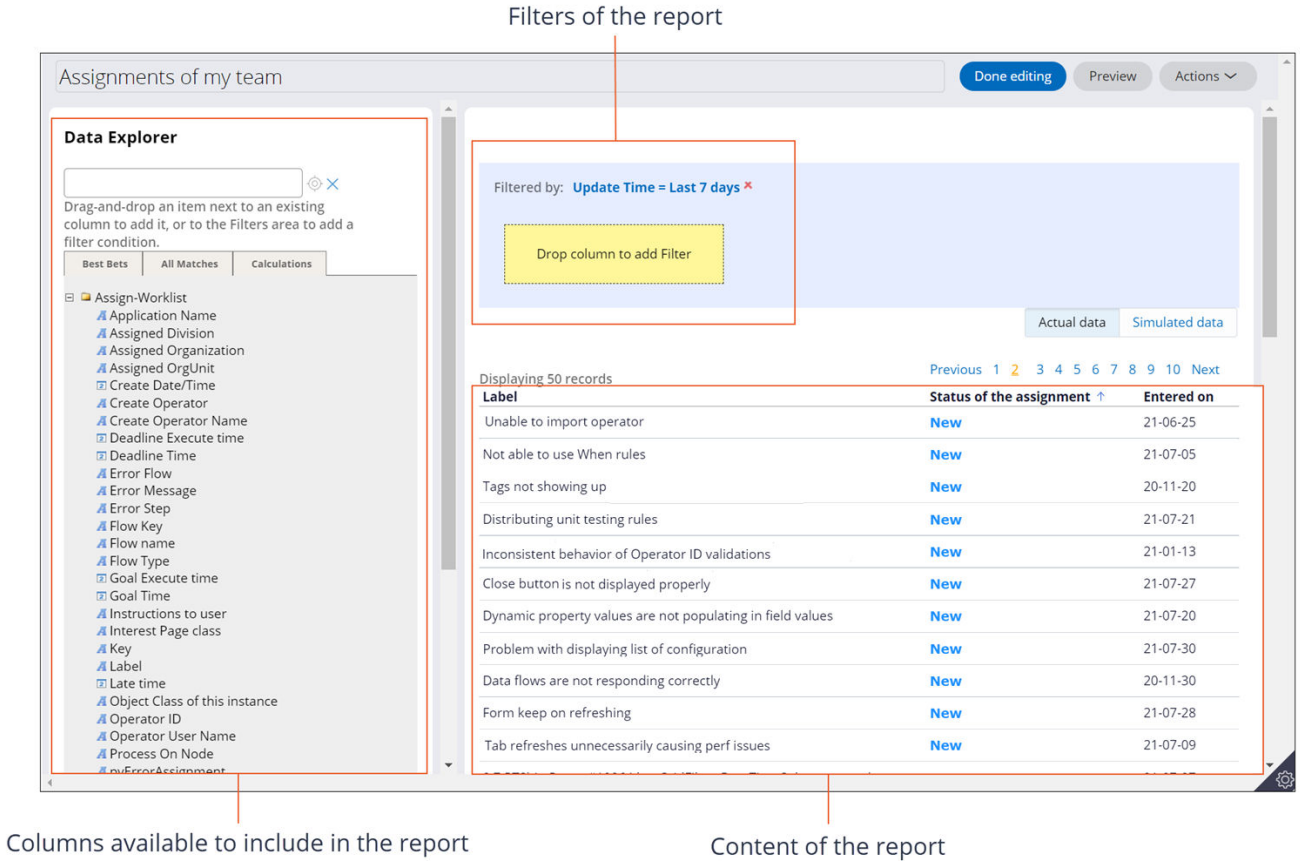
By accessing Report Editor, you can perform the following actions:

- Change the report name.
- Manage the report contents.

  For example, you can change column names, delete columns, or change the format of the data in columns.

- Add new columns to the report.
- Sort data in columns.
- Change column width.
- Filter the data in columns, and change the filter logic.
- Convert the report to a summarized report or a list report.
- Switch between simulated and actual data.
- Add or edit a chart (summarized reports only).

©2025 Pegasystems Inc



Report Editor tool with an exemplary report during customization

- **Data Explorer**

- Editing a report
- Adding columns to reports

Data Explorer

Data Explorer displays properties and calculations that you can include as columns to conveniently create comprehensive reports that meet your business needs. For example, in a report that lists resolved bugs, you can add a column of information about the time spent resolving the bugs, to analyze team performance.

Data Explorer is a part of Report Editor.

©2025 Pegasystems Inc

You can also use Data Explorer to define filter conditions. For more information about filter conditions, see Adding filter conditions.

Data Explorer divides the available properties and calculations into the following categories:

Best Bets

Properties that match your report type.

For example, in a list report about resolved bugs, you can include a column with a description of each bug.

All Matches

All available properties that you can use.

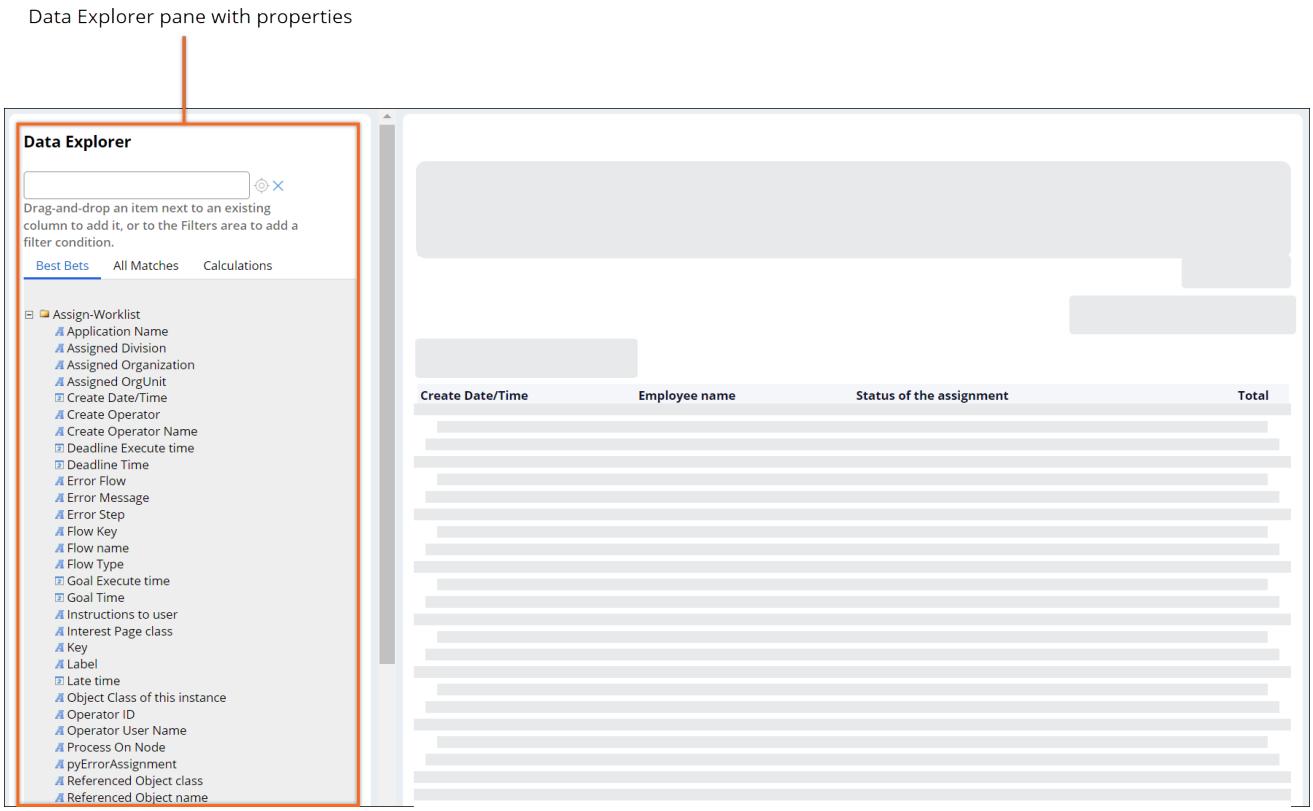
Calculations

SQL functions that you can use as columns or filters.

For example, you can use the **Older than n days** function to filter the report to display cases that remain open for a specific period of time.

The following graphics presents Data Explorer pane available in Report Editor.

©2025 Pegasystems Inc



Data Explorer pane

- Adding columns to reports
- Editing a report

Modifying list reports

Manage the appearance and content of list reports by modifying the column settings in the Report Viewer. For example, you can filter the content to display in columns, sort the data, or adjust column widths to make the most relevant data easier to see.The changes that you make apply only to the current view of report results.

1. In the header of Dev Studio, select the Portal in which you manage reports.
2. In the navigation pane, open the Report Browser by clicking **Reports**.
3. Select the reports that you want to view:
    - To view your custom reports, click the **My Reports** tab.
    - To view all available reports, click the **All reports** tab.

©2025 Pegasystems Inc

4. Click a report that you want to view.
5. In the Report Viewer, hover over the header of the column that you want to modify, and then click the **Open Menu** icon.
6. Modify the column:

Choices	Actions
Filter the column results	a. Click **Filter**. b. In the **Search Text** field, enter the text by which you want to filter the column, and then click **Apply**.
Sort the column results	a. If you want to display values in ascending order, click **Sort** > **Lowest to highest**. b. If you want to display values in descending order, click **Sort** > **Highest to lowest**.
Modify the column appearance	a. Click **Appearance**. b. In the **Column heading** field, enter a name for the column header. c. In the **Format values** field, press the Down arrow key, and then select a format for the values. d. In the **Column width** field, enter the column width in pixels or as a percentage value. e. Click **Submit**.
Summarize values of certain columns	a. Click **Summarize**. b. Select the columns with which you want to group the report data. A report that you summarize in this way displays row counts that are

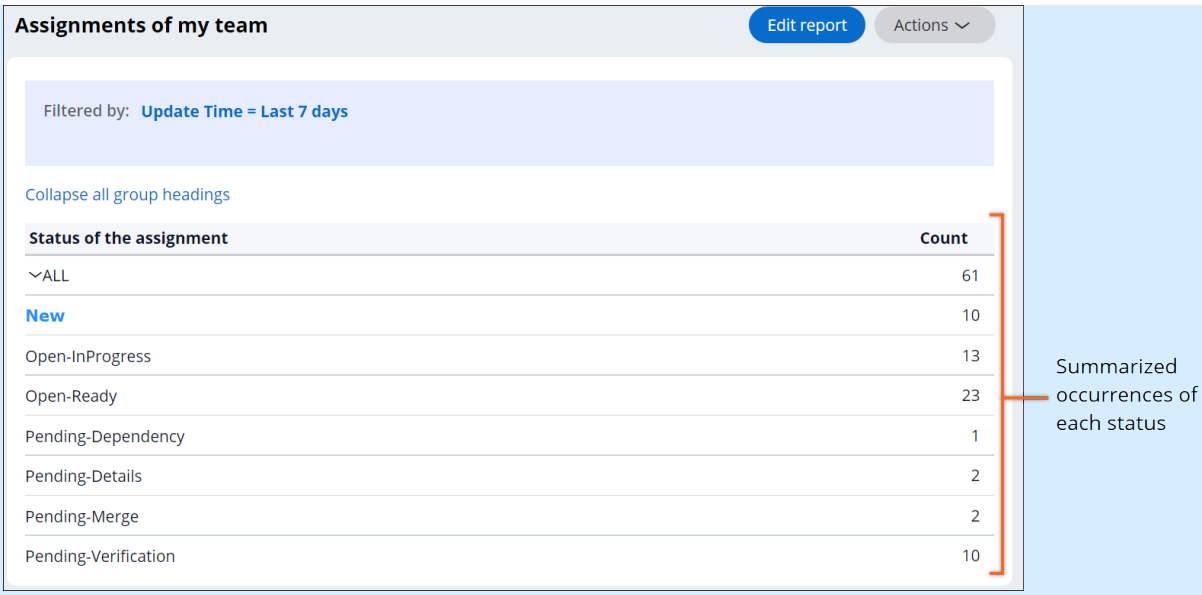
©2025 Pegasystems Inc

Choices	Actions
	grouped by the values in the columns that you select. c. **Optional:** To expand all group sections when the report opens in the Report Viewer, select the **Expand all group sections by default in initial display** check box. d. **Optional:** To exclude group headings from report results, select the **Do not display group headings** check box. The results display in a format that is similar to a spreadsheet, without subtotals, sub-averages, or similar values. e. Click **Submit**.
Remove the column	a. Click **Delete**. b. Confirm the deletion by clicking **Submit**.

For example:

In comprehensive reports that include a lot of data, you can count occurrences of certain values by summarizing columns. The following report presents the number of assignments with certain status.

©2025 Pegasystems Inc



Summarized values in a report

- Modifying summarized reports
- Report Viewer

Modifying summarized reports

Adjust the appearance and contents of summarized reports by modifying the column settings in the Report Viewer. For example, you can specify the format of values, or change the column width.The changes apply only to the current view of the report results.

1. In the header of Dev Studio, select the Portal in which you manage reports.
2. In the navigation pane, open the Report Browser by clicking **Reports**.
3. Select the reports that you want to view:
   - To view your custom reports, click the **My Reports** tab.
   - To view all available reports, click the **All reports** tab.
4. Click a report that you want to view.
5. In the Report Viewer, right-click the header of the certain column, and then modify that column:

©2025 Pegasystems Inc

Choices	Actions
Change the column heading	a. Click Heading. b. In the Column heading field, enter a new column heading. c. Click Apply changes.
Specify the format of the data values	a. Click Format. b. In the Format values field, press the Down arrow key, and then select the data format. c. Click Apply changes.
Adjust the column width	a. Click Width. b. In the Column width field, enter the column width in pixels or as a percentage value. c. Click Apply changes.
Remove the report column	a. Click Delete. b. Confirm the deletion by clicking Submit.
Sort the column results	a. If you want to display values in ascending order, click Sort > Lowest to highest. b. If you want to display values in descending order, click Sort > Highest to lowest.
Display values horizontally	Click Display values across columns.

- Modifying list reports
- Report Viewer

©2025 Pegasystems Inc

Adding filter conditions

Decide what data you want to present in your reports by adding filter conditions that provide options to display only relevant information. For example, you can filter a report to display sales from a specific year, to analyze purchase tendencies over time.By using Report Editor, you can create filter conditions that use properties or calculations to filter data. For more information, see Data Explorer.

1. Run the report that you want to update.
   For more information, see Running a report.
2. In the upper-right corner of the screen, click **Edit report**.
3. In the **Data Explorer** section, add a filter condition:

Choices	Actions
Add a function to the filter condition	a. From the **Calculations** tab, drag a specific function to the **Drop column to add filter** section. b. In the **Calculation builder** dialog box, in the **Function input** field, press the Down arrow key, and then select a property to calculate. **NOTE:** Some functions require more than one input property, so more **Function input** fields are available. c. Click **Submit**. d. In the **Edit filter** section, in the *Function name* list of operators,

©2025 Pegasystems Inc

Choices	Actions
	select a relational operator for the filter condition. e. In the *Function name* box, press the Down arrow key, and then select a property to compare with your function. For example, you can use the **Older than n days** function to filter the report to display cases that remain open for a specific period of time.
Add a property to the filter condition	a. From the **Best Bets** or **All Matches** tabs, drag a specific property to the **Drop column to add filter** section. For more information on Data Explorer tabs, see Data Explorer. b. In the **Edit filter** section, in the *Filter condition name* list of operators, select a relational operator for the filter condition. c. In the *Filter condition name* box, enter specific value, or press the Down arrow key, and then select a property to compare with your property. d. **Optional:** To add a function with which the system compares your property, click **Calculation Builder**, and specify the function. For example, you can use the **Day of the week** function to filter the report

©2025 Pegasystems Inc

Choices	Actions
	to display all resolved cases on a particular day of the week.

4. **Optional:** To change the name of the filter condition that is visible in the Report Viewer and the Report Editor, in the **Filter Caption** field, enter a new name.
5. **Optional:** To use a null comparison value if a comparison value is not specified, select the **Use null if empty** check box.
6. **Optional:** To apply a case insensitive comparison, select the **Ignore case** check box.

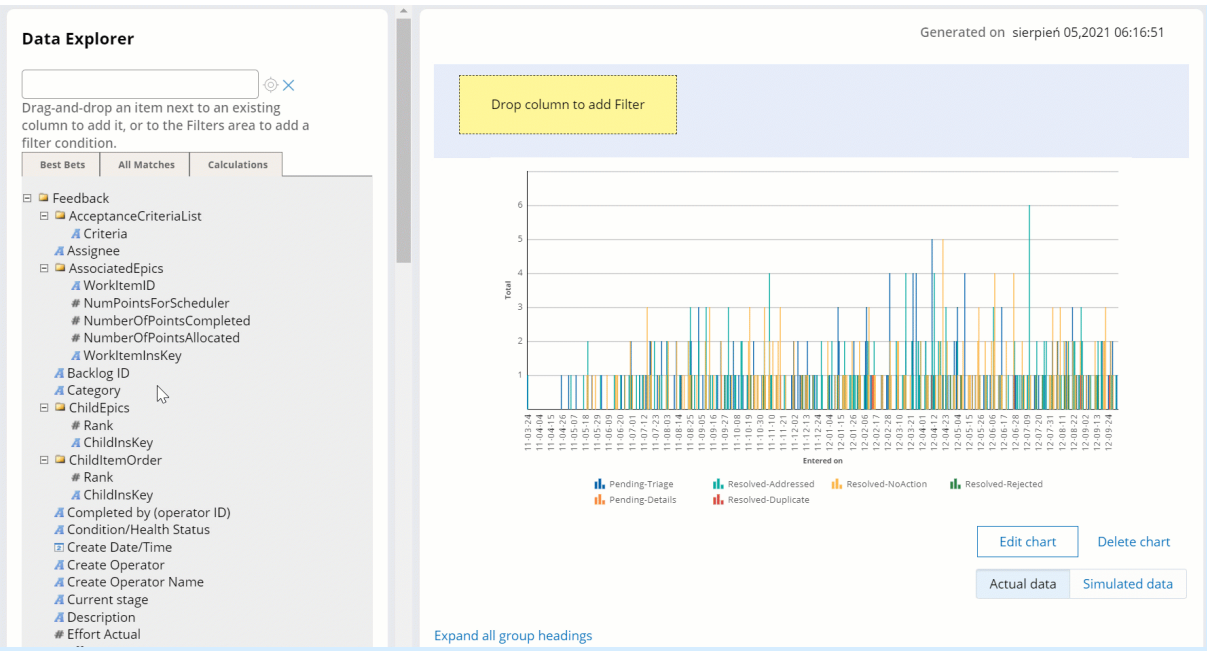
> **IMPORTANT:** Selecting the **Ignore case** checkbox results in adding the `UPPER` function to the `WHERE` clause of your query. As a result, it might affect the system performance.

7. Add the filter condition by clicking **Apply changes**.

For example:

The following visualization presents how to apply two filters that restrict the results to include resolved assignments that are created date prior to 2011-11-19.

©2025 Pegasystems Inc



Applying filters in summarized report

- **Editing filter conditions**
- **Editing filter logic**
- Editing a report

Editing filter conditions

Customize the filter conditions to display specific information in the report. For example, you can filter the report to display a list of user IDs that contains the division ID.Filter conditions compare the values of properties based on relational operators that indicate the relation between the properties. For example, if the filter condition contains the `is equal` relational operator, the report displays properties whose values are equal.

Before you begin:

Ensure that the report contains at least one filter condition. For more information, see Adding filter conditions.

©2025 Pegasystems Inc

1. Run the report that you want to update.
   For more information, see Running a report.
2. In the upper-right corner of the screen, click **Edit report**.
3. In the **Filtered by** section, click the filter condition that you want to edit.
4. In the **Edit filter** dialog box, edit the filter condition:

Choices	Actions
Select a property for the filter condition	a. In the *Filter condition name* list of operators, select a relational operator for the filter condition. b. In the *Filter condition name* box, press the Down arrow key, and then select a property to compare with your property.
Select a function for the filter condition	a. In the *Filter condition name* list, select a relational operator for the filter condition. b. Click **Calculation Builder**. c. In the **Calculation builder** dialog box, in the **Select a function** field, press the Down arrow key, and then click a function. d. In the **Function input** field, press the Down arrow key, and then select a property to calculate. Some functions require more than one input property, so more **Function input** fields are available. e. Click **Submit**.

©2025 Pegasystems Inc

Choices	Actions
Manually select values for the filter condition	a. In the *Filter condition name* list, select a relational operator for the filter condition. b. Click **Select values**. c. If you want to display all available values, click the **Available values** tab. d. If you want to display valid values only, click the **Valid values** tab. e. Select one or more values, and then click **Submit**.

5. **Optional:** To change the name of the filter condition that is visible in the Report Viewer and the Report Editor, in the **Filter Caption** field, enter a new name.
6. **Optional:** To use a null comparison value if a comparison value is not specified, select the **Use null if empty** check box.
7. **Optional:** To apply a case insensitive comparison, select the **Ignore case** check box.

> ⚠ **IMPORTANT:** Selecting the **Ignore case** checkbox results in adding the `UPPER` function to the `WHERE` clause of your query. As a result, it might affect the system performance.

8. Click **Apply changes**.

- Editing a report
- Report Viewer

Editing filter logic

Customize a report to display only data that is relevant to your needs by managing how the report filters information.If your report contains multiple filter conditions, you can

©2025 Pegasystems Inc

edit the filter logic to display only the required sets of data. By default, filters display data that matches all of the filter conditions. When you edit the filter logic, you can use parentheses and AND or OR to specify the filtering logic of complex conditions.

1. Run the report that you want to update.
   For more information, see Running a report.
2. In the upper-right corner of the screen, click **Edit report**.
3. Edit the filter logic for your report:
   a. Click **Change Logic**.
   b. In the **Filter logic** section, in the **Logic** field, enter the logic that you want to use to filter the report data.
      Pega Platform assigns a literal identifier to each filter condition. For example, to display the data that matches condition A and either condition B or C, enter `A AND (B OR C)`.

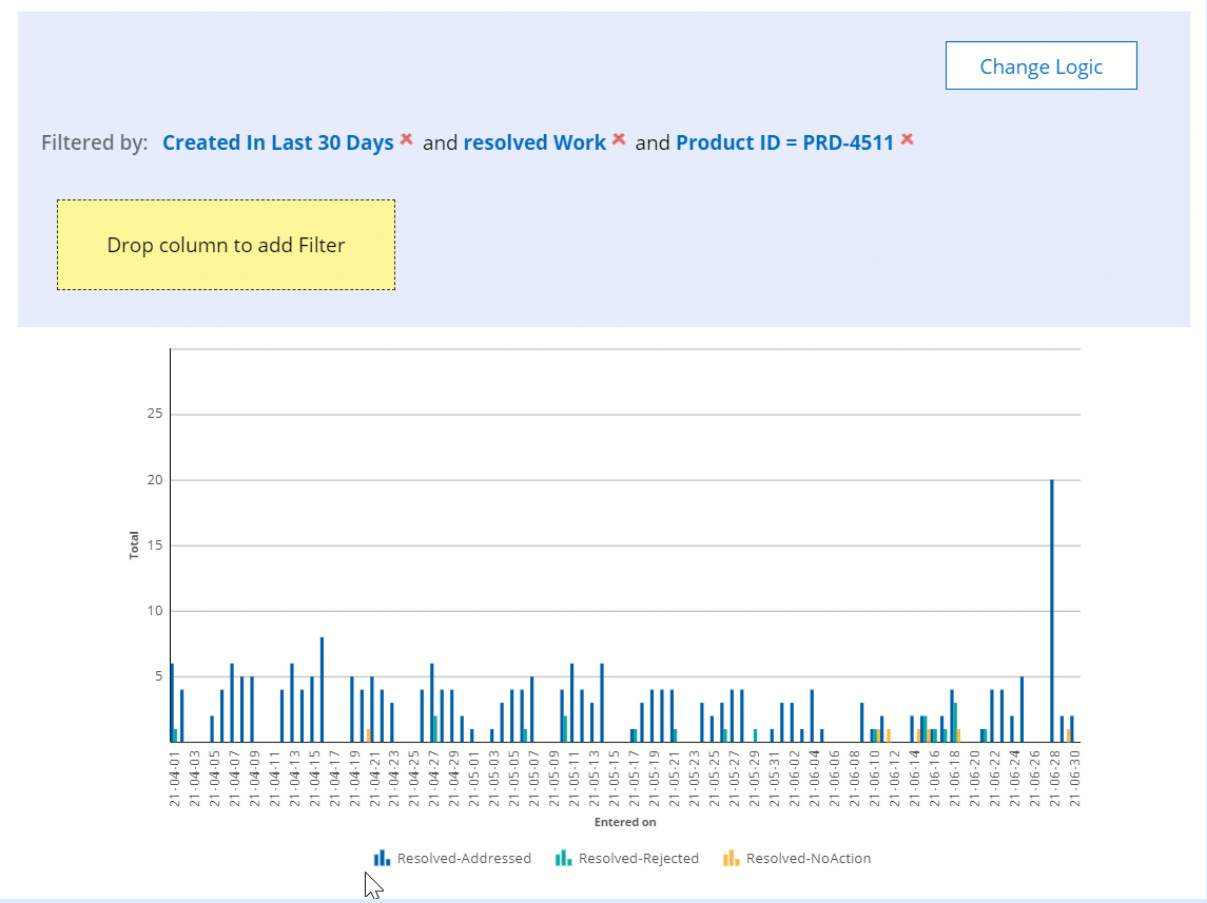
      > **NOTE:** Include all available filter conditions in the filter logic. Otherwise, the logic does not change.

   c. Click **Apply changes**.

For example:

The following visualization presents how to customize the filter logic. Initially, the system filters the results to retrieve data that match all three conditions (`A AND B AND C`). When the logic changes to `A AND (B OR C)`, the results match condition A and either B or C, hence the outcome is different.

©2025 Pegasystems Inc



Customizing filter conditions

- Editing filter conditions
- Adding filter conditions
- Editing a report

Managing charts in Report Editor

Make the report more readable by presenting data in the form of informative charts. For example, you can visualize financial data of your company to enhance the process of data analysis.

NOTE: Reports require summarized data to include a chart. The summarized data is a value that corresponds to particular events in the application. For

©2025 Pegasystems Inc

example, a chart requires the total number of created cases, rather than a detailed list of the titles of cases.

1. Run the report that you want to update.
   For more information, see Running a report.
2. In the upper-right corner, click **Edit report**.
3. If you want to add a chart, in the Report Editor, click **Add chart**.
4. If you want to manage the existing chart, in the Report Editor, click **Edit chart**.

> **NOTE:** Each report requires aggregated data to include a chart. You aggregate data by summarizing a column.

5. Customize the chart:

Choices	Actions
Select the chart type	a. Click **All chart types**. For more information about chart types, see Charts in reports. b. In the **Select Chart Type** modal dialog box, click a chart type that you want to include in the report. c. Click **Submit**.
Add or edit the chart title	a. Click **Title**. b. In the **Chart Title** window, enter the chart title, and then specify the font size and style. c. Click **Submit**.

©2025 Pegasystems Inc

Choices	Actions
Define the chart color options	**NOTE:** Available options depend on the chart type. a. Click **Color settings**. b. If you want to define the general chart color settings, click **General color settings**. c. If you want to define the colors that conditionally apply to the chart elements, click **Conditional colors**. d. If you want to define the colored lines that indicate value thresholds in the chart, click **Threshold colors**.
Adjust the chart general settings	**NOTE:** Available options depend on the chart type. a. Click **General settings**. b. To adjust the font settings, click **Format options**. c. To manage the placement of the chart legend, or manage the slider that filters values in the chart, click **Legend/Slider options**. d. To manage the chart display options available in the Report Viewer, click **Enable user commands**.

©2025 Pegasystems Inc

Choices	Actions
	For example, you can switch between two-dimensional and three-dimensional chart displays, or maximize the size of the chart for better readability.
Specify the chart content	a. From the **Available columns** section, drag the columns into the relevant chart drop zones. b. In a drop zone, next to a column name, access the data settings by clicking the **Properties** icon. For example, you can change the format and range of displayed values, or edit the axis names.

6. **Optional:** To style the charts in the application skin, see Styling charts in the application skin
7. Click **Apply changes**.
8. **Optional:** To preview the chart data, click **Preview**.

- Charts in reports
- Creating a report

Removing charts in Report Editor

Keep your reports up-to-date by removing redundant charts from summarized reports. For example, if you delete data that the chart uses, you can remove the chart to avoid presenting incomplete results.

1. Run the report that you want to update.
   For more information, see Running a report.
2. In the upper-right corner of the screen, click **Edit report**.

©2025 Pegasystems Inc

3. Remove the chart by clicking **Delete chart**.
4. Save your changes by clicking **Done editing**.

- Managing charts in Report Editor
- Editing a report

Converting a list report to a summarized report

Analyze the numerical data to discover the number of particular events in an application by converting a list report to a summarized report. For example, you can summarize the total number of user IDs in the application.

1. Run the report that you want to update.
   For more information, see Running a report.
2. In the upper-right corner of the screen, click **Edit report**.
3. Click **Actions** > **Summarize**.
4. In the **Summarize and Sort** section, in the **Summarize** list, select one of the following methods to summarize the values of each column.
   - To remove a column from the report, select **Do not include**.
   - To group the summarized values by the values in a column, select **GROUP BY**.
   - To display the number of rows in a column for each group, select **COUNT**.
   - To display the number of unique values in the column for each group, select **COUNT DISTINCT**.
   - To display the highest value in each group, select **MAX**.
   - To display the lowest value in each group, select **MIN**.

   > **NOTE:** You can select one option for each column.

5. In the **Sort** list, select one of the following methods to sort the values of each column.

©2025 Pegasystems Inc

- To display the values in ascending order, select **Lowest to highest**.
- To display the values in descending order, select **Highest to lowest**.
- To leave the order without changes, select **No sorting**.

6. **Optional:** To add a column that displays the number of rows, select the **Include a column of row counts** check box.
7. **Optional:** To expand all group sections when you open the report in the Report Viewer, select the **Expand all group sections by default in initial display** check box.
8. To exclude the group headings from the report results, select the **Do not display group headings** check box.
9. Convert the list report by clicking **Apply changes**.

- Report types
- Editing a report
- Report Viewer

Converting a summarized report to a list report

Make your report more readable and your data analysis more efficient by displaying the summarized columns of a report in the form of a list. For example, you can view a list of all open cases to verify the remaining workload.

1. Run the report that you want to update.
   For more information, see Running a report.
2. In the upper-right corner of the screen, click **Edit report**.
3. Click **Actions** > **List**.
4. In the **Confirm Conversion To List** modal dialog box, convert the report by clicking **Submit**.

- Report types
- Editing a report
- Report Viewer

©2025 Pegasystems Inc

Creating advanced reports

Facilitate the process of data analysis by presenting the data from the database in the form of a table.

In Pega Platform, you can create custom reports by managing various features to make the reports comprehensive and easy to follow. For example, you can specify custom actions that users can perform with reports.

> **NOTE:** You can use each report definition as a subreport to provide data to other report definitions.

1. In the header of Dev Studio, click **Create** > **Reports** > **Report Definition**.
2. Create a report definition:
   a. In the **Label** field, enter the title or description of the record.
   b. **Optional:** To edit the name, in the **Identifier** field, click **Edit**, and then enter a new name.
      By default, the **Identifier** field inherits the name from the **Label** field. A valid identifier name starts with a letter, and consists of letters, digits, and underscores.
   c. In the **Apply to** field, select a Class for the report definition, or select a work pool to report on all work types in the work pool.

> **NOTE:**
>
> For certain uses, you can select an abstract Class for the field, such as the *Work-* Class. The system then determines a work pool or a single Class that is the source of the report contents. For example, a report definition with *Work-* Class presents drill-down details that another report definition can reference.

©2025 Pegasystems Inc

> The list of available class names depends on the Ruleset that you select. Each class can restrict applying Rules to an explicit set of Rulesets as specified on the **Advanced** tab of the class form.

d. In the **Add to Ruleset** list, select the Ruleset in which you want to include the report definition.
e. **Optional:** To associate the report definition with a work item, in the **Work item to associate** field, press the Down arrow key, and then select a work item.
f. Click **Create and open**.

3. In the **Edit Report definition** page, define the contents and display of the report. For example, you can edit filters, include a chart, or manage the report body. For more information about the configurable options, see Report definitions.
4. Click **Save**.

- **Report definitions**
- **Associations**
- **Reporting landing page**
- **Excluding items from Report Browser**
- **Creating an advanced report category**
- **Creating a report shortcut**
- **Reviewing usage report statistics**
- **Specifying a data source for reports**
- Report types

PEGA ©2025 Pegasystems Inc

Report definitions

Report definition rules provide a convenient way to create advanced reports and meet complex user needs. You can make your reports more adaptive and flexible by customizing them with various options. For example, you can add custom columns to your report, and then include a chart to make data analysis more efficient.

Report definition rules define report definitions that provide more advanced options for displaying and managing data than standard reports that you can create in App Studio.

Report definition rules generate an SQL query that retrieves and sorts information from the PegaRULES database, an external database, or an Elasticsearch index.

By accessing a report definition rule, you can take advantage of the following features:

- Add custom columns to your report, and format the report data.
- Manage filter conditions.

  For example, you can add a new filter, and prohibit changes to the filter to ensure that report results remain the same for each user.

- Include or edit charts that present report data.
- Manage the appearance of reports in Report Viewer.

  For example, you can lock the report header to make it visible while you scroll down through the report, or manage the Visibility of certain report in Report Browser.

- Control the amount of resources that the system requires to run the report.
- View the list of associated classes that share matching values.
- Manage general data access settings.

  For example, you can select the **Report on descendant class instances** checkbox to include data from one or more descendant classes of the report definition's

©2025 Pegasystems Inc

primary class. If descendant classes are mapped to multiple class tables, the generated query uses UNIONs to include this data.

When selected, the following options are displayed:

Include single implementation class
: Uses implementation class mapping. When there are multiple implementation classes, this option has no effect. To search multiple implementation classes, create a custom version of the Report Definition that has the "applies to" implementation class.

Include all descendant classes
: Default for new reports. Includes all descendant classes except as constrained by a filter condition on *.pxObjClass*.

> **NOTE:** The **Include all descendant classes** option refers to the class specified in the **Applies to** field.

- **Managing filter options in Report Viewer**
- **Charts in reports**

- Managing filter options in Report Viewer
- Adding charts in the Report Definition rule form
- Creating simple associations between classes
- Setting resource limits for reporting

Managing filter options in Report Viewer

Control the way that Report Viewer displays reports by managing filter conditions on the report definition rule form. For example, you can prohibit changes to filter conditions to ensure that report results remain the same for each user.

©2025 Pegasystems Inc

1. In the navigation pane of Dev Studio, click **Records**, and then click **Reports** > **Report definition**.
2. In the list of report definitions, click the report definition for which you want to edit the filter conditions.
3. Manage the filter conditions by choosing from the following options:

Choices	Actions
Manage a specific filter	a. Click the **Query** tab. b. In the **Edit filters** section, manage the filters of the report definition. **For example:** Next to the specific filter, click the **Delete** icon to remove the filter. c. To manage the filter's options, next to the specific filter condition, click the **Options** icon, and customize the filter. **For example:** In the **Edit filter options** modal dialog box, in the **Report Viewer options** list, select **Allow any changes** to authorize users to edit or remove the filter condition. d. Click **Submit**.
Manage all filters	a. Click the **Report Viewer** tab

©2025 Pegasystems Inc

Choices	Actions
	b. In the **User actions** section, select the check boxes with the options that you want to apply to all filter conditions in the report. **For example:** Next to the **Enable actions** checkbox, click **Options**, and then select the **Export to Excel** checkbox to enable exporting the report as Excel file. Select the **Display report filters** check box to display all filters in the report when running that report in the Report Viewer.

4. In the upper-right corner of the screen, click **Save**.

Charts in reports

Present data in a clearer way by using a wide range of charts that you can include in your reports. For example, to analyze the current workload, you can display data about case statuses as a pie chart.Several charts include subtypes that provide more specialized ways of presenting data.

Available chart types

Column

Displays values as vertical bars.

The following subtypes are available:

©2025 Pegasystems Inc

- Clustered – Groups columns together.
- Stacked – Stacks columns of a given group together.
- 100% – Stacks columns together. The total height for each column is always 100% of the height of the chart.

Line

Connects values with a line.

The following subtypes are available:

- Segment – Indicates value variations with changes in the angle of the line.
- Curved – Indicates value variations with a curved line.
- Step – Indicates value variations with as steps, with the first value as a horizontal step.
- Reverse Step – Indicates value variations with steps with the first value as a vertical step from the first to the second value.
- Horizontal – Indicates value variations with points along a horizontal line.

Combo charts

Combine column charts and line charts. Combo charts include one vertical axis or two vertical axes. The two data sets share the same horizontal axis.

The following subtypes are available:

- Single Y-Axis Clustered – Single horizontal axis with clustered columns.
- Single Y-Axis Stacked – Single horizontal axis with stacked columns.
- Dual Y-Axis Clustered – Double horizontal axis that applies to columns and lines. The columns on the chart are clustered.
- Dual Y-Axis Stacked – Double horizontal axis that applies to columns and lines. The columns on the chart are stacked.

Area

Displays data sets one on top of another to identify relationships or differences among them.

PEGA ©2025 Pegasystems Inc

The following subtypes are available:

- Overlaid – Displays values as separate transparent ranges.
- Stacked – Displays values as non-transparent layers.
- 100% – Values cover the entire chart height.

Funnel

Displays values in descending order, which is represented by increasingly smaller graphics.

The following subtypes are available:

- Normal – Each segment of the funnel is connected. You can click one segment to separate it from the rest.
- Exploded – Separates each segment of the funnel.

Pyramid

Displays values in descending order represented by increasingly larger graphics. Mirror reflection shape of the funnel chart type.

The following subtypes are available:

- Normal – Each segment of the pyramid is connected. You can click one segment to separate it from the rest.
- Exploded – Separates each segment of the pyramid.

Bubble

Displays values as bubbles. The size of the bubble corresponds to the presented value.

Pie

Displays values as a segmented circle.

The following subtypes are available:

PEGA ©2025 Pegasystems Inc

- Normal – Each segment of the chart is connected. You can click the segments to separate them from the rest.
- Exploded – Disconnects the segments and locates them away from the center of the chart. You can click the segments to connect them together.
- Doughnut – The segments form a circle with an empty space in the center. You can click the segments to separate them from each other.

Bar

Displays values as horizontal bars.

The following subtypes are available:

- Clustered – Groups the bars together.
- Stacked – Stacks the columns of a given group together.
- 100% – Stacks the columns together. The total height for each column is always 100% of the height of the chart.

Spark

Compresses charts so that you can embed them in a form or layout to show a wide range of data in a single view.

The following subtypes are available:

- Line – Displays values as points connected with lines.
- Column – Displays values as clustered columns.

Gauge

Displays a single value as a gauge.

The following subtypes are available:

- 270 Degree Arch – Displays values in an arch with a marker in the center that indicates the value.
- Speedometer – Displays values in a gauge that resembles a speedometer with a marker in the center that indicates the value.

PEGA ©2025 Pegasystems Inc

- Half dial – Displays values in a semicircle with a marker in the center that indicates the value.
- Half arch – Displays values in an arch with a marker in the center that indicates the value.
- Dial – Displays values in a circle with a marker in the center that indicates the value.
- Linear – Displays values in a line graph with a marker that indicates the value.
- Horizontal LED – Displays values in a horizontal line graph with a marker that indicates the value. Black and white stripes mark the value range.
- Vertical LED – Displays values in a vertical line graph with a marker that indicates the value. Black and white stripes mark the value range.
- Cylinder – Displays values in a three-dimensional cylindrical graph with a metric on the left that indicates the value.
- Thermometer – Displays values in a gauge that resembles a thermometer with a metric on the left that indicates the value.

Map

Distributes geographical data by categories, regions, or entities on the map.

> **NOTE:** You can configure map charts in Dev Studio.

- **Defining associations for map charts**
- **Adding maps to reports**
- **Adding charts in the Report Definition rule form**
- **Removing charts in the Report Definition rule form**
- **Enabling users to view detailed chart data**
- **Exposing detailed chart data**

©2025 Pegasystems Inc

- **Styling charts in the application skin**

- Defining associations for map charts
- Adding charts in the Report Definition rule form
- Managing charts in Report Editor
- Editing a report

Defining associations for map charts

Ensure that maps display correct data by associating specific map regions with matching property values. For example, you can display traffic distribution on a map of New York by associating the map with the New York property value that includes that data.

For a specific map, set a property value that you use in your database for that region. You can provide a full region name, use an abbreviation, or provide a code.

> **Before you begin:**
>
> Ensure that you obtain the Maps component from Pega Marketplace. For more information, see Pega Maps.

1. In the header of Dev Studio, click **Configure** > **Reporting** > **Maps**.
2. In the **Map Type** field, press the Down arrow key, and then select the map of a specific region.

   > **For example:**
   >
   > To manage associations for countries worldwide, select the **World (Countries)** map.

3. In the **Class** field, press the Down arrow key, and then select the Class that you want to associate with the map.

©2025 Pegasystems Inc

4. In the **Ruleset** and **Version** fields, select the unlocked Ruleset and Ruleset version in which you want to save the map settings.
5. In the **Property value** column, update property values of specific regions. For example, for a map of France, you can enter `France` or `FR`.
6. Click **Save**.

- Adding maps to reports
- Reporting landing page

Adding maps to reports

Learn about the geographical distribution of specific events by displaying report data on a map. For example, you can use maps to display data about sales in specific regions, or incidents in specific states, and adjust your marketing strategy accordingly.

> **Before you begin:**
>
> Ensure that you associate map regions with specific property values.
>
> For more information, see Defining associations for map charts.

1. In the navigation pane of Dev Studio, click **Records**, and then click **Reports** > **Report definition**.
2. In the list of report definitions, click the report definition to which you want to add a map.
3. Click the **Chart** tab, and then click **Include chart**.
4. In the **Chart editor** section, click **All chart types**.
5. In the **Select Chart Type** modal dialog box, click **Map**, and then click **Submit**.
6. In the **Chart editor** section, in the **Map** field, press the Down arrow key, and then select the map with the regions that you require.
7. From the **Available columns** section, drag the columns with map regions onto the map.

©2025 Pegasystems Inc

- Adding charts in the Report Definition rule form

Adding charts in the Report Definition rule form

Create a graphical representation of report data by adding a chart to a report in the report definition rule form. For example, you can include a chart featuring specified data sets to make a report more readable.

> **NOTE:** Reports require summarized data to include a chart. The summarized data is a value that corresponds to particular events in the application. For example, a chart requires the total number of created cases, rather than a detailed list of the titles of cases.

1. In the navigation pane of Dev Studio, click **Records**, and then click **Reports** > **Report definition**.
2. In the list of report definitions, click the report definition that you want to edit.
3. On the **Query** tab, in the **Edit columns** section, in the **Summarize** column, ensure that the report contains at least one summarized column.
   The following values indicate summarized columns: **Count**, **Count distinct**, **Max** or **Min**.
4. If the report does not contain summarized columns, in the **Summarize** column, select how you want to summarize the values.
5. Click the **Chart** tab, and then click **Include chart**.
6. Customize the chart:

Choices	Actions
Select the chart type	a. Click **All chart types**. For more information about chart types, see Charts in reports. b. In the **Select Chart Type** modal dialog box, click a chart type that you want to include in the report.

©2025 Pegasystems Inc

Choices	Actions
	c. Click Submit.
Add or edit the chart title	a. Click Title. b. In the **Chart Title** window, enter the chart title, and then specify the font size and style. c. Click Submit.
Define the chart color options	**NOTE:** Available options depend on the chart type. a. Click Color settings. b. If you want to define the general chart color settings, click General color settings. c. If you want to define the colors that conditionally apply to the chart elements, click Conditional colors. d. If you want to define the colored lines that indicate value thresholds in the chart, click Threshold colors.
Adjust the chart general settings	**NOTE:** Available options depend on the chart type. a. Click General settings. b. To adjust the font settings, click Format options.

©2025 Pegasystems Inc

Choices	Actions
	c. To manage the placement of the chart legend, or manage the slider that filters values in the chart, click **Legend/Slider options**. d. To manage the chart display options available in the Report Viewer, click **Enable user commands**. For example, you can switch between two-dimensional and three-dimensional chart displays, or maximize the size of the chart for better readability.
Specify the chart content	a. From the **Available columns** section, drag the columns into the relevant chart drop zones. b. In a drop zone, next to a column name, access the data settings by clicking the **Properties** icon. For example, you can change the format and range of displayed values, or edit the axis names.

7. In the upper-right corner, click **Save**.
8. **Optional:** To preview the chart, run the report in the Report Viewer.
   For more information, see Running a report.

- Creating advanced reports
- Managing charts in Report Editor

©2025 Pegasystems Inc

Removing charts in the Report Definition rule form

Save space in reports by removing redundant charts from report definitions. For example, you can remove a chart when you delete the data that the chart uses, or in a report that presents open cases, you can remove a chart when the cases are resolved.

1. In the navigation pane of Dev Studio, click **Records**, and then click **Reports** > **Report definition**.
2. In the list of report definitions, click the report definition that contains the chart that you want to remove.
3. Click the **Chart** tab, and then in the **Chart editor** section, click **Remove Chart**.
4. In the upper-right corner, click **Save**.

- Adding charts in the Report Definition rule form

Enabling users to view detailed chart data

Provide application users with the option to analyze charts more conveniently by displaying data from embedded reports in the form of a list report. This approach is beneficial when users want to analyze specific data from a chart that contains multiple records.

After you enable the function, application users can view comprehensive information about application data that the system displays in a separate window.

Users can embed reports in their applications to gather relevant application data. For more information about embedding reports, see Adding a widget to your dashboard.

1. In the navigation pane of Dev Studio, click **Records**, and then click **Reports** > **Report definition**.
2. In the list of report definitions, click the report definition whose chart you want to embed in your application.
3. Click the **Chart** tab, and then, in the **Chart editor** section, click **General settings** > **Enable user commands**.

©2025 Pegasystems Inc

4. In the **Enable User Commands in Viewer** dialog box, select the **Show detailed data in embedded chart** check box.
5. Make the report visible in your application:
   a. Click the **Report Viewer** tab, and in the **User actions** section, select the **Display in report browser** check box.
   b. In the list that appears, select a category in which you want to store your report.
6. In the upper-right corner of the report definition, click **Save**.

- Adding charts in the Report Definition rule form
- Report Viewer
- Exposing detailed chart data

Exposing detailed chart data

Analyze the data in reports whose charts you embed in your application more conveniently by displaying comprehensive information in the form of a list report. For example, you can use this functionality to view and examine all data that the report references, not only the data that the chart displays.

1. If App Studio is not in Preview mode, in the header, click **Preview application**.
2. In the navigation pane of your application, click **Dashboard**.
3. In the section that contains your reports, next to the specific report whose data you want to display, click **Chart Actions** > **Show data**.

> **Result:**
>
> The system displays a new window that contains comprehensive report data.

- Enabling users to view detailed chart data

©2025 Pegasystems Inc

Styling charts in the application skin

Maintain consistent styling of report definition reports by using a chart theme. For example, you can set specific colors for all of the reports in your application, to avoid manually modifying each report.The charts in your application use an out-of-the-box theme until you create a new theme.

1. In the header of Dev Studio, click the name of the application, and then click **Skin**.
2. On the **Component styles** tab, click **Dynamic Layouts**.
3. In the left navigation pane, in the **Controls** section, click **Charts**.
4. Manage styles in your application:

Choices	Actions
Create new style	a. In the **My formats** tab, click **Add new style**. b. In the **Create a new format** dialog box, specify details of the new style, and then click **Submit**. c. Next to the **Charts theme file (JS)** field, open the active style by clicking the **Target** icon. **Result:** The style opens in an editor. d. On the **Main** tab of the style editor, customize attributes of the style to match your business requirements. **For example:** Change the font in your charts to OpenSans by

©2025 Pegasystems Inc

Choices	Actions
	setting the value of the *baseFont* attribute to OpenSans Add the *#FF0000* color to the palette in your charts by adding `#FF0000` to the *paletteColors* attribute. e. In the upper-right corner of the rule, click **Save**.
Customize existing style	a. In the **My formats** tab, select the style that you want to update. b. Next to the **Charts theme file (JS)** field, open the active style by clicking the **Target** icon. **Result:** The style opens in an editor. c. On the **Main** tab of the style editor, customize attributes of the style to match your business requirements. **For example:** Change the font in your charts to OpenSans by setting the value of the *baseFont* attribute to OpenSans Add the *#FF0000* color to the palette in your

©2025 Pegasystems Inc

Choices	Actions
	charts by adding `#FF0000` to the *paletteColors* attribute. d. In the upper-right corner of the rule, click **Save**.

5. In the upper-right corner of the screen, click **Save**.

- Adding charts in the Report Definition rule form

Associations

By default, reports include data from one Class only. You can extend the scope of reports by establishing a relationship between multiple classes based on matching values in pairs of properties. You can use the association rules by referencing specific data from other classes to make your reports more comprehensive.

In reports that include associations, you can include properties from classes that are referenced in the association. For example, an application for claims stores customer information in the *OSP-Data-Customer* Class, whereas the *OSP-Claims-Requests* Class holds data about customer claims. You can create a report that includes data from both classes by creating an association rule.

Association types

Pega Platform provides users with simple and advanced associations. Use simple associations for joining two classes, and apply advanced associations when you want to specify multiple join conditions. You can reuse each association rule among multiple reports. For example, you can associate a department name with the department budget, and then include the association in your financial report.

©2025 Pegasystems Inc

Joins

By default, report definitions include data from only the Class to which the report applies. By using the Class join functionality, you can make your reports more comprehensive.

In comparison with association rules, joins are not reusable across multiple report definitions, and apply only to the report that you edit. For example, if your report about insurance policies applies to the *Work-* Class, you can include specific data from different classes by joining the classes together.

- **Creating simple associations between classes**
- **Creating advanced associations between classes**
- **Referencing association rules in a report**
- **Viewing associations that are available for reporting**
- **Creating Class joins in reports**
- **Using join filters to combine data in reports**
- **Using aggregation to combine data in reports**

Creating simple associations between classes

Save time and create more comprehensive reports by creating relationships between classes based on their matching properties. For example, you can associate a department name with the department budget, and then include the association in your financial report.

> **NOTE:** Do not set foreign keys at database level, as it, for example, might affect migration from on-premises deployment to Pega Cloud.

©2025 Pegasystems Inc

1. In the header of Dev Studio, click **Create** > **Data Model** > **Association**.
2. Create an association record:
   a. In the **Label** field, enter a title or description for the association.

   > **NOTE:**
   >
   > The words `SELECT`, `ORDER`, and `JOIN` are restricted.

   b. In the **Context** section, in the **Apply to (Class)** field, press the Down arrow key, and then select a Class to which the association applies.
   c. In the **Add to Ruleset** list, select the Ruleset in which you want to include the association.
   d. In the upper-right corner of the tab, click **Create and open**.
3. Define associations between classes:
   a. On the **Data Access** tab, in the **Target type** list, select the other Class to which you want to link.
      The **Selected type** and **Target type** fields show the associated classes.
   b. In the **Foreign key** list, select a property of the **Selected type** Class that you want to link with the property of the **Target type** Class.
   c. In the **Key** list, select a property of the **Target type** Class that you want to link with the property of the **Selected type** Class.
4. In the upper-right corner of the tab, click **Save**.

> **What to do next:**
>
> By associating properties of two classes, you create an association rule that you can include as a column in your report to improve the process of creating reports. For more information, see Adding columns to reports.

©2025 Pegasystems Inc

Creating advanced associations between classes

Improve the functionality of the Class association feature by providing users with advanced options for creating complex associations between classes. For example, you can associate two classes whose properties do not share a common value by joining them with a separate Class to fetch the data from all three classes.For example, when you associate the *Work-* and the *Data-Admin-OrgUnit* classes, you can define a corresponding join between them by adding the *Assign-Worklist* Class to the association. Advanced associations provide the option to join the *Work-* Class with the *Assign-Worklist* Class, and then the *Assign-Worklist* Class with the *Data-Admin-OrgUnit* Class.

> **Before you begin:**
>
> Create an association record by following steps 1 through 2 in Creating simple associations between classes.

1. In Dev Studio, at the top of the **New** tab, click **Convert to advanced configuration**, and then confirm the conversion by clicking **Submit**.
2. In the **Class association** section, in the **Class name** field, specify the Class that you want to associate with the context Class.
   You specify the context Class in the **Apply to (Class)** field when you create an association record.
3. Create a Class join between classes:
   a. In the **Class join** section, click **Add Class join**.
   b. In the **Prefix** field, enter a short alias for the Class that you want to join with the context Class.
      The prefixes help the system to distinguish between classes and ensure that classes are joined accurately.

      For example, enter A.

   c. In the **Class name** field, enter the Class that you want to join with the context Class.

©2025 Pegasystems Inc

> **For example:**
>
> Enter *Assign-Worklist*.

a. Click **Edit conditions**.
b. In the **Column** field, enter the prefix of the Class that you want to join, and then specify any property of that Class.
   For example, if you set the prefix as A, enter `A.pyLabel`.
c. In the **Value** field, enter a property of the context Class, and click **Submit**.

> **Result:**
>
> The system creates a join between the context Class and the *Assign-Worklist* Class.

4. If your association is complex, and requires more than one join to fetch the required data, create additional joins by repeating step 3 to create a connection between the context Class, and the secondary Class.

> **For example:**
>
> Associate the *Assign-Worklist* Class with the *Data-Admin-OrgUnit* Class to fetch data from the context Class and the targeted *Data-Admin-OrgUnit* Class.

5. In the **Type** list, select the type of join to use.
   For example, select **Only include matching rows** to include rows that contain common data for both classes.
6. In the upper-right corner of the **New** tab, click **Save**.

Referencing association rules in a report

Make your reports more comprehensive by automatically including data from multiple classes at once. For example, you can reference the user identification number with the list of tasks assigned to that user to improve the performance analysis of that user.

©2025 Pegasystems Inc

Before you begin:

Create an association rule. For more information, see Associations.

1. In the navigation pane of Dev Studio, click **Records**, and then click **Reports** > **Report definition**.
2. In the list of report definitions, click the report definition that you want to edit.
3. On the **Query** tab, in the **Edit columns** section, click **Add column**.
4. In the **Column source** field, enter the name of the association rule that you want to reference in your report.
5. Click **Save**.

Result:

Your report now references association rules, which you can view and modify in the **Data Access** tab in the report definition.

Viewing associations that are available for reporting

Understand the fields that you can use for reports by viewing the associations in your application. Use the Visual Data Model to view how simple associations defined in your application relate to the different classes of your application.

When you are using Pega BI reporting mode, broken relationships, that is, relationships that do not have an association, are displayed using red lines. Ensure that your application has a complete Data Model by creating associations for the broken relationships.

1. In App Studio, click **Data** > **Data Model**.
2. Click **Actions** > **Reporting schema mode**.

- Associations

©2025 Pegasystems Inc

Creating Class joins in reports

Extend the scope of your report by displaying data from various classes. For example, if your report about insurance policies applies to the *Work-* Class, you can include specific data from different classes by joining the classes together.

By default, report definitions include data from only the Class to which the report applies. By using the Class join functionality, you can make your reports more comprehensive.

In comparison with association rules, joins are not reusable across multiple report definitions, and apply only to the report that you edit.

1. In the navigation pane of Dev Studio, click **Records**, and then click **Reports** > **Report definition**.
2. In the list of report definitions, click the report definition that you want to edit.
3. Click the **Data access** tab.
4. Create a join between classes:
   a. In the **Class joins** section, click **Add Class join**.
   b. In the **Prefix** field, enter a short alias for the Class that you want to join with the Class to which your report definition applies.
   The prefixes help the system to distinguish classes and ensure that classes are joined accurately.

   > **For example:**
   >
   > Enter A.

   c. In the **Class name** field, enter the name of the Class that you want to join.
   d. Click **Edit conditions**.
   e. In the **Column** field, specify a property of the Class that you want to join.

©2025 Pegasystems Inc

> **For example:**
>
> If you set the prefix as A, enter `A.pyID`.

f. In the **Value** field, enter a property of the Class to which your report definition applies, and then click **Submit**.

> **For example:**
>
> Enter `.pyResolvedUserID` to create a join between data in *A.pyID* and *.pyResolvedUserID* classes. The join provides the assignment ID and the name of the responsible assignee.

g. In the **Type** list, select the type of join to use.

> **For example:**
>
> Select **Only include matching rows** to include rows that contain common data for both classes.

5. Click the **Query** tab, and in the **Edit columns** section, click **Add column**.
6. In the **Column source** field, enter the prefix of the joined Class, and then select the specific properties that you want to include in the report.

> **For example:**
>
> Enter `A.pyID` to include the data of the joined *A.pyID* and *.pyResolvedUserID* classes.

7. In the upper-right corner of the report definition, click **Save**.

Using join filters to combine data in reports

Save time and enhance the process of creating reports by referencing data from other reports. You can use another report as a filter to include or exclude specific data in your report.The report that you reference becomes a subreport, which means that it

©2025 Pegasystems Inc

provides data to your main report. By using join filters, you can reference report definitions from multiple classes. For example, in a financial report of the company, you can include a subreport with a property that fetches the most recent expenses of employees. This approach provides a way to filter the contents of the main report to display how the recent expenses affect the company budget.

> **Before you begin:**
>
> Create a report definition to use as a subreport. For more information, see Creating advanced reports.

1. In the navigation pane of Dev Studio, click **Records**, and then click **Reports** > **Report definition**.
2. In the list of report definitions, click the report definition that you want to use as the main report.
3. On the **Query** tab, in the **Edit columns** section, define the contents of your main report.
   For more information, see Report definitions.
4. Add the subreport to your main report:
   a. On the **Data Access** tab, in the **Sub-reports** section, click **Add sub-report**.
   b. In the **Prefix** field, enter a unique alias for the subreport that you want to reference in the main report.
   The system uses prefixes to reference correct subreports.

   > **For example:**
   >
   > Enter `SUBREPORT1`

   c. In the **Class name** field, specify the Class to which the subreport applies.
   d. In the **Name** field, select the subreport.
5. Define how you want the subreport to function:
   a. Click **Configure**, and at the top of the **Configure sub-report use** dialog box, specify how many rows from the subreport to display.

©2025 Pegasystems Inc

> **For example:**
>
> Select More than 1 row if you want to fetch address data that includes multiple rows, such as the city name or postal code.

b. **Optional:** To specify which fields in the main report reference the subreport data, in the **Where will you use this sub-report?** section, select a specific check box.

> **For example:**
>
> Select the Columns to include check box to reference the data in the **Edit columns** section of the Query tab.

By default, the Right hand side of the filter conditions check box is selected.

6. In the **How will you join rows in the reports?** section, specify the join conditions:
   a. Click Add join condition.
   b. In the Sub-report column list, select the subreport property that you want to match with the main report.

> **For example:**
>
> Select Label.

   c. In the Value field, select a property of the main report that matches the subreport property.
   d. In the Do not match rows list, select how the system joins data.

> **For example:**
>
> Select Include only matching rows from both reports to fetch matching records from the subreport and the main report.

   e. Click Submit.
7. In the upper-right corner of the report definition, click Save.

©2025 Pegasystems Inc

What to do next:

Perform a test-run of the report by clicking **Actions** > **Run**.

Using aggregation to combine data in reports

Manage reports that use complex logic by including aggregated data from other reports. In your main report, you can use aggregate functions, such as `Is equal` or `Is less than`, to include specific data from other reports.For example, in your main report about managers, you can reference a subreport that fetches the number of all employees who directly report to those managers. You can set a filter condition to display only the managers who have more than 10 subordinates.

Before you begin:

Create a report definition to use as a subreport. For more information, see Creating advanced reports.

1. Combine the subreport data with the main report by following steps 1 through 6 in Using join filters to combine data in reports.
2. On the **Query** tab, in the **Edit filters** section, create a filter to display specific data:
   a. Click **Add filter**, and in the **Column source** field, reference the subreport property that fetches the target data.

      For example:

      Enter `SUBREPORT1.Subordinates_Count` to reference the *Managees Count* property of the subreport.

   b. In the **Relationship** list, select a relational operator for the filter condition.

©2025 Pegasystems Inc

> **For example:**
>
> Select **Greater than** to filter the results to display values greater than those specified in the filter condition.

c. In the **Value** field, specify a comparison value for the relational operator.

> **For example:**
>
> Enter `10` to display properties whose values are greater than 10.

d. Select the **Remove duplicate rows** check box to discard rows that contain duplicate values.

3. In the upper-right corner of the report definition, click **Save**.

> **Result:**
>
> The report displays the data that matches the requirements specified in the filter conditions.

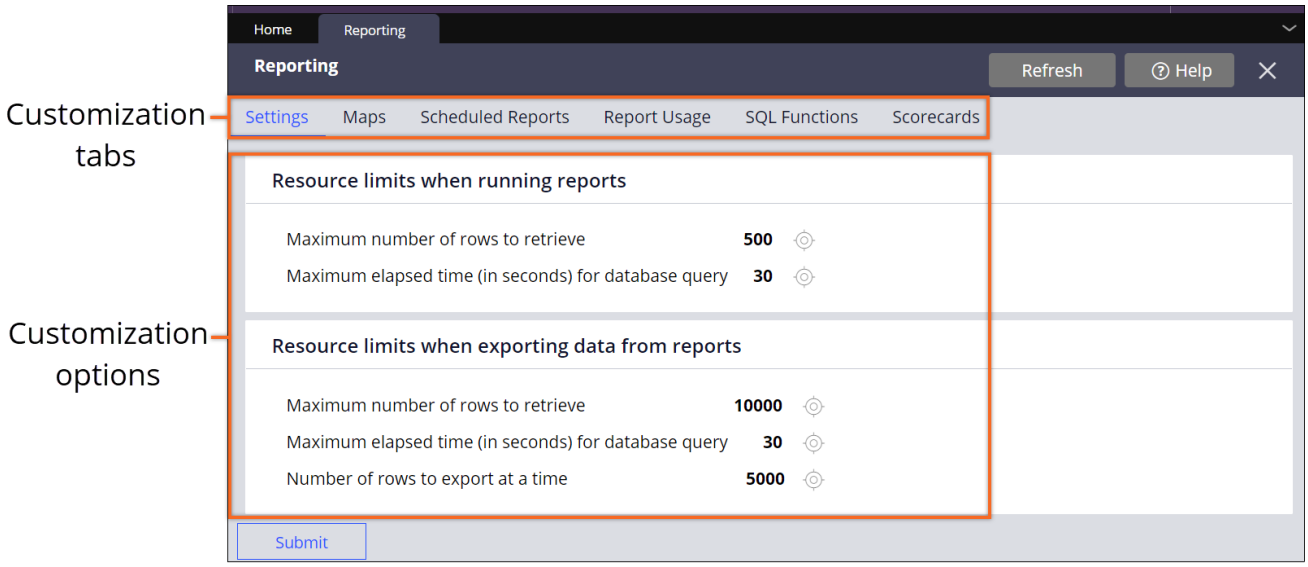
> **What to do next:**
>
> Perform a test run of the report by clicking **Actions** > **Run**.

Reporting landing page

In Pega Platform™ you can conveniently manage all reports within your application by accessing Reporting landing page. For example, you can specify capacity limits for all reports to improve system performance, or update the scheduled reporting tasks to adapt scheduled reports to your evolving business needs.
You can access the landing page in the header of Dev Studio by clicking **Configure** > **Reporting**, and then selecting the required area to customize.

The following figure shows the landing page with all customization tabs:

©2025 Pegasystems Inc



Reporting landing page

The landing page features the following tabs:

Settings

Manage the resources that are required to run reports and export report data.

For example, you can specify the maximum duration for database queries.

Maps

Ensure that maps display appropriate regions by associating regions with property values.

Scheduled reports

Manage scheduled reports.

For example, you can view and update task details, delete tasks, and perform test runs.

Report usage

View statistics about the usage and performance of reports in your current application.

©2025 Pegasystems Inc

SQL functions

View information about all available alias rules for SQL expressions.

For example, you can edit a function that permits only integer values between 1 and 100.

Scorecards

Manage available scorecard rules in your application.

For example, you can create new scorecards.

- **Setting resource limits for reporting**
- **Including report metadata in Excel reports**

Setting resource limits for reporting

Improve system performance by controlling the amount of resources that are required to run reports. You can set system-wide capacity limits for all reports by specifying parameters such as the maximum duration for database queries.

1. In the header of Dev Studio, click Configure > Reporting > Settings.
2. In the **Resource limits when running reports** and **Resource limits when exporting data from reports** sections, next to the specific option, click the Open the rule icon.

   > **For example:**
   >
   > Next to the Maximum number of rows to retrieve row, click Open the rule to manage the number of rows that the system exports.

3. In a new tab that the system displays, in the **Value per production level** section, enter values for specific production levels.

©2025 Pegasystems Inc

> **For example:**
>
> In the **Production** row, enter `5000` to export the specified number of rows.

The production level of your system is a property in the System data instance (*Data-Admin-System* Class) with a value between 1 (experimental system, low security) and 5 (production).

4. In the upper-right corner of the screen, click **Save**.

Including report metadata in Excel reports

Maintain clarity in the reports that you export as Microsoft Excel files by specifying how the files display additional information about the reports. For example, for a more comprehensive data analysis, you can include the metadata of your report on the Data tab of the spreadsheet.The system generates the Excel files based on the values that you specify in the *pyEnableExcelInfoWorksheet* When rule. By default, the **Data** tab includes report results, and the **Info** tab includes the following metadata: report title, used filters, date, and the number of results. You can save a copy of the *pyEnableExcelInfoWorksheet* When rule in your Ruleset, and then decide how the system generates reports that are in the Ruleset.

1. In the navigation pane of Dev Studio, click **Records**.
2. Expand the **Decision** category, and then click **When**.
3. Create a copy of the *pyEnableExcelInfoWorksheet* rule:
   a. In the list of rules, select the *pyEnableExcelInfoWorksheet* rule.
   b. On the **pyEnableExcelInfoWorksheet** tab, click the **Save as** button.
   c. On the **New** tab, specify the details of the rule.

   > **For example:**
   >
   > In the **Add to Ruleset** list, select the Ruleset in which you want to save the rule.

©2025 Pegasystems Inc

   d. In the upper-right corner of the tab, click **Create and open**.
4. On the main tab of the new When rule form, in the **When** section, click the existing value, and then click **Actions** > **Edit**.
5. In the **Condition** dialog box, specify whether to include the report's metadata in your Excel file.
   - To include the report metadata in the **Info** tab, enter `true`.
   - To display only the **Data** tab, enter `false`.
6. Click **Submit**, and then, in the upper-right corner of the tab, click **Save**.

Excluding items from Report Browser

Maintain clarity in your application by hiding reports and categories that you do not need at the moment. For example, if a report category does not include any reports, you can hide it in the Report Browser to save space and avoid confusion.

1. In the navigation pane of Dev Studio, click **Records**.
2. Expand the **Reports** category, and then select the type of item that you want to exclude from Report Browser.
   - To exclude a specific report category, click **Category**.
   - To exclude a specific report, click **Shortcut**.
3. Select a specific item that you want to exclude.

   > **For example:**
   >
   > Click **pyPerformanceAnalysis** to hide the “Analyze Performance” category.

4. Save a copy of the rule in your application Ruleset:
   a. In the upper-right corner of the rule tab, click **Save as**.
   b. On the **New** tab, specify the details of the rule.

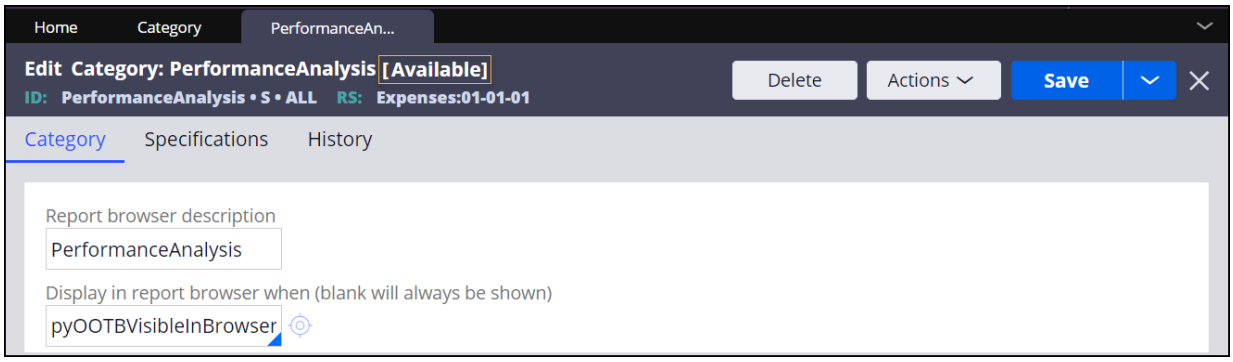
      > **For example:**
      >
      > In the **Context** section, in the **Add to Ruleset** list, select the Ruleset to which you want to save the rule.

©2025 Pegasystems Inc

c. In the upper-right corner of the **New** tab, click **Create and open**.

5. On the main tab of the new rule form, next to the item name, click **Available**, and then, in the **Availability** list, select **Blocked**.

Location of the Available button



6. Click **OK**.
7. Save the rule by clicking **Save**.

Creating an advanced report category

Organize reports more effectively by creating report categories in Dev Studio. For example, you can create a custom category that contains information about open cases so that you can conveniently analyze the remaining workload.By using Dev Studio to create report categories, you gain access to more advanced configuration options than in Report Browser. For example, you can specify which Access Group can see your report category, or add the category to a specific application.

1. In the header of Dev Studio, click **Create** > **Reports** > **Category**.
2. In the **Category Record Configuration** section, enter the category details:
   a. In the **Label** field, enter a short description of the category.
   b. **Optional:** To edit the unique identifier of the category, in the **Identifier** field, click **Edit**, and then enter a new name.

      By default, the **Identifier** field inherits the name from the **Label** field. A valid identifier name starts with a letter, and consists of letters, digits, and underscores.
   c. In the **Owner Type** list, select who can see the category.

©2025 Pegasystems Inc

For example, select **Shared**, and then specify the access group with which you want to share the category.

d. In the **Context** section, select the check box of the specific application in which you want to include the category.

e. In the **Add to Ruleset** list, select the Ruleset in which you want to include the category.

3. **Optional:** To associate the category with a work item, in the **Work item to associate** field, press the Down arrow key, and then select a work item.
4. Click **Create and open**, and then click **Save**.
5. **Optional:** To view all report categories in your application, in the left navigation pane, click **Records** > **Reports** > **Category**.

- **Managing report categories**

Managing report categories

Control the way that Report Browser displays specific report categories by customizing the categories. By modifying existing report categories, you can conveniently adapt to new requirements without the need to create additional categories. For example, you can change the name of a category, and specify who can see the category in Report Browser.

1. In the navigation pane of Dev Studio, click **Records**, and then click **Reports** > **Category**.
2. Click the name of the category for which you want to configure the display.
3. Manage how Report Browser displays information about the report category:

Choices	Actions
Edit the category label	On the **Category** tab, in the **Report Browser description** field, enter the name of the category.

©2025 Pegasystems Inc

Choices	Actions
Specify when Report Browser can display the category	a. On the **Category** tab, in the **Display in report browser when** field, specify the when condition that Report Browser uses to display the category. **NOTE:** This field is available for standard and shared categories only. For example, you can leave the field empty or enter `pyOOTBVisibleInBrowser` to always include the category in Report Browser. b. **Optional:** To edit the when condition, click **Open**, and then manage the condition settings.

4. In the upper-right corner of the screen, click **Save**.

Creating a report shortcut

Save time when searching for report definitions in your application by creating shortcuts that lead to these reports. In this way, you can effortlessly access report definitions that you need at the moment.

NOTE: If the Report Definition is built on abstract classes, such as *Work-*, and run from a shortcut rule, the system runs the report in the Current Work Pool context.

©2025 Pegasystems Inc

Before you begin:

Create a report, and then make it available in the Report Browser. For more information about creating reports, see Creating a report.

1. In the header of Dev Studio, click **Create** > **Reports** > **Shortcut**.
2. In the **Label** field, enter a title or description for the record.
3. **Optional:** To edit the shortcut name, in the **Identifier** field, click **Edit**, and then provide a new name.

   By default, the **Identifier** field inherits the name from the **Label** field. A valid identifier name starts with a letter, and consists of letters, digits, and underscores.
4. Define who can see the shortcut:

Choices	Actions
Display the shortcut in one of the standard categories	a. In the **Owner Type** list, select **Standard**. b. **Optional:** To edit the owner, in the **Owner** field, enter an owner name. By default, the **Owner** field value is ALL.
Display the shortcut in one of the shared categories	a. In the **Owner Type** list, select **Shared**. b. In the **Owner** list, select an Operator ID.
Display the shortcut in one of the personal categories	a. In the **Owner Type** list, select **Personal**. b. In the **Owner** list, select an Operator ID.

5. In the **Category name** list, select a shortcut category.
6. In the **Context** section, select Branch and Ruleset to which the shortcut applies.

©2025 Pegasystems Inc

7. **Optional:** To associate a work item with the shortcut, in the **Work item to associate** field, press the Down arrow key, and then select a work item.
8. Click **Create and open**.
9. In the **Edit Shortcut** tab, enter the shortcut details:
   a. In the **Shortcut title** field, enter a unique label for the shortcut.

   > **NOTE:** The system displays the shortcut label instead of the report definition title, which is visible after you open the shortcut.

   b. In the **Report description** field, enter a description for the report.
   c. In the **Rule type** list, select the report type to which the shortcut applies.
   d. In the **Applies to** field, press the Down arrow key, and then select an entity to which the shortcut applies.
   e. In the **Rule key** field, press the Down arrow key, and then select a rule key to which the shortcut applies.
10. **Optional:** To add a shortcut parameter, click **Add parameter**, and then specify the parameter details.
11. Click **Save**.
12. **Optional:** To view all of the report shortcuts for the application, click **Records** > **Reports** > **Shortcut**.

- **Managing reports shortcuts**

Managing reports shortcuts

Manage report shortcuts by renaming, moving, copying, or deleting them. The Report Browser displays manageable report shortcuts in the form of tiles.

1. If App Studio is not in Preview mode, in the header, click **Preview application**.
2. In the header of App Studio, select the Portal in which you manage reports.
3. In the header of Dev Studio, select the Portal in which you manage reports.

©2025 Pegasystems Inc

4. In the left navigation pane, open the Report Browser by clicking **Reports**.
5. Hover over a report, and then click the **Actions** icon.
6. Manage the report by performing one of the following actions:

Choices	Actions
Open the report	Click **Open**.
Edit the report details	a. Click **Edit**. b. To edit the report title, in the **Report title** field, enter a new title. c. To edit the report description, in the **Report description** field, enter a new description. d. Click **Submit**.
Move the report to a different category	a. Click **Move**. b. In the **Report category** list, select the category to which you want to move the report. c. Click **Submit**.
Copy the report shortcut	a. Click **Copy**. b. To edit the report title, in the **Report title** field, enter a new title. c. To edit the report description, in the **Report description** field, enter a new description. d. In the **Category type** list, set the report to public or private. e. In the **Report category** list, select the report category. f. Click **Submit**.
Remove a tile from the list	Click **Remove**.

©2025 Pegasystems Inc

Choices	Actions
Delete the report	a. Click **Delete**. b. Click **Submit**.
Schedule a report run	a. Click **Schedule**. b. Under the **Task Scheduling** section, schedule a run, and then click **Submit**. For more information about scheduling reports, see Scheduling reports.

- Adding columns to reports
- Creating a report
- Running a report

Reviewing usage report statistics

Analyze report definition usage and performance in the current application to determine which reports users do not use, and then remove these reports to maintain only the required reports.

By default, Pega provides several usage reports, but you can also create custom reports. For more information, see Creating usage reports.

1. In the header of Dev Studio, click **Configure** > **Reporting** > **Report Usage**.
2. In the **Name** column, click the report that you want to run.
3. Click **Actions** > **Run**.

> **Result:**
>
> A window displays the results in the Report Viewer.

- **Creating usage reports**

©2025 Pegasystems Inc

- **Customizing usage reports**

Creating usage reports

Gain insight into the usage and performance statistics of your application reports by creating custom usage reports. For example, you can create a report that displays a list of errors that you can use to identify issues with your application, and ease the troubleshooting process.

1. In the header of Dev Studio, click **Configure** > **Reporting** > **Report Usage**.
2. Click **Add Report**.
   A report definition rule form opens in a new tab.
3. Complete the report definition rule form by following 2.a through 4 in Creating advanced reports.

> **NOTE:** Ensure that you apply the report definition to the *Log-ReportStatistics* Class.

> **Result:**
>
> A report shortcut appears in the list of reports of the **Report Usage** landing page.

- Reviewing usage report statistics

Customizing usage reports

Display data in a clear and organized way by managing the appearance and content of usage reports. For example, you can edit the columns of a report to present the data as charts.

1. In the header of Dev Studio, click **Configure** > **Reporting** > **Report Usage**.
2. In the **Name** column, click the report that you want to customize.
   The report rule form opens in a new tab.

©2025 Pegasystems Inc

3. Customize the report by filling out the report rule form.
   For more information, see Report definitions.

   For example:

   - To manage filter conditions, on the **Query** tab, fill out the **Edit filters** section.
   - To present the data as a chart, on the **Chart** tab, click **Include chart**.
   - To manage the way the Report Viewer displays the report, on the **Report Viewer** tab, specify the display details.

4. Click **Save**.

- Reviewing usage report statistics

Specifying a data source for reports

Improve your system performance by changing the default database, from which reports obtain data, to a custom database. By creating and designating a separate database as the source of your reports, you reduce the time that is required to run reports.The custom database uses database tables to retrieve Class data.

Before you begin:

Work with a database administrator to create the reports database and to set up the database replication.

1. Enable Pega Platform to access your database by creating a database table:
   a. In the navigation pane of Dev Studio, click **Records**.
   b. Expand the **SysAdmin** category, and then click **Database Table**.
   c. In the upper-right corner of the window, click **Create**.
   d. In the **Create Database Table** tab, enter the short description for the table, specify the Class to which the database table belongs, and then click **Create and open**.

©2025 Pegasystems Inc

e. In the new database table rule form, provide the details for your database table, and then click **Save**.

> **For example:**
>
> In the **Reports database** field, enter the name of the custom database from which you want to obtain data.

2. Change the database from which your reports obtain data:
   a. In the navigation pane of Dev Studio, click **Records**.
   b. Expand the **Reports** category, and then click **Report definition**.
   c. In the list of report definitions, click the report definition for which you want to edit the source.

> **NOTE:** The report definition sources data from the database table that belongs to the same Class as the report definition.

   d. On the report definition rule form, click the **Data Access** tab.
   e. In the **General data access settings** section, in the **Reporting database** list, select **Prefer reporting data source if defined**.
   f. In the upper-right corner of the window, click **Save**.

- **Disabling usage of a reports database**

- Creating database data instances
- Database tables

Disabling usage of a reports database

You can disable a report from obtaining data from a reports database, even if the **Prefer reporting data source if defined option** is selected on the report definition rule form.

©2025 Pegasystems Inc

To disable use of a reports database, add the following entry to your `prconfig.xml` file:

```
<env name = "database/useReportingDatabase" value="false" ↗
```

The default value is `true`.

- Specifying a data source for reports

Data extraction

Create a broad view of your business by integrating Pega Platform data with data from other systems of record using Business Intelligence Exchange (BIX). BIX extracts transactional data from the Pega database into formats suitable for exporting to a reporting data warehouse, operational data store (ODS), or other downstream process. The BIX extraction process extracts data into .xml, .csv, and database schema formats.

Pega offers BIX as an optional add-on product. Contact your Pega Account Executive to ensure that your license includes BIX access.

Business Intelligence Exchange (BIX) is a ruleset, which extraction processes can be used to extract Class data from your application. BIX provides the extract functions of an ETL (extract, transform, and load) utility by using the *Pega-BIX* ruleset, which supports the *Rule-Admin-Extract* rule type. Data is extracted and exported in a format suitable for use in popular business intelligence applications, such as data warehouses. BIX is a batch processing tool that extracts data from one class at a time. You can use it to do a full extract or an incremental extract based upon when a target property changes.

- **BIX extraction overview**
- **Extract Rules overview**
- **Configuring BIX real-time data extraction**
- **Using the Job Scheduler rule to extract data with BIX**
- **Unique run identifier**

PEGA ©2025 Pegasystems Inc

- **Overview of BIX extractions in Pega Cloud environments**
- **Overview of BIX extractions in on-premises systems**
- **Running a BIX command-line extraction in on-premises systems**
- **BIX logging**
- **Optimizing BIX data extraction performance**
- **Business Intelligence Exchange - FAQ**
- **Troubleshooting BIX for on-premises environments**

BIX extraction overview

Extract data from your Pega Platform application using the Business Intelligence Exchange (BIX) *Extract* rule.

After enabling the built-in BIX Ruleset to your application, BIX extractions are made available in Pega Platform. For details, see Enabling the BIX Ruleset.

A BIX extraction, also referred to as an extraction or extraction process, extracts and exports Pega Platform Class data in XML, CSV, or database schema format. The application accomplishes the BIX extraction process through an Extract or Job Scheduler rule. For more information, see Extract Rules overview.

> ⚠ **IMPORTANT:** Pega Platform™ does not support extracts to Snowflake DB as the output destination for Extract Rules. Ensure that you define a different output for the extracted files.

Each Extract rule extracts data from a single Class. BIX extractions may be run manually from the Extract rule or automatically through the Job Scheduler rule. All extractions must occur from the same database. A single extraction process cannot concurrently extract from multiple databases or different database instances on the same machine.

PEGA ©2025 Pegasystems Inc

Each extraction process can have only a single data source. You can run different extraction processes, extracting from multiple Pega Platform databases, concurrently.

> **NOTE:** If you run Pega Platform in a client-managed cloud environment, refer to the on-premises BIX documentation.

Functionality	Pega Cloud Services	On-premises and client-managed cloud
BIX extractions run through the Extract rule	◻	◻
BIX extractions run through Job Scheduler rule calling *pxExtractDataWithArgs* for multiple extraction processes	◻	◻
Command-line processes run from any server with privileges to the source database and destination database		◻
BIX extractions pulled to Pega Cloud Services file repository using Pega SFTP Service	◻	

©2025 Pegasystems Inc

Functionality	Pega Cloud Services	On-premises and client-managed cloud
BIX extractions pushed to your enterprise SFTP service using an FTP listener	◻	

Functionality	Pega Cloud Services	On-premises and client-managed cloud
CSV & XML extraction	• Extractions have default location of Pega Cloud File Storage repository. Integration with SFTP allows clients to pull extractions from repository to enterprise SFTP service. • Extractions configured with dynamic system settings can incorporate a listener to push data generated from local utility nodes to FTP location.	Extractions default to local filepath.
Database Schema	• Extractions require an external database	• Extractions require an external database

©2025 Pegasystems Inc

Functionality	Pega Cloud Services	On-premises and client-managed cloud
	instance with read-only privileges created with a Java Database Connectivity (JDBC) URL. • Extractions to database use a data description language (DDL) to define database schema. The Extract rule form provides a link to share with your database administrator for the DDL schema.	instance with read-only privileges created with a Java Database Connectivity (JDBC) URL. • Extractions to database use a data description language (DDL) to define database schema. You provide a local filepath to create the DDL schema to share with your database administrator.

Platform restrictions

- The BIX Ruleset version you enable in your application Ruleset must reflect your Pega Platform version, including after you upgrade Pega Platform. For more information about enabling BIX with the correct Ruleset version, see Enabling the BIX Ruleset and Upgrading BIX.
- For database schema extract files: The BIX extraction destination database must be supported by your Pega Platform version.
- The destination database requires a Java Virtual Machine (JVM) running a Java version in parity with the Pega Platform Java version. For more information about external database connections, see Creating and updating external relational database instances with JDBC URLs.

PEGA ©2025 Pegasystems Inc

- If you are using ANT for on-premises command-line extractions, it must be version 1.5 or higher. Version 1.7.1 or higher is recommended.

- **Enabling the BIX Ruleset**

Enabling the BIX Ruleset

Pega Platform includes the Business Intelligence Exchange (BIX) Ruleset with standard installations as an optional add-on product. Contact your Pega Account Executive to ensure that your license includes BIX access. Your application disables the BIX Ruleset by default. Enable the Extract rule by adding the BIX Ruleset to your application Ruleset after purchase.

If an extraction rule is defined on a class with XML output type and the **Get all properties** checkbox is selected, BIX extracts all the properties present in the specified class's *pzPVStream* column. This occurs if the extraction rule is defined on a class that contains a BLOB. However, if the extraction rule is defined on a blobless class, BIX only adds the key columns and exposed columns such as *committedDatetime* and *pxObjClass*.

For more information, see 5.d in Creating and running an Extract rule.

1. In the navigation pane of Dev Studio, click **Records**.
2. Expand the **Application Definition** category, and then click **Application**.
3. Click your application rule.
4. In the **Application rulesets** section, click **+ Add Ruleset**.
5. Enter `Pega-BIX: xx-xx-xx` in the Application rulesets field where “xx” matches your version of Pega Platform.
   a. To see a list of BIX version, click the **properties** icon to open the BIX-Ruleset.
6. Click **Save**.
7. Log out and log back in.
   BIX is displayed as the **Extract** selection under **SysAdmin** in the Rules Explorer.

- **Upgrading BIX**

©2025 Pegasystems Inc

- Creating and running an Extract rule

Upgrading BIX

If you previously enabled the Business Intelligence Exchange (BIX) Ruleset and you have upgraded Pega Platform™, update the BIX Ruleset for your application.

Pega recommends to keep your environments current on the latest Pega Platform full release or a patch due to important security and reliability improvements delivered with each new version. Additionally, when you update Pega Platform, ensure that BIX version is also up to date.

> **NOTE:** You can add the new version of BIX on top of your existing version.

1. In the navigation pane of Dev Studio, click **Records**.
2. Expand the **Application Definition** category, and then click **Application**.
3. Click your application rule.
4. In the **Application rulesets** section, click **+ Add Ruleset**.
5. Select `Pega-BIX: xx-xx-xx` to the list of Application rulesets, where “xx” matches the Pega Platform version that you are currently running.
6. Click **Save**.
7. Log out and log back in.
   After upgrading the built-in BIX Ruleset in Pega Platform, the *Extract* selection under **SysAdmin** in the Rules Explorer becomes available to create and run BIX extractions.

> **What to do next:**
>
> For Pega Cloud Services clients, continue to Procedures for BIX extractions in Pega Cloud environments.
>
> For on-premises clients, you may need to accomplish the following tasks:

©2025 Pegasystems Inc

- If you were `pegarules.keyring` to extract the rules, perform the following steps:
  1. For on-premises deployments, regenerate the `pegarules.keyring` file.
  2. Copy the *Extract* rule XML to the appropriate path.
  3. Validate the ANT build file with the sample build file provided in the distribution package.

- Reconfigure your stand-alone command-line environment:

  1. Download the BIX distribution archive and expand it into a directory.
  2. Configure a JVM environment with access to the Pega Platform jar files and the appropriate database driver files
  3. Configure the `prconfig.xml`, `prbootstrap.properties`, and `prlog4j2.xml` files provided in the configuration directory of the BIX distribution to specify the source and target databases and split schema setup.
- Update custom work tables if they do not have the *pxCommitDateTime* and *pxSaveDateTime* columns and are using *pxUpdateDateTime* for incremental extracts. Incremental extracts are done using *pxCommitDateTime*. For more information, see BIX Performance.
  - Update your work tables to include the new columns.
  - Populate the new columns by running the resave utility.

  Apply DDL scripts manually to create the pr_extract_time and pr_sys_queues_bix tables if they are not present in the PegaRULES schema.

- Enabling the BIX Ruleset

Extract Rules overview

After enabling the built-in Business Intelligence Exchange (BIX) Ruleset on Pega Platform, you can run the BIX extraction processes from the *Extract* Rule to extract Class

©2025 Pegasystems Inc

data from your application. Create an *Extract* Rule to define the Class data to extract from your application using the Business Intelligence Exchange (BIX) Rule form.

> Extracting data from BIX

To extract data from within Pega Platform :

1. Enable the BIX Ruleset. For details, see Enabling the BIX Ruleset.
2. Create a new *Extract* Rule from the **New** tab. For details, see Creating and running an Extract rule.

> **NOTE:** Pega Platform does not support extracting to a Teradata database due to performance issues with the metadata queries that are common in BIX extractions.

3. Specify the Class properties to be extracted on the **Definition** tab.
4. Establish any filters for the data on the **Filter Criteria** tab. For details, see Extracting data in batch mode.
5. Specify the destination for the extraction process on the **File Specification** tab.
6. Run the *Extract* Rule. Running a BIX extraction through a *Job Scheduler* Rule is recommended for production environments. For details, see Using the Job Scheduler rule to extract data with BIX.
7. Review BIX logs. For details, see BIX logging.
8. Consider performance enhancements. For details see, Optimizing BIX data extraction performance.

> **NOTE:** When running an extract after upgrade, you might encounter the `ORA-00942` error that refers to the Oracle database. To fix the issue, grant the required privileges on the Oracle database. For more information, see Troubleshooting failed database connection after update or patch (Oracle database).

©2025 Pegasystems Inc

> Extractions for Pega Cloud® and on-premises applications

BIX extraction processes differ depending upon if you use Pega Cloud® services or an on-premises deployment for your application.

- For more detailed information about running a BIX extraction in Pega Cloud, including obtaining extract files from the Pega Cloud File Storage Repository, see Overview of BIX extractions in Pega Cloud environments.
- For more detailed information about running a BIX extraction in on-premises applications, see Overview of BIX extractions in on-premises systems.

> Extract process

Extractions are specified at the Class level, not at the application level. For example, if you have three case types in your application and would like to extract data for each, you must define an Extract rule for each of the three case types.

> **NOTE:** If you are extracting from a Class group, the extract files will include all classes in the Class group. If the Class is not a Class group, the extract files will only include instances of that Class.

You can specify the properties in each Class of your application that you want to extract. You can extract all properties, including Value Lists, Value Groups, and multiple levels of embedded Pages, Page Lists, and Page Groups. The BIX extraction for a Class can include any or all properties inherited by that Class from other classes. You can also filter the data to be extracted, extracting only the data that matches the conditions you set.

The data contained in these properties can be extracted to an XML file, to a Comma Separated Values (CSV) file, or to one or more tables in a relational database.

> **NOTE:** A single extraction process cannot concurrently extract from multiple databases or different database instances on the same machine. Each

©2025 Pegasystems Inc

> extraction process can have only a single data source. You can run different extraction processes, extracting from multiple Pega Platform databases, concurrently.

> Extraction frequency

BIX extractions do not extract data in real-time as the Pega Platform database is updated. Because BIX runs manually through the *Extract* Rule or automatically using a Pega Platform Job Scheduler Rule, it can be scheduled to run as frequently as desired.

> Schema

You can generate a database schema format for extractions based on your specifications in the *Extract* Rule definition. You can specify the table and column names and define the data type and precision. BIX provides an option to generate a Data Definition Language (DDL) file for the database schema it expects to write data to during an extraction. This file is used by a Database Analyst (DBA) to create the appropriate tables in the destination database. Any updates to the *Extract* definition that result in the addition of columns or tables need to be added manually by a DBA.

When generating to XML, the XML element names match the Pega Platform names and cannot be modified.

> Category

An Extract rule is an instance of the *Rule-Admin-Extract* Class. *Extract* Rules belong to the SysAdmin category.

BIX extractions include a time stamp and a batch identifier. This tagging can help you compare information from different extractions or isolate information from a specific extraction process.

- **Creating and running an Extract rule**
- **Extracting data by using secondary database**

©2025 Pegasystems Inc

- **More about Extract rules**

Creating and running an Extract rule

Obtain specific class data from your application by using Extract Rules. In this way, for example, you can conveniently integrate the data with other systems to create comprehensive reports, which monitor key metrics for your business.

Create and open an Extract rule to run a manual BIX extraction process. To limit the properties included in your extract, use filter criteria to extract specific properties from a Class. Filters can also exclude specific properties from an extraction and extract properties that have changed since the last extraction was run. Implementation of filter conditions can significantly impact BIX extraction performance.

For more information, see Optimizing BIX data extraction performance and Extracting data in batch mode.

If data that you try to extract is too large, BIX enables you to divide the scope into smaller parts to ensure smooth system performance. For more information, see Extracting large data by using BIX.

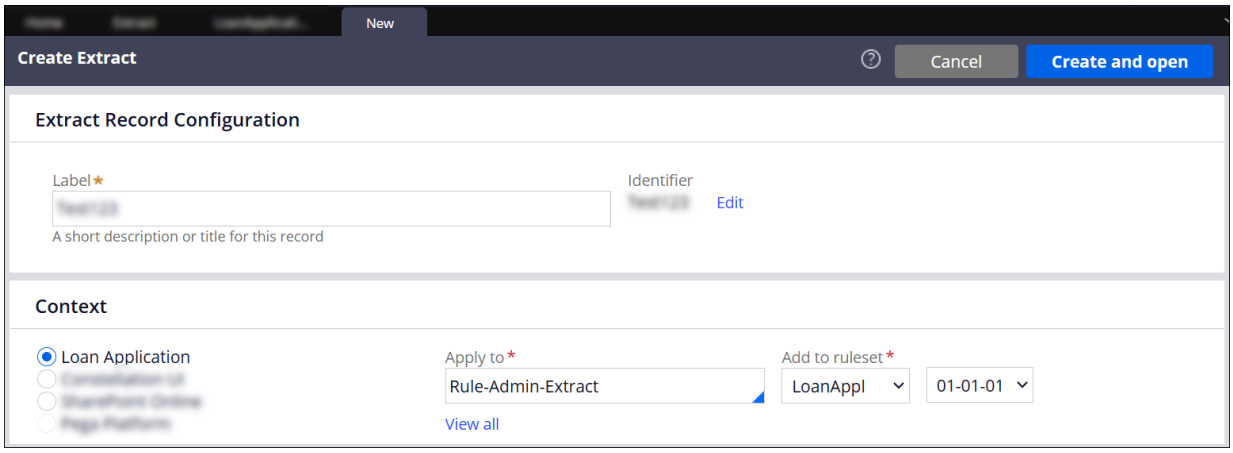
> **NOTE:** Manual BIX extractions are not designed to be run on a production environment. To run extractions as a background process on a production environment after creating an *Extract* rule, see Using the Job Scheduler rule to extract data with BIX.

Creating the Extract Rule

1. In the header of Dev Studio, click Create > **SysAdmin** > **Extract**.
2. In the **Extract Record Configuration** section, in the **Label** field, enter a name for the Extract rule.
3. In the **Context** section, select the application, Class, and Ruleset from which to extract data.

©2025 Pegasystems Inc

The following figure shows a sample customization of the Extract Rule:



Extract Rule

4. In the upper-right corner of the Rule, click **Create and open**.

Managing the Extract Rule

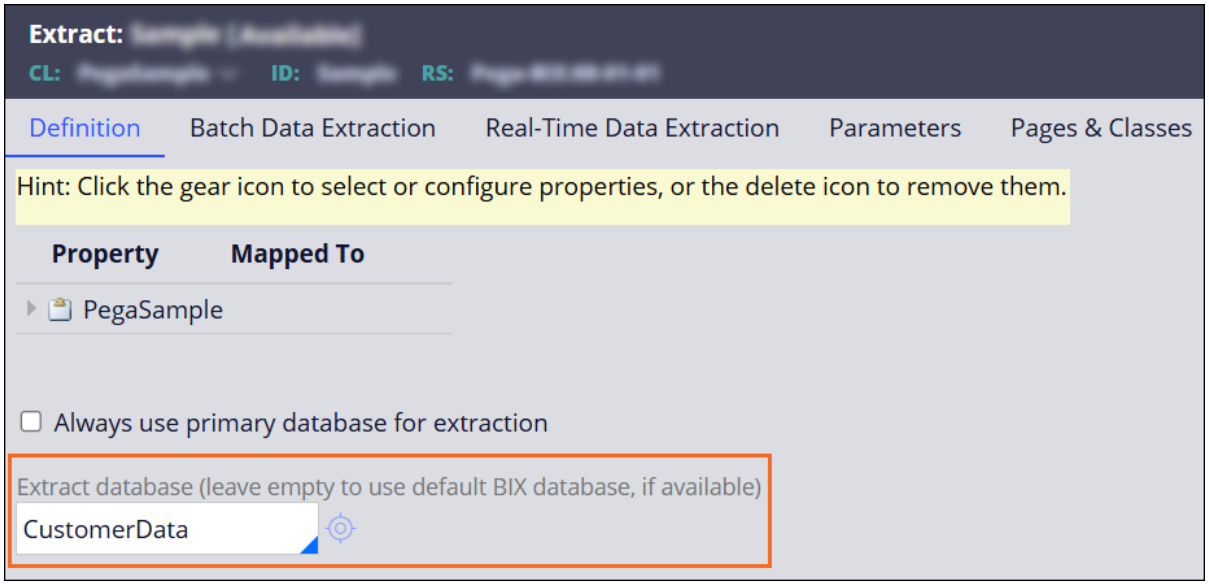
5. On the **Definition** tab, perform the following actions:
   a. In **Property** section, next to the specific Class, click the **Gear** icon.
   b. In the **Property Configurations** dialog box, in the **Select Properties** pane, select the check box of the page lists, page groups, and single value properties to extract, and then click **Submit**.
   c. **Optional:** To further specify nested pages or single value properties to extract, next to the specific page list or page group, click the **Gear** icon, and then specify the details.
   d. **Optional:** To extract all properties from a selected Class by using XML formats, select the **Get all properties** checkbox.

> **NOTE:** For non-BLOB tables, select properties manually. The **Get all properties** option fetches basic properties only.

6. **Optional:** To ensure that the extracts always use your primary database, select the **Always use primary database for extraction** checkbox.

©2025 Pegasystems Inc

7. **Optional:** To offload primary database and use a secondary database for BIX extracts, in the **Extract database** field, specify the secondary database to use. The following figure shows a sample Extract Rule, which has the `CustomerData` database specified to replace the primary database.



Extract Rule with specified secondary database

8. Click **Save**.
9. **Optional:** On the **Batch Data Extraction** tab, select filter conditions using properties of the primary Class.
   For more information, see Extracting data in batch mode.
10. In the **Parameters** tab, create parameters that populate the Clipboard at runtime. You use these parameters to customize BIX extractions by specifying them in the **Batch Data Extraction** tab.
    a. Enter the **Name** and **Description** for the parameter.
    b. In **Data Type**, enter the value type of the parameter
    c. In **Default** value, enter the parameters default value.
    d. Click **Save** to apply parameters.
    e. To specify these parameters in a Filter Criteria, enter these parameters in the **Batch Data Extraction** tab. For details, see Extracting data in batch mode.

©2025 Pegasystems Inc

11. **Optional:** In the **Pages & Classes** tab, create pages that populate the Clipboard at runtime. You use these Pages & Classes to customize data extractions by specifying them in the **Batch Data Extraction** tab.
    a. In **Page name**, identify a Clipboard page with the format *pyWorkPage.pyUserPage*.
    b. In the **Class** field, select the Class of the specified page.
    c. Click **Save** to apply the page and Class.
    d. To specify these pages and classes in a Filter Criteria, enter these pages and classes in the **Batch Data Extraction** tab. For details, see Extracting data in batch mode.
12. In the **Execution History** tab, view a list of concurrently running and previously run BIX extractions by clicking on each row.
13. To manually run a BIX extraction process, in the rule form toolbar, click **Actions** > **Run**.
14. In the **Label to use for this run** field, enter a unique identifier for this BIX extraction. Extract files contain the unique identifier to reference the *Extract* rule that ran the process. For details, see Unique run identifier.
15. For XML and CSV extracts, click **Perform Extract**.
16. For **Database Schema** extractions, perform the following tasks to generate a DDL for new extracts, changes to an existing extraction, or after a BIX upgrade:
    a. Click **Generate schema DDL** to generate a DDL file to create a formatted database table for the extraction process.
    b. If you chose to generate a manifest and store in a database schema, click **Generate manifest schema DDL** for the manifest table.

    > **NOTE:** Manifest files are generated by default on Pega Cloud.

    c. Click **Perform Extract**.
        - For Pega Cloud deployments, a URL appears for the DDL file.

©2025 Pegasystems Inc

- For on-premises deployments, a directory field appears for the system to download a DDL file to a location on the application server.
- Share the DDL file with your database administrator so that the administrator can format the external database table for extractions.

17. For **Database Schema** extractions, perform the following tasks to run the extraction after completing step 15:
    a. To run manual extractions to a database schema, repeat step 15 with the **Generating schema DDL** and the **Generate manifest schema DDL** check box cleared. BIX extractions through the *Job Scheduler* for database schema formats are now available to run after implementing schema DDL for a target database.
    b. In the **Number of records to update at a time** field, enter the number of records to insert to the target database in a batch.
    c. Click **Perform Extract**.
18. Repeat steps 13 to 16 to run additional manual extraction processes.

What to do next:

Retrieve your BIX extract files from their default save location:

- For Pega Cloud Services clients using XML or CSV formats, your BIX extract files are saved in the Pega Cloud Services File Storage repository. For more information, see Procedures for BIX extractions in Pega Cloud environments.
- For on-premises clients using XML or CSV formats, your BIX extract files are saved in the local directory that you specify.
- For database schema extractions, refer to your external database instance to view the extraction table data.

- **Extracting data in batch mode**
- Overview of BIX extractions in Pega Cloud environments

©2025 Pegasystems Inc

- Overview of BIX extractions in on-premises systems
- Creating and updating external relational database instances with JDBC URLs

Extracting data in batch mode

Decide what data you want to include in an Extract Rule by defining custom filter conditions. In this way, you retrieve only the most relevant information that you need at the moment, which saves time and streamlines data analysis.

For example, you can specify a filter to extract orders placed in the last month only to conveniently analyze recent sales trends.

> **IMPORTANT:** Pega Platform™ does not support extracts to Snowflake DB as the output destination for Extract Rules. Ensure that you define a different output for the extracted files.

1. Access an Extract Rule:
   - To access the existing Extract Rule, in the navigation pane of Dev Studio, click **Records** > **SysAdmin** > **Extract**, and then select the Rule that you want to edit.

     > **NOTE:**
     >
     > Only one Extract Rule in a given class can be enabled for real-time data extraction, hence the system does not permit using the `Save as` option on Extract Rules that already have the real-time data extraction enabled, and displays an error. If you want to use the `Save as` option, disable the real-time data extraction first.

   - To create a new Extract Rule, in the header of Dev Studio, click Create > **SysAdmin** > **Extract**.

©2025 Pegasystems Inc

For more information, see Creating and running an Extract rule.

Specifying criteria

2. On the **Batch Data Extraction** tab, in the **Criteria** section, select the required checkboxes:

Choices	Actions
Use last saved time (pxSaveDateTime) as start	Select the checkbox to extract all data created or updated since the last time the Extract rule ran, up to five minutes before the start of the extract. This assumes that engine startup from the command line takes five minutes. **NOTE:** This control is only applicable after the Extract rule has run at least once. Depending on the version of Pega Platform™, the filter uses different property: • Pega Platform version lower than 7.1.9 uses the **pxUpdateDateTime** property. • Pega Platform versions 7.1.9 to 8.8 uses the **pxCommitDateTime** property. • Pega Platform version 8.8.5 and higher uses the **pxSaveDateTime** property.

©2025 Pegasystems Inc

Choices	Actions
Skip standard filters	Select to exclude the primary key column (`pzInskey /pxObjClass`) from the filter during extraction. For more information, see Primary key usage in Pega Platform database tables.
Skip standard order by clause	Select to exclude default sorting based on the primary key column (`pzInskey / pxObjClass`) for the extraction.

Specifying filters

3. In the **Filter Criteria** section, click **Add filter**, and define a custom filter by performing the following actions:
   a. In the **Label** field, enter an alphanumeric label for the filter.

      > **NOTE:** Do not include spaces in the label.

   b. In the **Field** field, enter a property from the referenced Class that starts with a period.

      > **NOTE:** If you select **Best Bet** , the system displays the closest matching property for the entered string. In turn, selecting **All Matches** displays all matching properties that contain the string you entered.

   c. **Optional:** To use an SQL function, click the **Calculation builder** icon, and enter details.

©2025 Pegasystems Inc

d. In the **Operator** list, select a select a relational operator for the filter condition.

> **For example:**
>
> For the `.pyLabel` field, select **Is equal** operator to compare it with another value.

e. In the **Value** field, enter a value for the property that exists in the Class or that you created in the **Parameters** or **Pages & Classes** tabs to run against the field and operator.

f. To replace an empty **Value** field with a `null` value, select the **Use null if empty** checkbox.

> **Result:**
>
> Selecting the **Use null if empty** checkbox ensures that extract files do not contain empty values.

g. To ignore lowercase and uppercase in the **Field** and **Value** fields of the filter, select the **Ignore case** checkbox.

4. In the **Filter Criteria** section, in the **Filter conditions to apply** field, enter the logic that you want to use to filter data.

> **For example:**
>
> Enter `A AND (B OR C)` to display the data that matches condition A and either condition B or C.

> **NOTE:** The default field *pxUpdateDateTime* enables incremental extractions to limit extracted data to properties that contain an updated *pxUpdateDateTime* instance when compared to the last extraction.

©2025 Pegasystems Inc

Specifying output format

5. In the **Output** section, specify the directory details for the specific output file format you want to use for the extraction process:
   - To extract properties to an XML file according to top-level pages, page lists, and page groups with embedded XML tags, select **XML**.
   - To extract properties to comma-separated values in separate text files according to top-level pages, pages lists, page groups, and value properties, select **Comma Separated Value (CSV)**.
   - To extract properties directly into an existing relational database outside of the application, select **Database Schema**.

   > **IMPORTANT:** You must have insert and view-metadata privileges for the database into which you want to insert the Database Schema. For more information, contact your Database Administrator. Pega Platform does not support extracting to a Teradata database due to performance issues with the metadata queries that are common in BIX extractions.

   For more information about creating an external database instance, see Creating and updating external relational database instances with JDBC URLs.

The following tables feature all available customizations:

Output format	Fields
XML	• XML Schema Directory - Specify a local directory in which to place the XML schema file. • XML Schema File Name - Enter a name for the XML schema files.

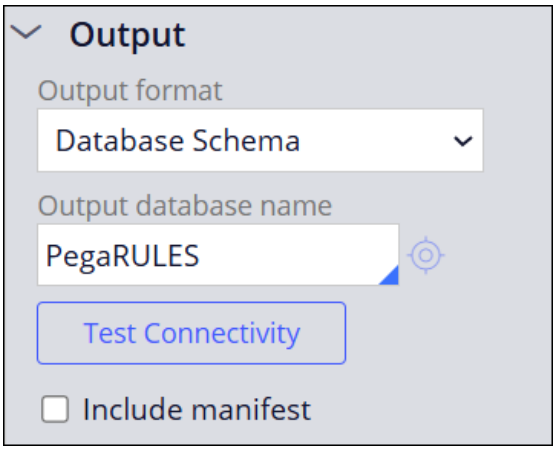
©2025 Pegasystems Inc

Output format	Fields
	• XML File Output Directory - Specify the target directory to place the XML output files. • XML Output File Name - Specify the target directory in which to place the XML output files. Pega Platform automatically adds .XML and.XSD extensions to the file name. Pega Platformsupports appending the following wild-card formatting strings to the output file name: • %i - (RunIdentifier) Adds a unique run-identifier to the extract file. ◦ Pega Platform generates a run-identifier consisting of the pzInsKey of the Extract rule followed by the sequence number which increments after each extraction process. • %s - Adds an automatically generated sequence number to each new extract. For example, `BIXExtract1`, `BIXExtract2` • %c - (ClassName) Adds the Class to which Pega Platform applied the Extract rule (see step 3). For example, `BIXExtract_PegaSample-Work`

©2025 Pegasystems Inc

Output format	Fields
	• %r - Adds the name of the Extract rule. For example, `BIXExtract_WeeklyExtract1` • %d - (ExtractRunDate) Adds the date of the extract run. • %t - (ExtractRunDateTime) Adds the date and time of the extract run. For example, specifying the output file name `BIXExtract_%s_%c_%r` generates in the output file name as `PegaExtract_1_PegaSample-Work_WeeklyExtract`. **NOTE:** Use %d and %t when running multiple extracts to get unique filenames and avoid Pega Platform overwriting previous extract files with the same name in the specified directory
CSV	Specify the target directory for the CSV extract files.
Database Schema	Specify a JDBC URL to an existing external database instance into which to insert the database schema. You must have the insert and view-metadata privileges for the database into

©2025 Pegasystems Inc

Output format	Fields
	which you want to insert the database schema. The following figure shows a sample configuration of the output format:  *BIX output format*

Output format	Fields
XML	XML & CSV extractions place the extract files into your Pega Cloud File Storage environment repository. To use a subfolder in your BIX folder to place your XML & CSV extract files, enter its name in the **Subfolder** field. **NOTE:** In the **Output Repository** field, you can specify your custom repository to which the system extracts data.
CSV	

©2025 Pegasystems Inc

Output format	Fields
	In this way, you do not use the regular Pega Cloud repository.
Database Schema	Specify a JDBC URL that links to an existing external target database instance into which to insert the database schema. You must have the insert and view-metadata privileges for the database into which you want to insert the database schema.

Output format	Fields
XML	• Manifest XML file output directory - Specify a local target directory to receive the XML manifest file. • Manifest XML output file name - Enter a name for the XML manifest files. Pega Platform automatically adds .xml and .xsd extensions to the file name. Pega Platform supports appending the following wild-card formatting strings to the output filename: • %i - (RunIdentifier) Adds a unique run-identifier to the extract file. ◦ Pega Platform generates a run-identifier consisting of the pzInsKey of the Extract rule

©2025 Pegasystems Inc

Output format	Fields
	followed by the sequence number which increments after each extraction process. • %s - Adds automatically generated sequence number of each new extract. For example, `BIXExtract1`, `BIXExtract2` • %c - (ClassName) Adds the Class to which Pega Platform applied the Extract rule (see step 3). For example, `BIXExtract_PegaSample-Work` • %r - Adds the name of the Extract rule. For example, `BIXExtract_WeeklyExtract1` • %d - (ExtractRunDate) Adds the date of the extract run. • %t - (ExtractRunDateTime) Adds the date and time of the extract run. For example, specifying the output file name `BIXExtract_%s_%c_%r` generates in the output file name as `PegaExtract_1_PegaSample-Work_WeeklyExtract`. **NOTE:** Use %d and %t when running multiple extracts to get unique filenames and avoid Pega Platform overwriting previous extract files with the

©2025 Pegasystems Inc

Output format	Fields
	same name in the specified directory.
CSV	Specify the target directory for the CSV manifest file.
Database Schema	Specify a JDBC URL to an existing external database instance into which to insert the database schema. You must have the insert and view-metadata privileges for the database into which you want to insert the database schema manifest files.

Output format	Fields
XML	XML and CSV extractions place the manifest files in the Pega Cloud File Storage repository.
CSV	
Database Schema	Specify a JDBC URL to an existing external database instance into which to insert the database schema manifest files. You must have insert and view-metadata privileges for the database into which you want to insert the database schema manifest files.

NOTE: Manifest files are generated by default on Pega Cloud.

©2025 Pegasystems Inc

6. **Optional: Applicable to Pega Cloud only:** To copy extract files to a custom output repository, in the **Output repository** field, specify the repository. For more information, see Extracting data to custom output repositories.

Testing connectivity

7. In the **Output** section, test the local application server filepath validity for the extractions by clicking **Test Connectivity**.
8. **Optional:** To use a property as a checksum SQL function for the extraction process, in the **Include check total property** field, enter a name of the property.

> **NOTE:** The option is not available on Pega Cloud and for high-throughput CSV extractions.

For more information, see Optimizing BIX data extraction performance.

9. In the upper-right corner of the Extract Rule, click **Save**.

Extracting data by using secondary database

Optimize your BIX extractions by directing them to a secondary database, thereby offloading the primary database. This approach prevents potentially heavy and long-running extracts from impacting your original database, ensuring optimal system performance.

> **NOTE:** The secondary database must provide the same schema as the original database.

©2025 Pegasystems Inc

Before you begin:

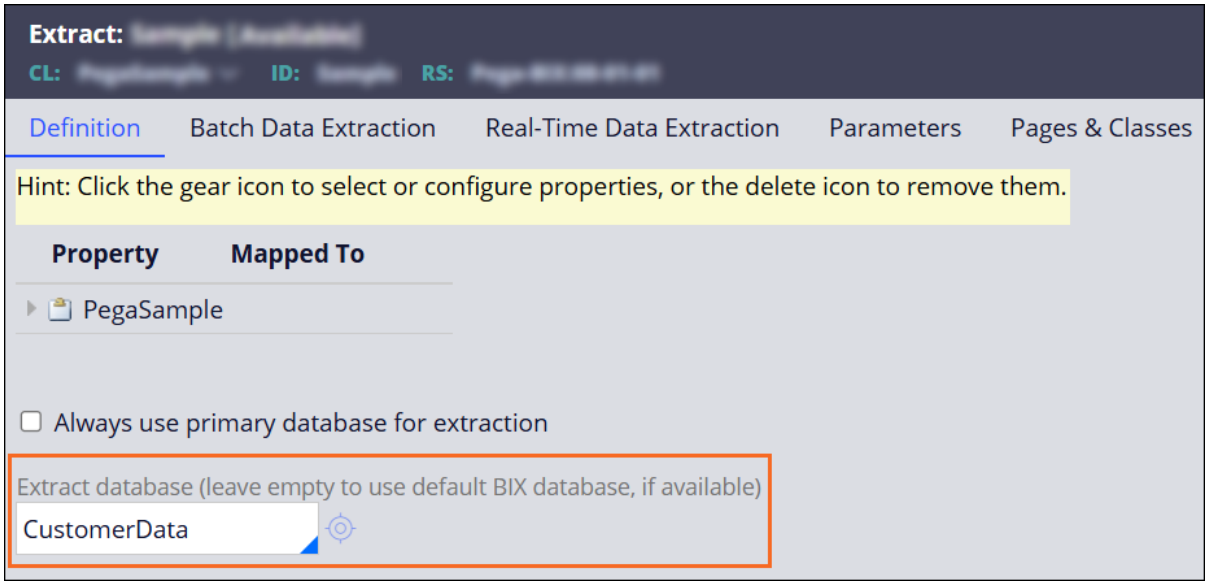
Ensure that your secondary database has the same schema as the original database.

1. In the navigation pane of Dev Studio, click **Records**.
2. Expand the **SysAdmin** category, and then click **Extract**.
3. In the list of Extract Rules, select the Rule that you want to edit.
4. On the **Definition** tab, in the **Extract database** field, specify the secondary database to use for the extraction.

   For example:

   Enter `CustomerData`.

   The following figure shows a sample Extract Rule, which has the `CustomerData` database specified to replace the primary database.



Extract Rule with specified secondary database

5. **Optional:** To test the extract, in the upper-right corner of the Extract Rule, click **Actions** > **Run**.
6. In the upper-right corner of the Extract Rule, click **Save**.

©2025 Pegasystems Inc

More about Extract rules

Testing

1. Click **Run**.
2. Enter a unique identifier and the maximum number of records to extract.
3. Click **Perform Extract**.
4. Review the resulting data and log files.

Scheduling Extract rule runs

For more efficient operation, you should schedule run of an Extract rule using a Pega Platform Job Scheduler (for Pega Cloud and on-premises) or from a Windows or UNIX/Linux command-line (for on-premises only). For details, see Using the Job Scheduler rule to extract data with BIX and Running the command-line extract process.

Exception handling

Executing BIX may result in errors or exceptions being reported. These exceptions may relate to syntax errors (in command-line execution), database constraint violations (when exporting to a database), or similar issues. For details, see BIX logging and error handling.

Rule operation

Extract rules can operate on the BLOB column of tables in the PegaRULES database or they can be used to extract data from tables with no BLOB column. However, when a Class table includes a BLOB (`pzPVStream` column), extraction always is performed from that BLOB value, and not from other database columns. For details, see Storage Stream.

If the Extract rule Class is a Class group, the query on the source database extracts all WorkObject instances of all classes within the Class group. If the extract rule Class is not a Class group, the query on the source database only gets instances of the particular Class.

PEGA ©2025 Pegasystems Inc

Forward and backward chaining during BIX extract (on-premises applications only)

For on-premises clients, in the `prconfig` file, a setting governs whether forward and backward chaining are enabled during a BIX extract. You should disable forward and backward chaining for better throughput when running BIX. For details, see Configuring optional prconfig.xml settings in on-premises systems.

- Extract Rules overview
- Creating and running an Extract rule
- Using the Job Scheduler rule to extract data with BIX

Configuring BIX real-time data extraction

Enable the real-time data extraction feature for quick transfer of data to another application. Real-time data extraction captures changes in the data of your Case Type or a Data Object and sends them in real-time to a subscribed customer's application, which can be either another Pega application, or a third-party application.

The real-time data extraction minimizes data latency and enhances the efficiency of data transfer between Pega Platform™ and customer’s applications. For example, in a banking application, any time a customer updates their information, if real-time data extraction is enabled, this data is immediately sent to a centralized customer data source. You can select the information which you want to send, such as the address or credit card details of the customer.

> **NOTE:** For initial data transfer, use batch data extraction, and real-time extraction to publish changes as they occur. The initial load ensures that the target system has complete and up-to-date information from Pega Platform. Subsequent changes are then extracted in real-time to keep the data current in a third-party application. Initial load of data is not applicable for new applications.

©2025 Pegasystems Inc

Before you begin:

Ensure that you have the Business Intelligence Exchange (BIX) Ruleset configured in the application stack. For more information, see Enabling the BIX Ruleset and Creating and running an Extract rule.

Pega Platform uses Extract Rules for real-time data extraction. For more information about Pega Business Intelligence Exchange data extraction, see BIX extraction overview.

1. Access an Extract Rule:
   - To access the existing Extract Rule, in the navigation pane of Dev Studio, click **Records** > **SysAdmin** > **Extract**, and then select the Rule that you want to edit.

     NOTE:

     Only one Extract Rule in a given class can be enabled for real-time data extraction, hence the system does not permit using the Save as option on Extract Rules that already have the real-time data extraction enabled, and displays an error. If you want to use the Save as option, disable the real-time data extraction first.

   - To create a new Extract Rule, in the header of Dev Studio, click Create > **SysAdmin** > **Extract**.

     For more information, see Creating and running an Extract rule.

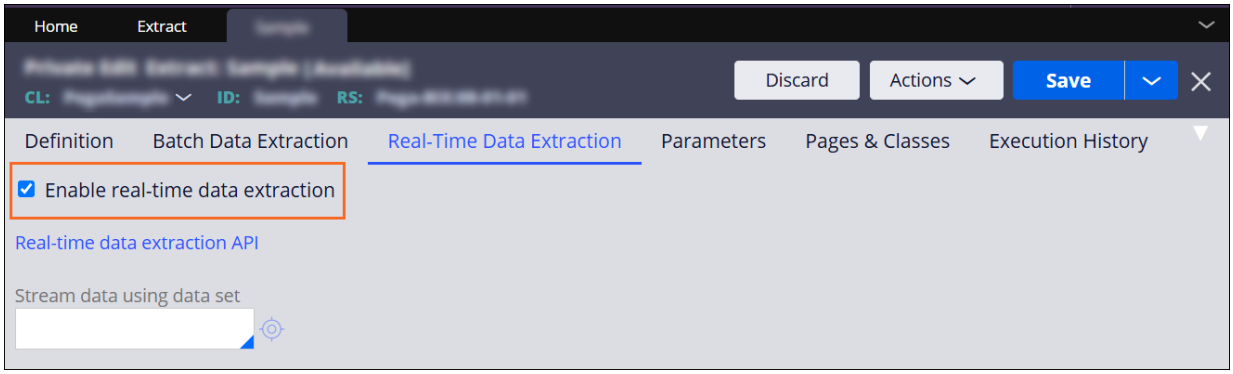
2. On the **Definition** tab of the Extract Rule, perform the following actions:
   a. Click the gear icon of the property that you want to configure.
   b. In the **Property Configurations** dialog box, select the properties to extract, and then click **Submit**.

©2025 Pegasystems Inc

3. On the **Real-Time Data Extraction** tab, select the **Enable real-time data extraction** checkbox.

> **NOTE:**
> - The system supports only one *Extract* Rule for each Case Type or Data Object.
> - The system supports only `Page`, `PageList` and scalar properties.
> - Delta of the events changes is not supported. The entire data set is extracted even if a single property changes.
> - Events are generated for each committed save operation to the database. For a single case, you might encounter multiple events being triggered and sent.

The following figure shows the **Enable real-time data extraction** checkbox in Dev Studio:



Enabling real-time data extraction

4. Depending on your preferences, you can obtain data in real-time by using one of the following methods:
   - To use APIs, leave the **Enable real-time data extraction** checkbox selected.

> **NOTE:**

©2025 Pegasystems Inc

Pega Platform provides the following out-of-the-box REST APIs to use with real-time data extraction:

Metadata API
: Provides the list of enabled data extraction objects and their IDs.

Registration API
: Registers third-party applications to connect to the system.

Un-register API
: Un-registers existing third-party applications.

Connect API
: Initiates polling with the specified metadata. When polling begins, any new updates become accessible to consumers.

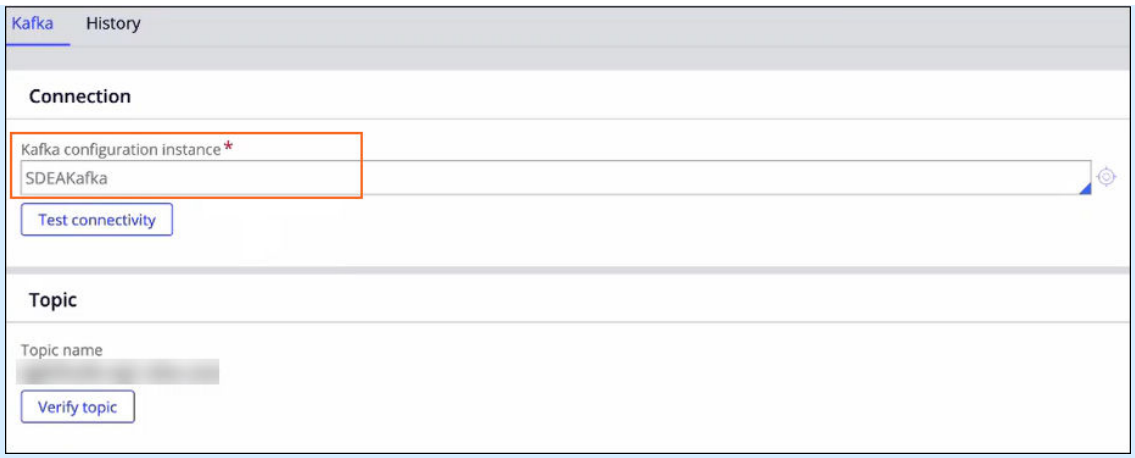
For more information, see Pega API services.

- To use a Stream Rule, in the **Stream data using data set** list, select the Rule that you want to use.

For example:

Stream Rule is a Kafka dataset Rule that a third-party application can select for real-time data extraction, and currently, only Kafka dataset rules are supported The following figure shows the *SDEAKafka* Rule selected as a Stream Rule.

©2025 Pegasystems Inc



Adding a Kafka Rule

5. Click Save.

Result:

With real-time data extraction enabled, whenever values of any property defined in the Case Type or Data Object configured in the Extract Rule change, the system passes the data to a customer's application.

Related concepts

- Data extraction
- Extract Rules overview
- BIX extraction overview
- Overview of BIX extractions in Pega Cloud environments
- Overview of BIX extractions in on-premises systems

Using the Job Scheduler rule to extract data with BIX

After you enable the built-in Business Intelligence Exchange (BIX) Ruleset on Pega Platform and create a Extract rule, you can schedule Business Intelligence Exchange (BIX) extractions using Pega Platform Job Scheduler rule.

©2025 Pegasystems Inc

Before you begin:

Gather the following information before extracting data with BIX using a *Job Scheduler* rule:

- The Extract rule **label**
- The Extract rule **Class group**

To run a BIX extraction process to a target database, the target database must be configured with the data description language (DDL) that is created during an initial BIX extraction process to import the database schema format. For details, see Creating and running an Extract rule.

You can run BIX extractions manually using the *Extract* rule, or automate the timing of BIX extractions using a Job Scheduler rule. Automated BIX extractions using a *Job Scheduler* rule run in the background processes of your system, which helps to optimize performance for your Pega application. The BIX extraction process cannot be run on more than one node at a time. For details, see Background processing best practices and Classifying nodes.

NOTE: For Pega Platform 8.4 and later, Job Scheduler rules functionally replace agents. For clients with agents running BIX extractions, it is best practice to replace them with a Job Scheduler rule. For more information, see Replacing an legacy Rule-type-agent with a job scheduler Rule.

1. In the header of Dev Studio, click Create > **SysAdmin** > **Job Scheduler**.
2. In the **Label** field, enter a name for the Job Scheduler rule.
3. In the **Context** field, select **BIX-enabled application**.
4. Click **Create and open** to display the Job Scheduler definition page.
5. In the **Runs on** drop-down list, click **Any one associated node**.

©2025 Pegasystems Inc

6. In **Schedule**, specify a period in which to run the scheduled extraction process. Each schedule form varies depending on the timeframe selected from the drop-down list.
7. In **Context**, select **Specify Access Group** and then enter the Access Groups that have access to Class *Rule-Admin-Extract*.
8. In **Class**, enter the Class for the extraction rule, *Rule-Admin-Extract*.
9. In **Activity**, enter the extraction activity, *pxExtractDataWithArgs*.
10. Click **Parameters** and enter the following BIX extraction specifications:
    a. In **pyExtractName**, enter the Extract rule **label**.
    b. In **pyExtractClassName**, enter the Extract rule **Class group**.
    c. In **pyArgs**, enter any runtime rule-specific parameters that you want to run on your BIX extractions. Enter the parameters in the `-n <value>` format if a value is required.

       For example, you enter `-c -f -x` to schedule extraction processes that contain all child classes from the selected Extract rule, include headers and footers in the output extract files, and stop the process upon the first encountered error. For a full list of Extract rule parameters, see BIX parameters and command-line arguments.

11. Click **Save** to schedule the extraction process.

    You can find a history of extraction results on the *Extract* rule **Execution History** tab.

- **Running an Extract rule by using an agent**

Related concepts

- BIX extraction overview

Related tasks

©2025 Pegasystems Inc

- Running an Extract rule by using an agent
- Replacing an legacy Rule-type-agent with a job scheduler Rule

Related reference

- BIX parameters and command-line arguments

Running an Extract rule by using an agent

Running an Extract rule using a Pega Platform agent.

To schedule an extract from within your application, modify the RunExtract agent that is in the Pega-BIX Ruleset to periodically invoke the *pxExtractDataWithArgs* activity to run a BIX Extract rule. This activity takes as arguments the Class and name of the Extract rule to run and, as a single string, the list of BIX command-line parameters desired for each run. The agent must only be enabled on the dedicated node for BIX processing to help avoid any negative performance impacts to production while the Extract rule is running.

> **NOTE:** For Pega Platform 8.4 and later, Job Scheduler rules functionally replace agents. For clients with agents running BIX extractions, it is best practice to replace them with a Job Scheduler rule. For more information, see Replacing an legacy Rule-type-agent with a job scheduler Rule.

> **NOTE:** For on-premises clients, the `agent/threadpoolsize` setting in the `prconfig.xml` file controls the number of threads available in the agent pool, and consequently, the number of agents that can be run concurrently. If you schedule more agents to run concurrently than the value of the `agent/threadpoolsize` setting, you might get a stale thread error. Increase the value of the `agent/threadpoolsize` setting. The default value of the `agent/threadpoolsize` setting is 5.

PEGA ©2025 Pegasystems Inc

1. On the **Schedule tab** of the Agent rule, enter `Rule-Admin-Extract` in the **Agent Name** field.
2. In the **Mode** field for the agent, select **Advanced** unless you are creating queue items that a standard mode agent requires.

> **NOTE:** Agents in standard mode that do not have queue items created for them will appear to have run, but the activity is not executed.

3. Enter `pxExtractDataWithArgs` in the **Activity name** field.
4. Click **Parameters** to enter command line parameters in the following format when you are using one or more parameters.

> **NOTE:** If you are using an agent that calls pxExtractDataWithArgs, place quotes around the values. The extract fails if you do not use quotes. For example:

```
z "TEST-TESTAPP-WORK G-12" -Z "TEST-TESTAPP-WORK G-12"
```

If you are using an agent that calls a wrapper activity that calls pxExtractDataWithArgs, place quotes around the values and around the entire parameter string. Use the backslash to escape the quotes around the values. For example:

```
"z \"TEST-TESTAPP-WORK G-12" -Z \"TEST-TESTAPP-WORK G-12\""
```

The -i, -a, -p, -T, and -P command-line parameters are not supported for pxExtractDataWithArgs. The -X, -b, and -f command-line parameters cannot be used if pxExtractDataWithArgs is used on a Pega Cloud instance.

5. On the **Security tab**, configure the Access Group to be the application Access Group.

©2025 Pegasystems Inc

6. Select **Bypass activity authentication**.
7. Create an Agent schedule and map the agent to the BIX node type.

> **NOTE:** This functionality is available when you purchase and install the BIX application.

> **What to do next:**
>
> See Optional command-line BIX parameters and Agents rules.
>
> See Developer Help topics Optional command-line BIX parameters and Agents rules for more information.

- Extract Rules overview
- Running a BIX command-line extraction in on-premises systems
- Associating agents with node types
- Agent schedule data instances

Unique run identifier

When using Business Intelligence Exchange (BIX), you can provide a unique run identifier for each BIX extraction process to mark additions to both parent and child tables in the database, and parent and child CSV files. This enables you to accomplish the following actions after BIX extraction process:

- Easily identify all inserted records for a particular BIX extract run, both in the main table for a Class and in subsidiary tables created for page lists, page groups, and value lists in the Class.
- Delete records from a failed BIX run so that an extract can be re-run without producing incorrect or redundant data.

©2025 Pegasystems Inc

Two fields have been added to the main table for each extracted Class in the database, **pxExtractDateTime.TIMESTAMP** and **pxRunIdentifier VARCHAR(255)**.

The **pxExtractDateTime** field has also been added to all subsidiary tables for the Class in the database.

The value of the **pxExtractDateTime** field matches the value of *pyLastUpdateTime* in the table **pr_extract_time** for the run. This allows you to perform a join action and obtain all of the records inserted for a BIX run.

You can also include a run identifier through the –k parameter option. For details, see BIX parameters and command-line arguments.

BIX can still run in backward compatibility mode for tables without new fields. If the new fields exist in the database, they are filled in by BIX; otherwise, BIX runs in backward compatibility mode.

- BIX logging
- BIX error handling

Overview of BIX extractions in Pega Cloud environments

After enabling the built-in Business Intelligence Exchange (BIX) Ruleset on Pega Platform, you can extract data from your Pega Platform application with an Extract or Job Scheduler rule in your Pega Cloud environment.

> **NOTE:** The extracted files use the following naming convention: `BIX_`*Class name*`_`*Extract rule name*`_`*Creation date*`_`*Sequence ID*.
> The elements comprising the file name:
>
> *Class name*
> Class name where the extraction rule is defined.

©2025 Pegasystems Inc

Extract rule name
Name of the extract rule.

Creation date
File creation date in `yy/mm/dd` format. The date is based on the timezone where the application server resides.

Sequence ID
Incremented based on the number of runs for that extract rule in a single day, regardless of extract's success or failure.

After enabling BIX, you can configure BIX data extractions using Pega Platform application Class data for manual or automatic extract processes. Performing BIX extractions in Pega Cloud Service environments contain key differences compared to on-premise deployments:

- The BIX extraction configuration for XML or CSV formats and their manifest files places the extract files into your Pega Cloud File Storage repository. Clients do not have direct access to the Pega Cloud File Storage repository. To obtain your BIX extract files, you must move the files to a location in your enterprise network using one of the following SFTP-based methods:
  - Configure an SFTP server in your enterprise to integrate the Pega Cloud SFTP service to pull your BIX extractions off the Pega Cloud File Storage repository.
  - Configure a BIX-specific FTP listener in Pega Platform to push your BIX extractions to an SFTP server in your enterprise network.
- CSV extract files can be password protected.
- Pega Cloud environments do not support running the BIX extraction process from the command-line.
- BIX extractions to a target database require generating a DDL file referenced through a URL in your Pega Platform, which you must then share with your database administrator (DBA) in order for the DBA to format the required database tables in the target database.

PEGA ©2025 Pegasystems Inc

> **NOTE:** You can enable the *BIX/enableToFileOptionOnCloud* dynamic system setting to control how BIX handles the extraction of complex data structures, especially PageList and ValueList properties, when generating output files in a Cloud environment. Once enabled, BIX generates a separate output file for each PageList or ValueList property defined in an Extract Rule, and uses the following naming convention: *Class name_Extract Rule name_Sequence ID_ToFileName*, where *ToFileName* is specified in the Extract Rule and applies only to PageList and ValueList-specific files.

SFTP-based methods to obtain extract files in Pega Cloud

If you have upgraded from Pega Platform 7.3 or earlier, or you have previously used an FTP listener to receive BIX extract files, it is recommended that you pull the extract files from the Pega Cloud File Storage repository using the Pega SFTP service. For details, see *Post-upgrade tasks following Pega Platform upgrades (7.x to 8.x)* in Pega Cloud environments.

You enable either SFTP option for obtaining extract files by using dynamic system settings (DSS). For details about creating DSS used to enable each BIX extraction location option, see Creating a dynamic system setting. Refer to the following table to enable the Pega Cloud SFTP service method or the FTP listener method, and then configure each setting by following the instructions in their respective articles:

DSS	DSS fields to pull files from Pega Cloud File Storage using your SFTP server (default)	DSS fields to push files to your SFTP server using an FTP listener
`BIX/useCustomersSFTP`	`false`	`false` **NOTE:** When set to `false` in Pega Cloud 3, the

©2025 Pegasystems Inc

DSS	DSS fields to pull files from Pega Cloud File Storage using your SFTP server (default)	DSS fields to push files to your SFTP server using an FTP listener
		system uses the default repository location for file export, such as `file://pegacloudrepository/bix`.
`BIX/sharedPath`	N/A	`file://pegacloudrepository:/bix`
`BIX/ignoreValidation`	N/A	`true`
	For more information about configuring a BIX SFTP Server, see Configuring the Pega Cloud SFTP service to access BIX extract files.	For more information about configuring a BIX FTP listener, see Configuring an FTP listener to access BIX data extracts in Pega Cloud environments.

If you do not subscribe to the Pega Cloud SFTP service, or cannot connect to the BIX FTP listener, consider using an activity to obtain your extract files. For more information, see Configuring an activity to access BIX extract files.

For more information about running a BIX extraction in Pega Cloud applications, see Procedures for BIX extractions in Pega Cloud environments.

- **Procedures for BIX extractions in Pega Cloud environments**

©2025 Pegasystems Inc

- **Configuring the Pega Cloud SFTP service to access BIX extract files**
- **Configuring an FTP listener to access BIX data extracts in Pega Cloud environments**
- **Configuring an activity to access BIX extract files**
- **Troubleshooting BIX in Pega Cloud environments**
- **Extracting data to custom output repositories**

Procedures for BIX extractions in Pega Cloud environments

After enabling the built-in Business Intelligence Exchange (BIX) Ruleset on your Pega Platform application, BIX extractions are available to run through an *Extract* or *Job Scheduler* rule. XML & CSV extractions and their manifests must be obtained via an SFTP Server or an FTP listener.

To create an *Extract* rule, run an initial extract, and retrieve extract files in your Pega Cloud environment, refer to the following articles:

1. Enable the BIX Ruleset on your application. For more information, see Enabling the BIX Ruleset.
2. Create and run an initial *Extract* rule through one or more of the following methods:
   a. Create and run a BIX *Extract* rule. For more information, see Creating and running an Extract rule.
   b. Run an extraction process through the *Job Scheduler* rule. For more information Using the Job Scheduler rule to extract data with BIX.
3. Retrieve the BIX extract files for XML & CSV formats through one of the following methods:
   a. Pull BIX extract files from the Pega Cloud File Storage repository with the SFTP service. For more information, see Configuring the Pega Cloud SFTP service to access BIX extract files.

PEGA ©2025 Pegasystems Inc

b. Push the BIX extract files to a shared path using an FTP listener. For more information, see Configuring an FTP listener to access BIX data extracts in Pega Cloud environments.
c. Push the BIX extract files to a local repository through an activity. For more information, see Configuring an activity to access BIX extract files.

- Overview of BIX extractions in Pega Cloud environments
- Configuring an FTP listener to access BIX data extracts in Pega Cloud environments
- Configuring the Pega Cloud SFTP service to access BIX extract files
- Creating and running an Extract rule

Configuring the Pega Cloud SFTP service to access BIX extract files

After a Business Intelligence Exchange (BIX) extraction process in a Pega Cloud environment, obtain the BIX extract files from your Pega Cloud File Storage repository. You can use the Pega Cloud SFTP service to pull the extract files onto your enterprise SFTP server.

The BIX extraction process for XML and CSV formats, as well as their manifest files, places the BIX extract files into your Pega Cloud File Storage repository. Clients do not have direct access to the Pega Cloud File Storage repository and must pull BIX extract files to a location in your enterprise network using the Pega Cloud SFTP service. Pega Cloud environments also support the configuration of a BIX-specific FTP listener in Pega Platform, which can push your extract files to an SFPT server in your enterprise. For details, see Configuring an FTP listener to access BIX data extracts in Pega Cloud environments.

Set up the Pega SFTP Service and obtain your BIX extraction files by performing the following actions:

1. Log in to your **My Support Portal** account.
2. In the Support Request tab, click **Create** to file a Cloud service request.

©2025 Pegasystems Inc

3. Send the public key and the list of approved IP addresses in the body of the service request form.
   The Pega Global Client Support team receives your service request and then deploys the SFTP service.
4. Global Client Support sends you a file containing the SFTP hostname, SFTP username, and folder URL that you use to access the SFTP service.

   Global Client Support sends the hostname and URL in the following format:

   ```
   <SFTP hostname>:<client name/designation>.<project>.<type of environment[dev-test, staging, production]>.sftp.pegacloud.net
   ```

5. Configure your SFTP client or SSH shell using the hostname, SFTP username, and folder URL to interact with the Pega Cloud Services SFTP service.
6. Pull the extract files from the repository into a location in your enterprise SFTP server.
7. Download the extract files.

- Overview of BIX extractions in Pega Cloud environments
- Procedures for BIX extractions in Pega Cloud environments
- Configuring an FTP listener to access BIX data extracts in Pega Cloud environments

Configuring an FTP listener to access BIX data extracts in Pega Cloud environments

After a Business Intelligence Exchange (BIX) extraction process in a Pega Cloud environment, obtain the BIX extract files from your Pega Cloud File Storage repository. You can use an FTP listener rule to push the extract files to your enterprise SFTP server.

PEGA ©2025 Pegasystems Inc

> **Before you begin:**
>
> Set up an SFTP server in your domain with an SFTP or SSH client (if one does not already exist).

The BIX extraction process for XML and CSV formats, as well as their manifest files, places the BIX extract files into your Pega Cloud File Storage repository. Clients do not have direct access to the Pega Cloud File Storage repository and must push the extract files to an SFTP server in your enterprise network using a BIX-specific FTP listener in Pega Platform. Pega Cloud environments also support the use of the Pega Cloud SFTP service to pull the extract files onto an SFTP server in your enterprise. For details, see Configuring the Pega Cloud SFTP service to access BIX extract files. For more information about enabling the FTP listener, see About FTP Server data instances.

1. In the navigation pane of Dev Studio, click **Create** > **SysAdmin** > **Dynamic System Settings**.
2. Define a relative path on the destination SFTP site where the files will be sent with the following actions:
   a. In the **Name** field, enter a short description for the rule.
   b. In **Owning Ruleset**, enter the owning Ruleset for BIX.
   c. In **Setting Purpose**, enter `BIXFTPRelativeRemotePath`.
   d. Click **Create and open**.
   e. In **Value**, enter the target path to which to push the BIX extraction files.
3. To provide an access location and credentials to your SFTP server, click **Integration-Resources** > **FTP Server**.
   a. Select the **BIXFTPServer**.
   b. On the **Environment** tab, enter your SFTP server **Host name** and **port number**.
   c. In **Authentication**, click the **properties** icon to define a **user name** and **password** with which to access the SFTP server.
   d. Click **Save**.

©2025 Pegasystems Inc

   e. Return to the *BIXFTPServer* rule and click **Save**.
   f. In the *BIXFTPServer* rule *BIXFTPServer*, click **Test Connectivity** to verify the configuration and save this instance.
   g. Click **Test Connectivity** in *BIXFTPServer* to confirm connectivity.
4. To configure the Service Package data instance with the operator Access Group that has privileges to run a BIX extraction process, click **Integration-Resources** > **Service Package**.
   a. Select **BIXFTPServicePackage**.
   b. In the **Service Access Group**, enter the Access Group for the operator with privileges to run the BIX extraction process.
   c. Click **Save**.
5. To enable the BIXFTPListener, click **Integration-Resources** > **File Listener**.
   a. Select **BIXFTPListener**.
   b. Clear the **Block Startup** check box.
   c. Click **Save**.
6. From Admin Studio, verify that the BIXFTPListener is running.
7. You can test the BIX extraction process, and verify the success of the process after manually running an *Extract* rule, by confirming the arrival of the BIX extraction file on your SFTP server. When running a manual test, ensure that you only extract a small number of Class instances. For more information, see Creating and running an Extract rule.

- Overview of BIX extractions in Pega Cloud environments
- Procedures for BIX extractions in Pega Cloud environments
- Managing listeners

Configuring an activity to access BIX extract files

After a Business Intelligence Exchange (BIX) extraction process completes in a Pega Cloud environment, obtain the BIX extract files from your Pega Cloud File Storage repository. You can use an activity to pull the extract files onto your enterprise local

©2025 Pegasystems Inc

storage. The activity can be helpful when the BIX FTP listener encounters an error, a transfer fails, or if you do not subscribe to the Pega SFTP service.

Before you begin:

Prepare your BIX extract files and find the directory in which they reside to determine the activity code.

1. Place files in your Pega Cloud File Storage repository by running a BIX extraction process.

   For more information about the BIX extraction process, see Creating and running a BIX extract rule.

2. Create a library with which to integrate a function.
3. Find the extract file and directory of the extract file in the Pega Cloud File Storage repository by using a repository API.

   Use *D_pxListFiles* to find the file path and file name of your BIX extract files.

   For more information about using repository APIs, see Repository APIs.

The BIX extraction process for the XML and CSV formats, as well as their manifest files, places the BIX extract files into your Pega Cloud File Storage repository. You do not have direct access to the Pega Cloud File Storage repository but you can pull BIX extract files to a location in your enterprise network by using the Pega Cloud API through an activity.

1. In the header of Dev Studio, click **Create** > **Technical** > **Function**.
2. In the **Label** field, enter a name for the rule.
3. In the **Library** field, select a library that contains an empty Java directory.
4. In the **Parameters** section, enter the following parameter names and Java types:

Name	Java type
tools	PublicAPI

©2025 Pegasystems Inc

Name	Java type
repositoryName	String
filePath	String
fileName	String

5. Click Create and open.
6. On the Java tab, in the **Compilation status** section, select the Function ready to be compiled? check box.
7. In the Java source field, enter the following code:

```
String fullFileName = "file://" + repositoryName + ":" + filePath+ "/" + fileName;
try {

        HashStringMap aMap = new HashStringMap();
        aMap.put("ContentDisposition", "attachment; filename=\"" + fileName + "\"
");
        aMap.put("contentType", "application/octet-stream");

        PRFile fileToDownload = new PRFile(fullFileName);
        if(fileToDownload.exists()) {
        tools.sendFile(fileToDownload, false, aMap, true);
        oLog.error("pyDownloadZipFile: File exists and sent " + fullFileName);

        } else {
        oLog.error("pyDownloadZipFile: Could not download file. File " + fullFileNa
me + " does not exist.");
        tools.getPrimaryPage().addMessage("DownloadFailedFileDoesntExist");

}
} catch (Exception e) {
oLog.error("pyDownloadZipFile: Failed to download file" + e.getMessage());
```

©2025 Pegasystems Inc

```
}
```

8. Click Save.
9. In the header of Dev Studio, click Create > Technical > Activity.
10. In the activity rule, enter a label, select an application and the associated Ruleset, and then click Create and open.
11. On the **Steps** tab, for the first step, enter the following information:
    a. In the Method field, enter `Property-Set`.
    b. In the Description field, enter `Set the file parameters`.
    c. Click the Expand icon.
    d. In the **Method Parameters** section, in the PropertiesName field, enter `Param.fullFileName`.
    e. In the PropertiesValue field, enter `"file://"Param.repositoryName+":"+"/"+Param.fileName`.
    f. Click the add row button.
    g. In the PropertiesName field, enter `Param.Checkfile`.
    h. In the PropertiesValue field, enter `Param.filePath+"/"+Param.filename`.
    i. Click Jump.
    j. In the resulting window, select the Enable conditions after this action check box.
    k. In the When field, enter `@hasMessages(D_pxGetFile[repositoryName:Param.repositoryName,filePat responseType:"STRING"])`.
    l. In the If true list, select Jump to Later Step.
    m. In the True param field, enter `DPAGE`.
    n. In the If false field, enter Continue Whens.
    o. Click Submit.
12. For the second step, enter the following information:
    a. In the Method field, enter `Java`.
    b. In the Description field, enter `Java code to download the ZIP file`.

©2025 Pegasystems Inc

c. Click the **Expand** icon.
d. In the **PropertiesName** field, enter `Param.fullFileName`.
e. In the **PropertiesValue** field, enter `"file://"Param.repositoryName+":"+"/"+Param.fileName`.
f. Click the **add row** button.
g. In the **PropertiesName** field, enter `Param.GetfilePath`.
h. In the **PropertiesValue** field, enter `Param.filePath+"/"+Param.filename`.
i. Click **Jump**.
j. In the resulting window, select the **Enable conditions after this action** check box.
k. In the **When** field, enter `hasMessages`.
l. In the **If true** list, select **Jump to Later Step**.
m. In the **If false** list, select **Exit Activity**.
n. Click **Submit**.

13. For the third step, enter the following information:
    a. In the **Method** field, enter `Property-Set`.
    b. In the **Description** field, enter `Using the function instead of Java step.`
    c. Click the **Expand** icon.
    d. In the **PropertiesName** field, enter `Param.Test`.
    e. In the **PropertiesValue** field, enter the name of the function that you created.
14. For the fourth step, enter the following information:
    a. In the **Method** field, enter `Show-Page`.
    b. In the **Step Page** field, enter `Attachment Page`.
    c. In the **Description** field, enter `Show if the Java page has an error.`
    d. Click **When**.
    e. In the resulting window, select the **Enable conditions before this action** check box.
    f. In the **When** field, enter **Always**.
    g. In the **If true** list, select **Continue Whens**.

©2025 Pegasystems Inc

h. In the **If false** list, select **Skip Step**.
i. Click **Submit**.
j. Click **Jump**.
k. In the resulting window, select the **Enable conditions after this action** check box.
l. In the **When** field, enter **Always**.
m. In the **If true** field, enter **Exit Activity**.
n. In the **If false** field, enter **Continue Whens**.
o. Click **Submit**.

15. For the fifth step, enter the followings information:
    a. In the **Method** field, enter `Show-Page`.
    b. In the **Step Page** section, enter `D_pxGetFile[repositoryName:Param.repositoryName,filePath:Param.Check responseType:"STRING"`.
    c. In the **Description** field, enter `check if file/filepath/repositoryname does not exist`.
16. Click **Save**.
17. On the **Actions** list, select **Run**.
18. In the **fileName** parameter field, enter the name of the extract file.
    The file name should reflect the extract file placed in the repository.
19. In the **filePath** parameter field, enter `bix/work_BIXFTPListener/completed`.
20. In the **repositoryName** field, enter `pegacloudrepository`.
21. Click **Run**.

> **Result:**
>
> Your browser opens a new tab and displays a download prompt for your BIX extract file.

- Overview of BIX extractions in Pega Cloud environments
- Procedures for BIX extractions in Pega Cloud environments

©2025 Pegasystems Inc

- Configuring an FTP listener to access BIX data extracts in Pega Cloud environments

Troubleshooting BIX in Pega Cloud environments

When you run a Business Intelligence Exchange (BIX) extraction process in Pega Cloud from a Pega Platform Extract rule or from a Pega Platform Job Scheduler rule, learn to recognize the most common errors in the extraction process in Pega Cloud environments. Refer to the following tables to troubleshoot the root causes and resolve or prevent problems with extractions.

Common issues

When you want to diagnose an issue with a BIX extraction process in a Pega Cloud environment, first verify that you have specified the BIX configuration settings correctly. For more information, see Specifying configuration settings for BIX in a Pega Cloud environment.

Do not use agents to automate the BIX extraction process. For Pega Platform 8.4 and later, Job Scheduler rules replace agents for automating BIX extraction processes. Clients with agents running BIX extractions should replace the agents with a Job Scheduler rule. For more information, see Replacing an legacy Rule-type-agent with a job scheduler Rule and Using the Job Scheduler rule to extract data with BIX.

You can find a record of errors in the BIX extraction processes in the BIX log files. For more information about BIX logging, see BIX logging. For information on common BIX errors, refer to the following sections in this article:

- An extraction process in Pega Cloud environments using filter conditions or parameters through the Job Scheduler causes several errors
- Transfer fails when File Listener invokes FTP data instance for new compressed files that are transported to target FTP location
- Extraction process fails with query timeout exception
- Generated CSV extract or manifest file cannot be parsed

PEGA ©2025 Pegasystems Inc

- Java SQLException for schema generation
- System time zone ignored

An extraction process using filter conditions or parameters with the Job Scheduler causes several errors

The following solutions apply when you run the extraction process using filter conditions or parameters through the Job Scheduler to retrieve Class instances. The data extraction writes the requested data from the BLOBs of the Class instances to a set of files in a temporary folder, compresses the files, and then copies the ZIP file to the folder location specified in the `Pega-RULES!BIX/sharedpath` dynamic system setting (DSS).

Symptom	Explanation	Solution
An extraction process fails and produces the following error in the log files: `:onPegaCloud is needed to run the extract on cloud instance`	The `Pega-RULES!onPegaCloud` DSS is not set to true.	Set the DSS setting for onPegaCloud to true. • DSS setting: `onPegaCloud` • Ruleset: *Pega-RULES* For more information about editing a DSS value, see Editing a dynamic system setting.
	The run-time environment is not correctly identified as Pega Cloud.	Set `BIX/sharedpath` to a folder location with read-write access for Pega Platform.
An extraction process does not fail, but it does not extract the data based on the configured Extract rule.	The filter conditions in the Extract rule are incorrect, and the incorrect set of instances is selected.	Correct the Filter conditions of the Extract rule.

©2025 Pegasystems Inc

Symptom	Explanation	Solution
		For more information about filter criteria, see Extracting data in batch mode.
An extraction process fails. For CSV extraction, the error logged is: `Error while adding csv data` For XML extraction, the error logged is: `Could not write object" for xml extraction` For Manifest files, the error logged is: `Could not write manifest summary` For an IOException, the error logged is: `No space left on device`	There is not enough space in the temporary folder used when generating files.	Increase the space allotted to the folder that is used to store temporary files.
An extraction process fails and produces the following error in the log files: `Error creating zip file`	There is not enough space in the folder used to save the compressed file.	Increase the space allotted to the folder that is used to store the compressed file.

©2025 Pegasystems Inc

Symptom	Explanation	Solution
For an IOException, the error logged is: `No space left on device`		
An extraction process fails and produces the following error in the log files: `Error creating zip file` (with the relevant stack trace).	A problem occurred during compression.	File a service request with Pega Cloud services that contains the logged stack trace.
An extraction process fails and produces the following error in the log files: `Unable to process <Extract classname>` (with the relevant stack trace).	BIX generates an error during its run for some other reason, for example, after running out of memory.	File a service request with Pega Cloud services that contains the logged stack trace.

Transfer fails when File Listener invokes FTP data instance for new compressed files that are transported to target FTP location

The following solutions apply when you use a File Listener to access your extract files. File Listeners determine when a new compressed file has been written to the folder location and invoke the FTP data instance in order to transfer the file. Transfer of the file to the target FTP site can then fail for several reasons.

Symptom	Explanation	Solution
The file transfer fails and the file remains in the	A defect in the file listener causes the compressed file	Verify the correct configuration of the FTP

©2025 Pegasystems Inc

Symptom	Explanation	Solution
shared path location: `<sharedpath>/work_BIXFTP listener`. The error logged is: `FTP connection failed`.	to remain in the folder without Pega Platform invoking the FTP data instance.	listener rule. For more information, see Configuring an FTP listener to access BIX data extracts in Pega Cloud environments.
The file transfer fails and the file remains in the location of `<sharedpath>/work_BIXFTPListener/completed`. The file uses an `.err` extension. The error logged is: `FTP connection failed. Logon failure: unknown username or bad password`	The **Environment** tab of the FTP Server data instance BIXFTPServer is not correctly configured to access the customer's FTP site, and the transfer does not take place.	Verify the correct configuration of the **Environment** tab of the FTP Server data instance. For more information, see step 3.b of Configuring an FTP listener to access BIX data extracts in Pega Cloud environments.
The file transfer fails and the file remains in the location of:`<sharedpath>`. The error logged is: `InvalidConfigurationException: No such Rule-Application instance <Access`	The Authentication Profile data instance BIXFTPAuthProfile is not correctly configured with the login credentials of the FTP Server.	Verify the correct configuration of the **Authentication** profile. For more information, see step 3.c of Configuring an FTP listener to access BIX data extracts in Pega Cloud environments.

©2025 Pegasystems Inc

Symptom	Explanation	Solution
`group's application name>`		
The File Listener cannot poll for the ZIP file, and the file remains in the location of `<sharedpath>`.	The Service Package data instance `BIXFTPServicePackage` is not correctly configured with the operator Access Group.	Verify the correct configuration of the **Service Access Group**. For more information, see step 4.b of Configuring an FTP listener to access BIX data extracts in Pega Cloud environments.
The file transfer fails and the file remains in the location of `<sharedpath>/work_BIXFTPListener/completed` with `.err` extension. The error logged is: FTP `connection failed`.	• The File Listener BIXFTPListener has not been enabled by clearing the **Block Startup** check box. • VPN tunnel to client is not set up. • Client FTP site is unavailable. • Network outage occurs during transfer.	Verify that you have cleared the **Block Startup** option. For more information, see step 5.b of Configuring an FTP listener to access BIX data extracts in Pega Cloud environments. For other issues, check the network connections between you and your Pega Cloud environment.
File transfer is not processed by the BIXFTPListener after trying to transfer files to client SFTP location.	This error usually indicates that the ZIP file content has been modified or tampered with since the time it was generated in the Pega Cloud environment. This usually	Turn off checksum validation that detects that the file has been modified through the associated dynamic system setting.

©2025 Pegasystems Inc

Symptom	Explanation	Solution
Error logged: `Error: Failed check sum validation for <Filename>.zip`	does not happen in a production environment.	• DSS name: `BIX/ignoreValidation` • Ruleset: *Pega-RULES* • Value: `true` For more information about editing a DSS value, see Editing a dynamic system setting.
The file transfer fails with no error in the BIX logs.	Pega Platform places BIX logs in temporary file location.	Verify your BIX logs to identify the temporary location of all the extracted files on the Pega Cloud instance. File a service request for Pega Cloud services to retrieve the files.
	Pega Platform creates an `.ERR` file instead of a `.ZIP` after file transfer.	Make a request to Pega Cloud services to recover the file by modifying `.ERR` to `.ZIP`.

Extraction process fails with query timeout exception

Symptom	Explanation	Solution
An extraction process fails with a query timeout exception error.	A query timeout exception error can occur when the rule runs against a database table with a significantly larger amount of records than in the previous extract.	Increase the query timeout value in DSS. • DSS name: `BIX/selectQueryTimeout` • Ruleset: *Pega-RULES*

©2025 Pegasystems Inc

Symptom	Explanation	Solution
		NOTE: Restarting Pega environment is not required. For more information about editing a DSS value, see Editing a dynamic system setting.

Generated CSV extract or manifest file cannot be parsed

Symptom	Explanation	Solution
When you run an extraction process using a Job Scheduler with additional parameters, Pega Platform cannot parse the generated CSV extract or CSV manifest file through a downstream process, such as a file listener using a parse delimiter rule.	CSV parsing failures can occur when the parse delimiter in Pega Platform interprets the CSV extract and misinterprets the file usage of commas.	Add the `-l` parameter when running the extraction process through the Job Scheduler. For more information, see the following topics. • BIX parameters and command-line arguments • Using the Job Scheduler rule to extract data with BIX

©2025 Pegasystems Inc

Java SQLException for schema generation

Symptom	Explanation	Solution
Schema generation for the target database fails with a Java Exculpation, and produces the following error in the log files: `Error: "java.sql.SqlException: A PRPC-supported database platform was not found:`	When using a database schema as an output, Pega Platform must support the database version that you are using.	Do not extract data to a database that Pega Platform does not support. For more information about supported databases, see the Pega Platform Support Guide.

System time zone ignored

Symptom	Explanation	Solution
When you run an extraction using the Job Scheduler rule, the BIX extraction process does not consider the time zone when you use the following parameters: `-u <start date>`, `-U <end date>`, `-d <start date>` and `-D <end date>`.	The time zone information must also be passed with the parameters `-u <start date>`, `-U <end date>`, `-d <start date>` and `-D <end date>`. The time zone must reflect the thread from which the extraction process runs.	Include time zone information in parameters for`-u <start date>`, `-U <end date>`, `-d <start date>` and `-D <end date>`. For more information, see BIX parameters and command-line arguments

Specifying configuration settings for BIX in a Pega Cloud environment

Ensure that you have specified the following configuration settings to your meet your extraction process requirements.

PEGA ©2025 Pegasystems Inc

Location	Setting name	Description	Default value
DSS	BIX/generateXSDForAllProperties Ruleset: *Pega-RULES*	Generates XSD when you select GetAllProperties for the XML type output. The default value (false) disallows the generation of XSD for GetAllProperties.	false
DSS	BIX/selectQueryTimeout Ruleset: *Pega-RULES*	Specifies the time in seconds before the query to fetch records from the source database times out.	30s
DSS	BIX/setCommandLineDelimiterForCSVManifestFile Ruleset: *Pega-RULES*	By default, the command line option -l is applicable to the CSV output file only. Change the default value of this DSS to true to override the default setting and enable this option for the CSV Manifest as well.	false

©2025 Pegasystems Inc

Location	Setting name	Description	Default value
		For more information, see the following topics: • BIX parameters and command-line arguments • Using the Job Scheduler rule to extract data with BIX	
DSS	`BIX/sharedPath` Ruleset: *Pega-RULES*	Specifies the location of the shared path where files are copied. Applicable only when `BIX/useCustomersSFTP` DSS is set to true.	`file://pegacloudrepository:/bix`
DSS	`BIX/useCustomersSFTP` Ruleset: *Pega-RULES*	Specifies the client SSH File Transfer Protocol (SFTP) site for data transfer. The default value (false) overrides this to use the Pega	`false`

©2025 Pegasystems Inc

Location	Setting name	Description	Default value
		Cloud SFTP data transfer site. For more information, see Configuring the Pega Cloud SFTP service to access BIX extract files.	
DSS	BIX/relativeDestinationPathOnCloud Ruleset: *Pega-RULES*	Specifies the location in the Pega Cloud Services File Storage repository to send extract files.	<Customer_Rootpath>/bix
DSS	BIXFTPRelativeRemotePath Ruleset: *Pega-RULES*	Specifies the relative path on the client SFTP site to send extract files.	
DSS	BIX/ignoreValidation Ruleset: *Pega-RULES*	Specifies ignoring the checksum validation when the File Listener processes the extract files from the path represented by the BIX/SharedPath DSS.	false

©2025 Pegasystems Inc

Location	Setting name	Description	Default value
		The default value (false) enables checksum validation. For more information, see step 8 of Creating and running an Extract rule.	

- Configuring an FTP listener to access BIX data extracts in Pega Cloud environments
- BIX parameters and command-line arguments
- Using the Job Scheduler rule to extract data with BIX

Extracting data to custom output repositories

Eliminate the need to push data through the Pega Cloud repository by specifying a custom output repository for the files that you extract by using Business Intelligence Exchange (BIX). In this way, you save time and extract data directly to the specified repository.

1. Access an Extract Rule:
   - To access the existing Extract Rule, in the navigation pane of Dev Studio, click **Records** > **SysAdmin** > **Extract**, and then select the Rule that you want to edit.

> **NOTE:**
>
> Only one Extract Rule in a given class can be enabled for real-time data extraction, hence the system does not permit using the Save

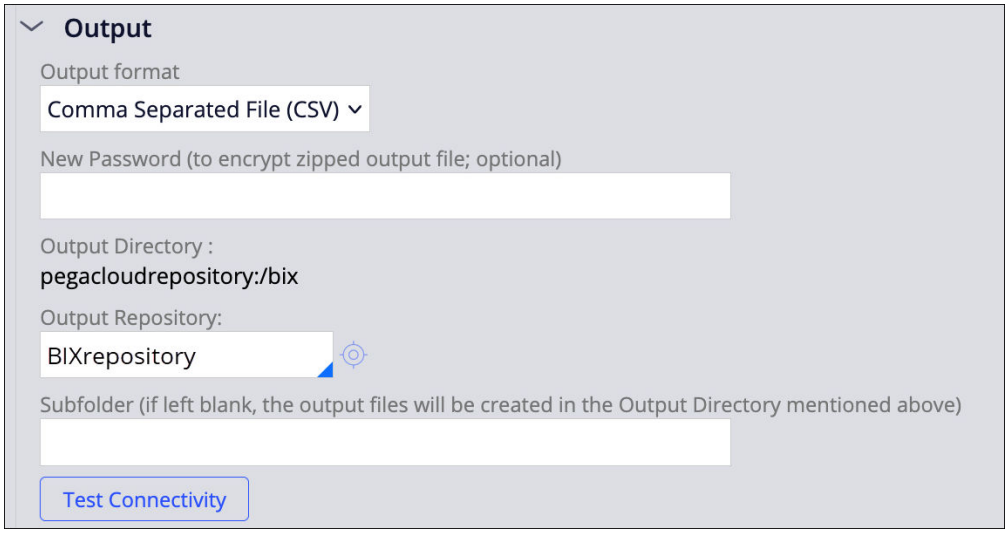
©2025 Pegasystems Inc

> as option on Extract Rules that already have the real-time data extraction enabled, and displays an error. If you want to use the Save as option, disable the real-time data extraction first.

- To create a new Extract Rule, in the header of Dev Studio, click Create > SysAdmin > Extract.

  For more information, see Creating and running an Extract rule.

2. On the Batch Data Extraction tab, in the **Output** section, specify the output repository:
   a. In the Output format list, select either Comma Separated File (CSV) or XML.
   b. **Optional:** To encrypt the output files, in the New password field, enter the password.
3. In the Output repository field, specify the repository to which you want to extract files.
4. **Optional:** To specify a subfolder for the output files, in the Subfolder field, enter the location name.
   The following figure shows a sample Extract Rule with *BIXrepository* specified as the custom output repository:



Specifying custom repository

©2025 Pegasystems Inc

5. In the upper-right corner of the Extract Rule, click **Save**.

- Extracting data in batch mode

Overview of BIX extractions in on-premises systems

Pega Business Intelligence Exchange (BIX) extracts application Class data from your on-premises Pega Platform environment. For on-premises systems, BIX is available when you purchase and install the BIX application.

After enabling the BIX Ruleset for your on-premises application, you can extract application Class data manually through the Extract rule, through the Job Scheduler rule, or through the command-line interface. When you run extracts in an on-premises environment, note the following:

- CSV, XML, manifests, and DDL files are sent to a directory local to your application server.
- Database extracts require providing your database administrator (DBA) a DDL file required to format external database tables.

Running BIX extractions

Pega Business Intelligence Exchange (BIX) extracts application Class data from your on-premises Pega Platform environment. Business Intelligence Exchange (BIX) extracts for on-premises applications are available to run through an Extract rule, Job Scheduler rule, or command-line interface.

1. Enable BIX on your application. For more information, see Enabling the BIX Ruleset.
2. Create and run an initial extract rule. For more information, see Creating and running an Extract rule.:
3. Configure the Pega Platform database from which you want to extract data. For more information, see Configuring the source database for BIX command line extractions .

©2025 Pegasystems Inc

4. For Database Schema extractions, configure the target database to which you want to extract data. For more information, see Configuring the target database for BIX command line extractions.
5. **Optional:** To save information about the command-line extractions in a log file, configure command line BIX logging. For more information, see Configuring BIX logs for command-line extractions in on-premises systems.
6. To store usernames and passwords in an encrypted form, enable password encryption for command-line BIX extractions. For more information, see Enabling password encryption for BIX command-line extractions in on-premises systems.
7. Run the command line extraction process. For more information, see Running a BIX command-line extraction in on-premises systems.

BIX parameters and command-line arguments

Pega supports running Business Intelligence Exchange (BIX) from a Pega Platform *Job Scheduler* rule, or the command-line, in on-premises systems. You can input various parameters to filter what is extracted from your application as described in the following table.

Parameter	Description
-a <username>	Rule resolve *Rule-Admin-Extract* instances based on the user Access Group. **NOTE:** This option is not available when running an Extract rule with pxExtractDataWithArgs.
-b <number of updates before a batch is issued>	Specify the batch size (the number of extracted Class instances) for commits to a relational database.

©2025 Pegasystems Inc

Parameter	Description
	This option only applies to databases with JDBC drivers that support batch updates. **NOTE:** This option is not available when running an Extract rule in the Pega Cloud.
-B	Specify how to format extracted Boolean values. The options are: • 10 • YN • YesNo • yesno • TF • TrueFalse • truefalse • YESNO • TRUEFALSE
-c	Include children classes. Use this parameter to extract the data for each sibling Class of the Class (Class group/ non-Class group) for which the extract is defined. **NOTE:** This option is not available for high-throughput CSV downloads. For more information,

©2025 Pegasystems Inc

Parameter	Description
	see Optimizing BIX data extraction performance.
-d <start date>	Only extract instances with a *pxCreateDateTime* equal to or greater than <start date>. <start date> must be in the format “20110823T164017.000”, following the Java pattern “yyyyMMddTHHmmss.SSS”. **NOTE:** Clear the **Use Last Updated Time as Start** check box on the **Filter Criteria** tab of the Extract form when you use this parameter, because it does not override the check box setting.
-D <end date>	Only extract instances with a *pxCreateDateTime* equal to or less than <end date>. <end date> must be in the format “20110823T164017.000”, following the Java pattern “yyyyMMddTHHmmss.SSS”. **NOTE:** Clear the **Use Last Updated Time as Start** check box on the **Filter Criteria** tab of the Extract

©2025 Pegasystems Inc

Parameter	Description
	form when you use this parameter, because it does not override the check box setting.
-f	Force the extract to stop at the first error that it encounters. This option only works with a batch size greater than 1. See BIX logging and error handling for information on conditions where encountering an error does not force the extract to stop. **NOTE:** This option is not available when running an Extract rule in the Pega Cloud.
-g <start date>	Only extract instances with a *pxCommitDateTime* equal to or greater than <start date>. <start date> must be in the format "20110823T164017.000", following the Java pattern "yyyyMMddTHHmmss.SSS" and relative to the local time zone. **NOTE:** Clear the **Use Last Update Time as Start** check box on the **Filter Criteria** tab of the Extract

©2025 Pegasystems Inc

Parameter	Description
	form when you use this parameter, because it does not override the check box setting.
-G <end date>	Only extract instances with a *pxCommitDateTime* equal to or less than <end date>. <end date> must be in the format "20110823T164017.000", following the Java pattern "yyyyMMddTHHmmss.SSS" and relative to the local time zone. **NOTE:** Clear the **Use Last Update Time as Start** check box on the **Filter Criteria** tab of the Extract form when you use this parameter, because it does not override the check box setting.
-i <InstanceName>	Specify the *pxInsName* of the *Rule-Admin-Extract* or a comma-delimited list of *Rule-Admin-Extract* instances. **NOTE:** This option is not available when running an Extract rule with pxExtractDataWithArgs.

©2025 Pegasystems Inc

Parameter	Description
-I <path and filename>	Use an Extract rule specified in an external file in XML format. **NOTE:** If an extract has any filter that uses unexposed property references that are passed in by the external file, then extraction fails with the following exception: “Please enter or select a property that is exposed as a column in the database.” This option is not available when running an Extract rule with pxExtractDataWithArgs.
-J	Specify the time zone that you want to convert DateTime properties to when extracting to XML or CSV output. You can specify the time zone in the following formats: • Time zone with an ID: For example, to specify America/Los_Angeles. Specify this time zone as "-J""America/Los_Angeles" • Three letter time zone: For example, to specify India Standard Time, use "-J" "IST"

©2025 Pegasystems Inc

Parameter	Description
	• UTC+/-dd:dd: Specify UTC plus or minus hours and minutes. For example, "-J" "UTC+5:30" If not specified, the default is GMT.
-k <Identifier>	Add a unique identifier for this run. Use alphanumeric characters or underscores only with no spaces. **NOTE:** This option is not available when running an Extract rule in the Pega Cloud.
-l <delimiter>	Specify a user-defined character as the delimiter separating extracted data. **NOTE:** The following characters are not available for use as delimiters, as they signify other things to BIX: • Double-quote (“) • Asterisk (*) • Hyphen (-) BIX also does not support multi-line delimiters. If the delimiter length is greater than 1, then BIX reports an error.

©2025 Pegasystems Inc

Parameter	Description
-n	Retrieves Class instances that satisfy the filter conditions with an Extract rule from the Pega Platform production database without sorting them by how recent they are. This improves performance, but makes it impossible to perform a partial re-extraction following any errors.
-o	Overrides the default setting that combines filter conditions specified on *pxCreateDateTime* and *pxUpdateDateTime* using the AND Boolean operator and instead combines them using the OR Boolean operator.
-p <password>	Rule resolve *Rule-Admin-Extract* instances based on the user Access Group. **NOTE:** This option is not available when running an Extract rule with pxExtractDataWithArgs.
-P	Overrides the output directories specified on the **File Specification** tab in the Extract rule form. When specified, all output files are directed to this location.

©2025 Pegasystems Inc

Parameter	Description
	NOTE: This option is not available when running an Extract rule with pxExtractDataWithArgs.
-s <number>	No longer supported.
-t <threads>	No longer supported.
-u <start date>	Only extract instances with a *pxUpdateDateTime* equal to or greater than <start date>. <start date> must be in the format “20110823T164017.000”, following the Java pattern “yyyyMMddTHHmmss.SSS”. You can also optionally include the time zone, for example, "20110823T164017.000 GMT". If you do not include the time zone, the time zone is the locale that is set on the thread or the default locale of "en_US" and time zone of "EST". **NOTE:** Clear the **Use Last Updated Time as Start** check box on the **Filter Criteria** tab of the Extract form when you use this parameter because it does not override the check box setting.

©2025 Pegasystems Inc

Parameter	Description
-U <end date>	Only extract instances with a *pxUpdateDateTime* equal to or less than <end date>. <end date> must be in the format “20110823T164017.000”, following the Java pattern “yyyyMMddTHHmmss.SSS”. You can also optionally include the time zone, for example, "20110823T164017.000 GMT". If you do not include the time zone, the time zone is the locale that is set on the thread or the default locale of "en_US" and time zone of "EST". **NOTE:** Clear the **Use Last Updated Time as Start** check box on the **Filter Criteria** tab of the Extract form when you use this parameter because it does not override the check box setting.
-x	Include header and footer information for XML or CSV output. **NOTE:** This option is not available for high-throughput CSV downloads. For more information,

©2025 Pegasystems Inc

Parameter	Description
	see Optimizing BIX data extraction performance.
-X <Path><FileName>	Generate only a set schema file for the export. No data is exported. To use the -X parameter, the output format of the *Rule-Admin-Extract* rule that you specify must be **Database schema**. If the output format is XML or CSV, a PRRuntimeException is generated. **NOTE:** This option is not available when running an Extract rule in the Pega Cloud.
-z <start key>	Only extract instances with a *pzInsKey* equal to or greater than <start key>. **NOTE:** The comparison is character by character with the *pzInsKey* and is not an integer comparison with the work object's ID.
-Z <end key>	Only extract instances with a *pzInsKey* equal to or less than <end key>.

©2025 Pegasystems Inc

Parameter	Description
	NOTE: The comparison is character by character with the *pzInsKey* and is not an integer comparison with the work object's ID.

The *pzInsKey* values of Class instances whose extraction resulted in errors are logged automatically. You do not need to set a flag to enable this behavior.

NOTE: When running a BIX extract from command-line on UNIX using the -u, -U, -d, or -D parameters, you need to set the timezone using a JVM argument, such as: `-Duser.timezone="America/New York"`.

Running a BIX command-line extraction in on-premises systems

After creating a Business Intelligence Exchange (BIX) Extract rule in Pega Platform, you can directly call the *ExtractImpl* Java Class from the Pega Platform distribution directories to run an BIX extraction process from the command line. Running a BIX extraction process from the command line allows for running an extraction process without requiring the use of Pega Platform to run the process.

This content only applies to on-premises systems and is not supported in Pega Cloud® environments.

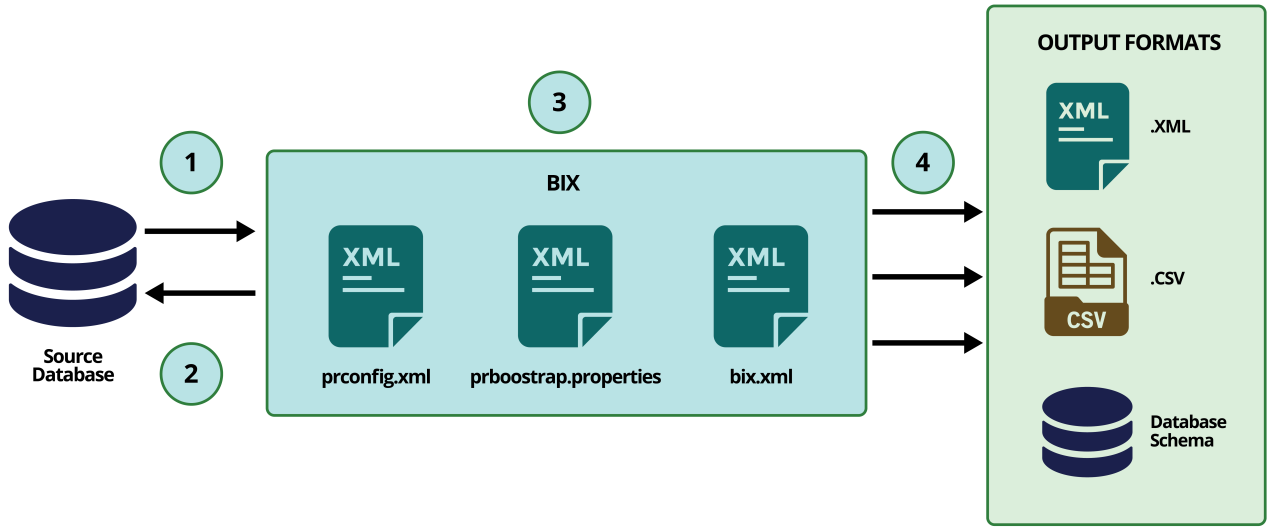
During command-line BIX extraction, a stand-alone instance of the PegaEngine is started without requiring opening Pega Platform to run the extract process without activating the Extract rule. The database user that runs the command-line BIX extraction must have all privileges that are needed to start the engine.

©2025 Pegasystems Inc

You can add other optional parameters that filter the data by any property, including date ranges, and specify other processing options. For more information, see Setting optional command-line BIX parameters.

Running an Extract rule from the command line works in the following way:

BIX extraction process from the command line



1. BIX accesses the Pega Platform (source) database by using the information that is defined in the BIX `prconfig.xml` and `prbootstrap.properties` file.
2. BIX `prbootstrap.properties` file starts up the PegaEngine with the Pega Platform source database details, and then gets the `ExtractImpl.Class` code.
3. With the information that is configured in the Java method, `bix.xml` file, and specified in the Extract rule, BIX extracts the specified data from the Pega Platform source database.
4. The extracted data is output in the format that is specified in the Extract rule.

©2025 Pegasystems Inc

- **Configuring the source database for BIX command-line extractions in on-premises systems**
- **Configuring the target database for BIX command-line extractions in on-premises systems**
- **Configuring optional prconfig.xml settings in on-premises systems**
- **Configuring BIX logs for command-line extractions in on-premises systems**
- **Enabling password encryption for BIX command-line extractions in on-premises systems**
- **Running a BIX command-line extraction in on-premises systems**
- **Running a BIX command-line extraction using ANT build files in on-premises systems**

- Extract Rules overview
- Configuring optional prconfig.xml settings in on-premises systems
- Creating and running an Extract rule

Configuring the source database for BIX command-line extractions in on-premises systems

Configure the source database by editing the `prbootstrap.properties` and `prconfig.xml` files in the configuration directory of your BIX distribution to add database connection settings that specify Pega Platform database from which you extract data. Configuring the source database permits command line extractions

The `prconfig.xml` file contains sample database connection settings for an SQL database that you can modify for your Pega Platform database.

©2025 Pegasystems Inc

NOTE: The following steps outline the procedure for specifying typical database connection settings. Consult your database administrator (DBA) to resolve requirements that are specific to your installation.

This content only applies to on-premises systems and is not supported in Pega Cloud® environments.

Before you begin:

Before you run a BIX extraction process from the command-line, you must accomplish the following tasks:

- Enable the BIX Ruleset. For more information, see Enabling the BIX Ruleset.
- Create a BIX Extract rule. Run the extract as a user that belongs to the same access group as the user that runs the Pega Platform web node. For details, see Creating and running an Extract rule. Note the following BIX extraction information:
  - The label of the Extract rule
  - The output format of the extract files (CSV, XML, or Database Schema)
  - The extraction directory where you want to place the extract files
- Access the following files within the distribution directories of Pega Platform:
  - `PRPC-8.x.x-8xx\Additional_Products\BIX\config\prconfig.xml`
  - `PRPC-8.x.x-8xx\Additional_Products\BIX\prbootstrap.properties`
- Download your appropriate JAR file for your source database and place it in the following directory:
  - `\PRPC-8.x.x-8xx\Additional_Products\BIX\lib`
- Access the following database information:

©2025 Pegasystems Inc

- The database type (PostgreSQL, Oracle, IBM) that is used for the source database, including the following:
  - The Java Database Connectivity (JDBC) URL. For more information, see your database documentation.
  - The associated JAR file for your source database driver downloaded in the `lib` folder of your distribution directory.
  - The database driver Class for the source database. For more information, see your database documentation.
- The rule schema of the source Pega Platform database.
- The data schema of the source Pega Platform database.
- The username and password credentials for the source database. For more information about configuring a database using JDBC URLs, see Creating and updating external relational database instances with JDBC URLs.

1. Open the `prbootstrap.properties` file with your text and source-code editor, such as Notepad++.
2. Uncomment (remove the # symbol) from the following lines:

```
com.pega.pegarules.bootstrap.allclasses.dbcpsource=example.oracle
```

```
example.oracle.url=jdbc:oracle:thin:@hostname:1521:rules11g
```

```
example.oracle.jdbc.Class=oracle.jdbc.OracleDriverle.oracle.username=<username>
```

```
example.oracle.password=<password>
```

```
com.pega.pegarules.bootstrap.allclasses.schema=<rule schema>
```

```
com.pega.pegarules.bootstrap.datatables.schema=<data schema>
```

©2025 Pegasystems Inc

3. Enter your source database information to the elements in the uncommented lines.
   Replace `example` and `oracle` in each element with your *database name* and the *database type* as in the following examples. Include angle brackets (<>) as shown in specific input values.
   a. For the `com.pega.pegarules.bootstrap.allclasses.dbcpsource` element, enter `NameForDatabase.DatabaseType`.
   b. For the `NameForDatabase.Database.url=` element, enter the `Database URL`.
      For additional database URL details, refer to your database documentation.

For example:

Database type	Driver Class
Oracle	`jdbc:oracle:thin:@//serverName:port/service-name-or-SID`
IBM	`jdbc:db2://serverName:port/dbName`
SQL Server	`jdbc:sqlserver://your_sql_server_host:1433;SendStringParametersAsUnicode=false;SelectMethod=cursor;ProgramName=${NodeName}:${SystemName}.${ConnectionID}r`
PostgreSQL	`jdbc:postgresql://localhost:5432/dbName`

©2025 Pegasystems Inc

c. For the `NameForDatabase.Database.username=<username>` element, enter `Username for database connection`.
d. For the `NameForDatabase.Database.password=<password>` element, enter `Password for database connection`.
e. For the `Database.jdbc.Class=` element, enter `Database driver Class`.
For additional database driver details, refer to your database documentation.

For example:

Database type	Driver Class
Oracle	`oracle.jdbc.OracleDriver`
IBM	`com.ibm.db2.jcc.DB2Driver`
SQL Server	`com.microsoft.sqlserver.jdbc.SQLServerDriver`
PostgreSQL	`org.postgresql.Driver`

f. For the `com.pega.pegarules.bootstrap.allclasses.schema=` element, enter `<rule schema of source database>`.
g. For the `com.pega.pegarules.bootstrap.datatables.schema=` element, enter `<data schema of source database>`.

4. Save and close the `prbootstrap.properties` file.
5. Open the `prconfig.xml` file with your text and source-code editor, such as Notepad++.
6. Uncomment (remove the #) from the following lines:

```
<env name="database/drivers" value="oracle.jdbc.OracleDriver;com.ibm.db2.jcc.DB2Driver" />
```

©2025 Pegasystems Inc

```
<env name="database/databases/PegaRULES/url" value=" jdbc:sqlserver://serv
erName:1433;SelectMethod=cursor;SendStringParametersAsUnicode=false" />
```

```
<env name="database/databases/PegaRULES/userName" value="a_username"
/>
```

```
<env name="database/databases/PegaRULES/password" value="a_password " /
>
```

```
<env name="database/databases/PegaDATA/url" value="jdbc:sqlserver://server
Name:1433;SelectMethod=cursor;SendStringParametersAsUnicode=false " />
```

7. Above the line containing the element `<env name="database/drivers" value=" " ↗`, add the following elements:

```
<env name="database/databases/PegaDATA/userName" value="a_username " /
>
```

```
<env name="database/databases/PegaDATA/password" value="a_password " />
```

> **NOTE:** If you plan to enable password encryption for the BIX command-line extraction process, you do not need to uncomment the username (`/username" value="a_username" ↗`) and password (`/password" value="a_password↗`) elements. For more information, see Enabling password encryption for BIX command-line extractions in on-premises systems.

8. Enter your source database schema information to the elements in the uncommented lines.
   Replace the example `Oracle` input values with input values for your source database in quotation marks as shown in the following entries.

©2025 Pegasystems Inc

a. For the `<env name="database/databases/PegaRULES/defaultSchema" value=" " ↗` element, enter "*name of the rule schema of source database*".
b. For the `<env name="database/databases/PegaDATA/defaultSchema" value=" " ↗` element, enter "*name of the data schema for source database*".
c. For the `<env name="database/drivers" value=" " ↗` element, enter "*Database driver Class*".
   For additional database driver details, refer to your database documentation.

   For example:

Database type	Driver Class
Oracle	`oracle.jdbc.OracleDriver`
IBM	`com.ibm.db2.jcc.DB2Driver`
SQL Server	`com.microsoft.sqlserver.jdbc.SQLServerDriver`
PostgreSQL	`org.postgresql.Driver`

d. For the `<env name="database/databases/PegaRULES/url" value=" " ↗` element, enter "*database URL*".
   For additional database URL details, refer to your database documentation.

   For example:

Database type	URL example
Oracle	`jdbc:oracle:thin:@//serverName:port/service-name-or-SID`

©2025 Pegasystems Inc

Database type	URL example
IBM	`jdbc:db2://serverName:port/dbName`
SQL Server	`jdbc:sqlserver://your_sql_server_host:1433;SendStringParametersAsUnicode=false;SelectMethod=cursor;ProgramName=${NodeName}:${SystemName}.${ConnectionID}`
PostgreSQL	`jdbc:postgresql://localhost:5432/dbName`

e. For the `<env name="database/databases/PegaDATA/userName" value= ↗` element, enter "*database username*".
f. For the `<env name="database/databases/PegaDATA/password" value="a_password" ↗` element, enter "*database password*".
g. For the `<env name="database/databases/PegaRULES/userName" value="a_username" ↗` element, enter "*database username*".
h. For the `<env name="database/databases/PegaRULES/password" value="a_password" ↗` element, enter "*database password*".

NOTE: Set the value of `database/databases/PegaDATA/userName` and `database/databases/PegaDATA/password` to the user name and password of a user for the Pega Platform database.

©2025 Pegasystems Inc

What to do next:

If you chose the XML or CSV format for the BIX extraction process when you create the Extract rule, you can configure BIX logging, enable password encryption, and run the command-line extraction. For more information, see Configuring BIX logs for command-line extractions in on-premises systems, Enabling password encryption for BIX command-line extractions in on-premises systems, and Running a BIX command-line extraction in on-premises systems.

If you chose the Database Schema format for the BIX extraction process when you created the Extract rule, configure your target database. For details, see Configuring the target database for BIX command-line extractions in on-premises systems.

- Configuring the target database for BIX command-line extractions in on-premises systems
- Creating and updating external relational database instances with JDBC URLs

Configuring the target database for BIX command-line extractions in on-premises systems

The BIX Database Schema output format writes the extracted data directly to a target database. If you want to use the Database Schema output format for the BIX command-line extraction process, add a second set of database connection settings in the `prconfig.xml` file that specify the target database into which you extract data.

To support this output, provide a second set of database connection settings specifying the target database where you want to load the data.

This content only applies to on-premises systems and is not supported in Pega Cloud® environments.

©2025 Pegasystems Inc

Before you begin:

You must configure the source database before configuring the target database. For details, see Configuring the source database for BIX command-line extractions.

- You must have access to the prconfig file located in the following distribution directory: `PRPC-8.x.x-8xx\Additional_Products\BIX\config\prconfig.xml`
- You must have access to the following target database information:
  - The Output Database name you defined when creating the Extract rule.
  - The database type (PostgreSQL, Oracle, IBM) used for the target database, including the Java Database Connectivity (JDBC) URL. See your database documentation for details.
  - The rule schema name of the target database.
  - The data schema name of the target database.
  - The username and password credentials for the target database.

1. In the BIX `prconfig.xml` file, copy and paste the database connection settings that you created for the source database.
2. Below the line containing the element `<env name="database/databases/PegaDATA/password" value="a_password" ↗`, copy the following the database connection elements for the target database:

```
<env name="database/databases/<TargetDBName>/defaultSchema"
value=" " ↗
<env name="database/databases/<TargetDBName>/username" value
=" " ↗
<env name="database/databases/<TargetDBName>/password" value
=" " ↗
```

©2025 Pegasystems Inc

3. In each element name, change the string `<TargetDBName>` to the name of the target database that you specified as the `Output Database Name` in the Extract rule.

For example:

Original element	Updated element
`database/databases/PegaRULES/url`	`database/databases/TargetDB/url`
`database/databases/PegaRULES/userName`	`database/databases/TargetDB/userName`
`database/databases/PegaRULES/password`	`database/databases/TargetDB/password`

4. Create a formatted database table for the command-line extraction process by generating a DDL file. For more information, see Generating a DDL file for command-line Database Schema extractions. For more information, consult your database administrator and see the Pega Community article How to configure non-J2EE database connections in the prconfig.xml file.

- Configuring the source database for BIX command-line extractions in on-premises systems
- Running a BIX command-line extraction in on-premises systems

Configuring optional prconfig.xml settings in on-premises systems

You can add parameters to the `prconfig.xml` file that filter the data by any property, including date ranges, enable or disable chaining, and specify whether records added to the database while the extract is running are included in the extract.

This content only applies to on-premises systems and is not supported in Pega Cloud® environments.

©2025 Pegasystems Inc

Add the following settings to the end of the `prconfig.xml` file:

- **Optimize XML output** – Set to true by default to facilitate extracting a large number of Class instances to XML: `<env name= “compatibility/BIXUseOptimizedClipboardXML” value= “true” ↗`

- **Disable forward chaining** – Enable or disable forward chaining calculations of properties when loading the Clipboard for BIX extract: `<env name= “compatibility/BIXdisableForwardChaining” value= “true” ↗`

  The default setting is TRUE.

- **Disable backward chaining** – Enable or disable backward chaining calculations of properties when loading the Clipboard for BIX extracts: `<env name=”compatibility/BIXdisableBackwardChaining” value = “true” ↗`

  The default setting is TRUE.

- **Disable declarative processing** – Enable or disable declarative processing: `<env name=”declarative/enabled” value = “false” ↗`

  > **NOTE:** Only use this setting when you are running BIX from the command line. If you use this setting on a node to improve performance, you cannot open or create rules on that node.

- **Include Class instances added during a batch run** – Enable or disable whether Class instances added to the Pega Platform database after a batch run begins execution are included in the extract: `<env name=”bix/useHistoryClasses” value=”true” ↗`

©2025 Pegasystems Inc

> **NOTE:** `bix/useHistoryClasses` only works when using the **Use Last Updated Time as Start** filter option. If command-line parameters such as –u, -U, -d, and -D are passed while running the BIX script, `bix/useHistoryClasses` will not have any effect even if it is set to `true`. For details on command-line parameters, see BIX parameters and command-line arguments.

- Running a BIX command-line extraction in on-premises systems

Configuring BIX logs for command-line extractions in on-premises systems

You can configure Business Intelligence Exchange (BIX) logs for command-line extractions by editing the `prlog4j2.xml` file. BIX logs capture timestamped progress each table batch extracted, interruptions in the extraction process, source and target database connectivity errors. You can locate the `prlog4j2.xml` file in the `config` folder of your Pega Platform distribution.

For more information about logging levels, see Logging for VM-based deployments and containerized deployments on Kubernetes.

This content only applies to on-premises systems and is not supported in Pega Cloud® environments.

1. Open the `prlog4j2.xml` file with your text and source code editor Notepad++.
2. In the `Loggers` Class, find the `<Logger name="com.pega.pegarules.data.internal.access.ExtractImpl" additivity="false" level="debug"> <AppenderRef ref="BIX"↗` element.
3. Edit the parameter `level=" "` to adjust your logging level.
4. Save and close the `prlog4j2.xml` file.
5. Review your log files. For more information, see BIX logging.

©2025 Pegasystems Inc

- Configuring the target database for BIX command-line extractions in on-premises systems
- Configuring the source database for BIX command-line extractions in on-premises systems
- Enabling password encryption for BIX command-line extractions in on-premises systems

Enabling password encryption for BIX command-line extractions in on-premises systems

Enable security for the database username and password by implementing a Java Cryptography Extension (JCE) keyring encryption. To do so, create a `pegarules.keyring` file and append the encryption information to your command-line extraction Java method.

Business Intelligence Exchange (BIX) command-line extract files store usernames and passwords within log and configuration files in plain text. Enabling password encryption for BIX command-line extractions stores usernames and passwords in an encrypted form.

This content only applies to on-premises systems and is not supported in Pega Cloud® environments.

1. Open the `prconfig.xml` file with your text and source-code editor, such as Notepad++.
2. Below the line that contains the element `<env name="database/databases/PegaDATA/password" value="a_password " ↗`, add the following keyring prefix element to create the `BIX-extract.kerying`: `<env name="identification/KeyringPrefix" value="BIX-extract" ↗`
3. Specify an encryption algorithm by adding the following `identification/KeyringAlgorithm` element:

```
<env name="identification/KeyringAlgorithm" value="AES" />
```

©2025 Pegasystems Inc

4. Specify the length of the key that you want to use to encrypt the file by adding the following `identification/KeyringLength` element:

```
<env name="identification/KeyringLength" value="256" />
```

5. Save and close the `prconfig.xml` file.
6. Run your command-line extraction with the following changes:
   a. Add the `KeyringImpl` Java Class: `com.pega.pegarules.exec.internal.util.crypto.KeyringImpl`.
   b. Add the directory and file name in which you want to generate the `pegarules.keyring` file.

      For example:

      `.\config\pegarules.keyring`

   c. Add the directory location of the `prconfig.xml` file.

      For example:

      `.\config\prconfig.xml`

   d. Add the *Extract* Java Class and its distribution directory.

      For example:

      `C:\imp`
      `com.pega.pegarules.data.internal.access.ExtractImpl`

   For details about running a command-line extraction process, see Running a BIX command-line extraction in on-premises systems.

   For example:

   Full encryption information:

©2025 Pegasystems Inc

```
com.pega.pegarules.exec.internal.util.crypto.KeyringImpl .\config\pegaru
les.keyring .\config\prconfig.xml C:\imp com.pega.pegarules.data.interna
l.access.ExtractImpl
```

Full command-line extraction with appended encryption information:

```
java -Xms512m -Xmx768m -classpath".;lib\prbootstrap-ap
i.jar;lib\prbootstrap.jar;lib\prdbcp.jar;lib\db2jcc_v95
.jar;lib\jsr94-1.0.jar;$CLASSPATH$ -Dcom.pega.pegarules
.bootstrap.properties.url=config\prbootstrap.properties
-Dpegarules.config=config\prconfig.xml -Dpegarules.logging
.configuration=config\prlog4j2.xml -Dcom.pega.pegarules.b
ootstrap.ignorejndi=true com.pega.pegarules.pub.PegaRUL
ES com.pega.pegarules.crypto.KeyringImpl .\config\pegaru
les.keyring .\config\prconfig.xml C:\imp com.pega.pegarul
es.data.internal.access.ExtractImpl -i PegaSample!Sampl
eBIX
```

Running a BIX command-line extraction in on-premises systems

After you configure your source database environment, you can run the Business Intelligence Exchange (BIX) command-line extraction process.

Before you begin:

This content only applies to on-premises systems and is not supported in Pega Cloud® environments.

Complete the following tasks:

- Create an Extract rule. For more information, see Creating and running an Extract rule.

©2025 Pegasystems Inc

- Obtain the Extract rule label.
- Obtain the directory for the `prconfig.xml`, `prbootstrap.properties`, and `prlog4j2.xml` files. For more information, see Configuring the source database for BIX command-line extractions in on-premises systems.
- Obtain the PegaRules username and password created during the source database configuration.
- If you enabled password encryption, obtain the file path for `pegarules.keyring`. For more information, see Enabling password encryption for BIX command-line extractions in on-premises systems.
- Obtain the names of JAR files in the `lib` directory for the Pega Platform source database.

1. In the command line, begin a Java call by specify your initial and maximum JVM memory with the `Xms` and `Xmx` parameters.
   a. Set the initial Java heap size to 512m (-Xms512m).
   b. Set the maximum heap size to at least 768m (-Xmx768m).

   For example:

   ```
   java -Xms512m –Xmx768m
   ```

2. Reference the file path for the BIX `prconfig.xml`, `prbootstrap.properties`, and (if used) `prlog4j2.xml` files by using the following parameters:
   a. For the `prconfig.xml` file, enter `-Dpegarules.config=<prconfig_filepath>`.
   b. For the `prbootstrap.properties` file, enter `-Dcom.pega.pegarules.bootstrap.properties.url=<bootstrap.properties`
   c. **Optional:** For the `prlog4j2.xml` file, enter `-Dpegarules.logging.configuration=<prlog4j2_filepath>`.

©2025 Pegasystems Inc

For example:

```
-Dpegarules.config=./config/prconfig.xml -Dpegarules.logging.configuration=config/prlog4j2.xml -Dcom.pega.pegarules.bootstrap.properties.url=config/prbootstrap.properties
```

3. If you enabled password encryption, add the file path for the `pegarules.keyring` file.
   For more information, see Enabling password encryption for BIX command-line extractions in on-premises systems.
4. Disable Java Naming and Directory Interface (JNDI) databases to extract from the configured source Java Database Connectivity JDBC URL database by using the following parameter:

   ```
   -Dcom.pega.pegarules.bootstrap.ignorejndi=true
   ```

5. Specify the location of .JAR files that are used in the BIX source database library (`lib`) directory by setting the classpath with the parameter `-classpath`. Include quotes (" ") around the classpath value.

   For example:

   ```
   -classpath ".;lib\prbootstrap.jar;lib\prdbcp.jar;lib\db2jcc_v95.jar;lib\jsr94-1.0.jar;$CLASSPATH$"
   ```

6. **Optional:** To use a username and password with the `-a` and `-p` parameters for the extract files according to your source database configuration, call *com.pega.pegarules.pub.PegaRULES*

NOTE: If you enabled password encryption for BIX command-line extractions, you do not need to call `PegaRules` with the username and

©2025 Pegasystems Inc

password elements. For details, see Enabling password encryption for BIX command-line extractions in on-premises systems.

For example:

```
com.pega.pegarules.pub.PegaRULES com.pega.pegarules.data.internal.access.ExtractImpl -a $USERNAME$ -p $PASSWD$
```

7. Call the Extract Java Class *com.pega.pegarules.data.internal.access.ExtractImpl* and specify an Extract rule label with the `-i` parameter.

For example:

```
com.pega.pegarules.pub.PegaRULES com.pega.pegarules.data.internal.access.ExtractImpl -a $USERNAME$ -p $PASSWD$ -i SampleBIX_CSV
```

8. Enter the node type and node type and node settings for the extract process.

For example:

```
-DNodeType=BIX -DNodeSettings="Pega-IntegrationEngine/disableEpochDateTime=false;Pega-IntegrationEngine/disableEpochDate=false;Pega-RulesEngine/EnableAttributeBasedSecurity=false"
```

9. **Optional:** To customize your configuration, include additional command-line extraction parameters.
   For more information, see BIX parameters and command-line arguments.

For example:

You can add the following functionality with additional BIX parameters:

©2025 Pegasystems Inc

- Filter the data by date range.
- Specify the number of records to process before a batch commit.
- Include child classes in the extract files.

10. **Optional:** To use an external XML file for the extraction process, perform the following actions:

> **NOTE:** This step requires running an instance of Pega Platform

a. In the navigation pane of Dev Studio, click **Records** > **SysAdmin** > **Extract**, and then select the Extract rule to run.
b. On the rule form, click **Actions** > **View XML**.
c. Copy the resulting page and save the content as an XML file.
d. Begin the command-line extraction method as in step 1.
e. When you call *com.pega.pegarules.data.internal.access.ExtractImpl* to specify an Extract rule label, provide the file path in which you created the XML file and the XML file name with the parameter and arguments `-I <filepath> <DDL filename>`.

> **For example:**
>
> ```
> com.pega.pegarules.pub.PegaRULES com.pega.pegarules.data.internal.access.ExtractImpl -I ./XMLpath External_XML
> ```

11. Perform the command-line extraction by running the Java method.

> **NOTE:** For Database Schema extractions, ensure that your DBA formats the target database by using the DDL file. For more information, see Generating a DDL file for command-line Database Schema extractions.

©2025 Pegasystems Inc

For example:

A complete command-line extraction method:

```
java –Xmx768m java -Dpegarules.config=./config/prconfig.xml -Dpegarules.logging.configuration=config/prlog4j2.xml -Dcom.pega.pegarules.bootstrap.properties.url=config/prbootstrap.properties -Dcom.pega.pegarules.bootstrap.ignorejndi=true -classpath ".;lib\prbootstrap.jar;lib\prdbcp.jar;lib\db2jcc_v95.jar;lib\jsr94-1.0.jar;$CLASSPATH$" com.pega.pegarules.pub.PegaRULES com.pega.pegarules.data.internal.access.ExtractImpl -a $USERNAME$ -p $PASSWD$ -i SampleBIX_CSV -DNodeType=BIX -DNodeSettings="Pega-IntegrationEngine/disableEpochDateTime=false;Pega-IntegrationEngine/disableEpochDate=false;Pega-RulesEngine/EnableAttributeBasedSecurity=false"
```

Result:

After the BIX command-line extraction process, your XML and CSV BIX extract files are saved in the local directory that you specified during the Extract rule creation. For database schema extractions, see your external database instance to view the extraction table data.

- **Generating a DDL file for command-line Database Schema extractions**
- Running a BIX command-line extraction in on-premises systems
- Creating and running an Extract rule

Generating a DDL file for command-line Database Schema extractions

For Database Schema extractions, generate a DDL file to share with your database administrator (DBA) so that they can format the external database table for extractions.

©2025 Pegasystems Inc

This content only applies to on-premises systems and is not supported in Pega Cloud® environments.

1. From the command line, begin the Java method with an initial Java call of *com.pega.pegarules.data.internal.access.ExtractImp*.
2. Append the file path in which to create the DDL file and the DDL file name with the parameter and argument `-x <filepath><DDL filename>`.

   For example:

   ```
   com.pega.pegarules.pub.PegaRULES com.pega.pegarules.data.internal.access.ExtractImpl -I ./XMLpath External_XML
   ```

3. Provide your DBA with the DDL file and ensure that your DBA formats the target database by using the DDL file.
4. Return to Running a BIX command-line extraction in on-premises systems to perform a BIX extraction process from the command for Database Schema extract file formats.

Running a BIX command-line extraction using ANT build files in on-premises systems

The Apache Another Neat Tool (ANT) build file acts as a script to run the Business Intelligence Exchange (BIX) command-line extraction process. Call the ANT build file to achieve the same Java functionality of a command-line extraction process without entering the entirety of the extraction Java method in the command line. The BIX ANT file, `bix.xml`, uses the same Java methods as the task Running the BIX command-line extraction process.

This content only applies to on-premises systems and is not supported in Pega Cloud® environments.

1. In your Pega Platform distribution directory, open the `bix.xml` file with your text and source-code editor.

©2025 Pegasystems Inc

The following XML code is displayed:

```
<?xml version="1.0" encoding="UTF-8"?>

<project name  = "Sample" default = "Bix" >

<target name="Bix">
      <property name="ARGLINE" value="PegaSample!Sample"/>

      <echo message="Performing Export"/>

            <condition property="javaVersion" value="6">
                  <matches string="${java.version}" pattern="1.6.*" />
            </condition>

            <echo message="using java version ${javaVersion}"/>

            <java classname="com.pega.pegarules.pub.PegaRULES" failonerror="t
rue" fork="true">
                        <classpath>
                  <fileset dir="lib/">
                                    <include name="*.jar"/>
                  </fileset>
                  <pathelement path="."/>
                  <pathelement path="${java.Class.path}"/>
            </classpath>
                     <jvmarg value="-Xms128m"/>
            <jvmarg value="-Xmx512m"/>
                        <sysproperty key="com.pega.pegarules.bootstrap.properties.
url" value="config/prbootstrap.properties" />
            <jvmarg value="-Dpegarules.config=config/prconfig.xml"/>
                        <jvmarg value="-Dpega.prbootstrap.config=config/prbootstra
p.properties"/>
```

©2025 Pegasystems Inc

```
                    <jvmarg value="-Dcom.pega.pegarules.bootstrap.ignorejndi=true"/>
                    <jvmarg value="-Dpegarules.logging.configuration=config/prlog4j2.xml"/>
                    <jvmarg value="-Dpega.tmpdir=."/>
        <arg value="com.pega.pegarules.data.internal.access.ExtractImpl" />
                <arg line="-i"/>
        <arg value="${ARGLINE}" />
                <!--<arg line="-a"/>
                <arg value="administrator@pega.com"/>
                <arg line="-p"/>
                <arg value="install"/> -->

            </java>
</target>
</project>
```

2. Find and edit the following elements in the `bix.xml` file. Include quotes in the values.

Element	Value
`<property name="ARGLINE" value=" "`	Set to the label of the Extract rule that you want to execute.
`<fileset dir="lib/">`	Enter the name of the library directory. The default name is `lib`.
`<include name="*.jar"↗`	Specify the JAR files that are used in the source database.
`<pathelement path="."↗`	Enter the location of the library directory.
`<pathelement path="$ "↗`	Set the Java Class path.
`<jvmarg value=>`	Specify your JVM memory:

©2025 Pegasystems Inc

Element	Value
	• Set the initial Java heap size to 512m (-Xms512m). • Set the maximum heap size to at least 768m (-Xmx768m)
`<jvmarg value="-Dpegarules.config=config/prconfig.xml"↗`	Add the directory for the BIX `prconfig.xml` file.
`<jvmarg value="-Dpegarules.logging.configuration=config/prlog4j2.xml"↗`	Add the directory for the BIX `prlog4j2.xml` file.
`<jvmarg value="-Dpega.prbootstrap.config=config/prbootstrap.properties"↗`	Add the directory for the BIX `prbootstrap.properties` file.

3. Save and close the `bix.xml` file.
4. Open a command prompt and run the build with the following command:

```
ant -f bix.xml
```

After using an ANT build file to run the BIX command-line extraction process, your XML and CSV extract files are saved in the local directory that you specified during the Extract rule creation. For database schema extractions, see your external database instance to view the extraction table data.

- Extract Rules overview
- Running a BIX command-line extraction in on-premises systems

BIX logging

The PegaBIX log file contains information about Business Intelligence Exchange (BIX) extractions that can be useful to determine what caused a problem during a BIX extraction process. BIX extracts Class data then writes out a batch of records, one batch

PEGA ©2025 Pegasystems Inc

at a time. These events, including any error messages, are recorded in the PegaBIX log file.

- To configure and view log files, see Viewing and configuring BIX logs.

- When a BIX extraction is run from within your application, any errors are displayed on-screen and are also logged.
- Adding parameters to BIX extractions through the Job Scheduler or command-line can stop the extraction process at their first error. For more information, see BIX parameters and command-line arguments

- **Viewing and configuring BIX logs**

- **BIX error handling**

- Running a BIX command-line extraction in on-premises systems
- Using the Job Scheduler rule to extract data with BIX
- Viewing and configuring BIX logs

Viewing and configuring BIX logs

Business Intelligence Exchange (BIX) logs are visible in System Operations and can be downloaded.

1. In the header of Dev Studio, click **Configure** > **System** > **Operations** > **Logs**.
   a. To change the logging level settings for BIX logs, click **Logging level settings**.
   b. In the **logger name** field, enter `com.pega.pegarules.data.internal.access.ExtractImpl`.
   c. In the **Current Level** drop down, select a logging level. Logging levels set to DEBUG or ALERT can lead to BIX performance issues including slow extractions or full log files. For more information about logging levels, see Logging for VM-based deployments and containerized deployments on Kubernetes.
2. Click **Log files**.

©2025 Pegasystems Inc

3. Select the **BIX** to view the BIX logs.
4. Download the log files. Log file locations depend on system deployment. For on-premises deployments, see Downloading log files.
    a. For Pega Cloud Services applications, log in to your **My Support Portal** account.
    b. Navigate to the **My Pega Cloud** tab.
    c. On the **My Pega Cloud** page, click the **Gear** icon.
    d. Select **Download logs**.
    e. Select the BIX **File Type**.
    f. Click **Generate log bundle**, then click **Download the log bundle** in the dialog box to download the BIX log files.

- Extract Rules overview
- BIX logging
- Overview of BIX extractions in Pega Cloud environments
- Overview of BIX extractions in on-premises systems

BIX error handling

Errors normally terminate the Business intelligence exchange (BIX) extract. The extraction may continue to run during certain exceptions:

- An invalid Class definition for a single Class described in a set of XML files
- An invalid property definition
- An invalid property transformation to the specified data type

BatchUpdateException error

Committing an extract batch of Class instances to a relational database may be interrupted because of issues with the data, syntax errors, a database constraint violation, or some other issue raised by the database.

©2025 Pegasystems Inc

Different databases respond differently to batch update exceptions. Generally, the database throws a `BatchUpdateException` error, which BIX records in the PegaBIX log file.

When an error occurs, BIX behaves differently depending on the database platform that it is working with: Oracle, MS-SQL, or Db2 LUW.

Troubleshooting BatchUpdateException for databases other than Oracle

The server continues processing the operations in a batch even if one of the operations caused a database error. The database returns an exception that contains the exact information of the record in error. This information can be used to pinpoint the failure to the *pzInskey* of the record involved.

In this scenario, BIX supports two modes of execution:

- Continue processing records even when there are failures (the default mode)
- Stop processing when the first error is encountered

In default mode, all successful inserts are committed and the *pzInsKeys* of the failing entries are listed in the log file.

To rectify the work object and rerun the extract, specify the same value for *pzInskey* in the –z parameter for Job Scheduler extractions and –Z for command-line extractions.

> **NOTE:** When the `–f` parameter is in effect, the extraction process terminates after the first error is encountered. All successful batch inserts up to that point are committed. The batch that caused the failure is rolled back completely.

Troubleshooting BatchUpdateException for Oracle

Oracle stops processing the remaining operations in the batch if it encounters an error. The exception returned by the database does not point to the exact operation that

©2025 Pegasystems Inc

caused the exception. For that reason, all of the *pzInsKeys* of the records in the batch are included in the BIX error log. The records from the batch that were successfully committed are not indicated because the Oracle driver does not provide that information.

In this scenario, BIX supports two modes of execution:

- Continue processing the records even when there are failures (the default mode)
- Fail on first error

In default mode, all successful inserts up to the record that failed are committed and the entries in the batch that followed the error entry are not processed. All of the *pzInsKeys* of that batch are written out to the log file.

To rectify the error and rerun BIX for the remaining records in the batch, specify the *pzInsKey* range for the remaining records using the –z for Job Scheduler extractions and –Z for command-line extractions.

NOTE:

- When the `-f` option is in effect, the extraction process terminates after the first error is encountered. All of the successful batch inserts up to that point are committed. The batch that caused the failure is rolled back completely.

- Running a BIX command-line extraction in on-premises systems
- Using the Job Scheduler rule to extract data with BIX
- BIX logging

Optimizing BIX data extraction performance

Understanding performance enhancements of the Business Intelligence Exchange (BIX) configurations will help you to maximize the throughput of BIX extractions from your

©2025 Pegasystems Inc

applications. Refer to the following table to incorporate enhancement considerations into your BIX extraction process as needed.

Location of BIX improvement	Performance enhancement consideration
BIX logs	Setting BIX logging to a level of DEBUG or ALERTS can cause performance issues.
BIX ruleform > Definition Tab > Format > Properties	XML-based extractions occur much faster than CSV-based extractions or database schema extractions because data in embedded page lists and page groups does not require normalization in XML as it is being written.
BIX ruleform > Definition Tab > Format > Properties	Configuring fewer properties for BIX extractions and configuring properties that are not nested within a Class structure increases extraction performance.
BIX ruleform > Filter Criteria	Running multiple BIX extractions with different filter conditions that use the same Class in parallel enhances performance. For example, using different filter criteria for separate product lines, geographies, case resolutions, or organizational divisions.
. > Filter Criteria	Enabling incremental BIX extractions limits the properties extracted to those with an updated *pxCommitDateTime* instance when compared to the last extraction to increase subsequent BIX data extraction performance. Properties with an updated *pxCommitDateTime* include classes created or changed by an operator or automatically by the application.

©2025 Pegasystems Inc

Location of BIX improvement	Performance enhancement consideration
	NOTE: The BIX extraction process skips properties that contain *pxCommitDateTime* instances with a `null` value including child classes incorporated into the extraction by the `-c` parameter in the Job Scheduler ruleform.

BIX high-throughput data extractions

Pega Platform uses high-throughput data extractions for CSV extractions for specific databases to improve performance of the BIX extraction process. BIX high-throughput data extractions occur automatically under the following extraction process conditions:

- The BIX extraction process occurs from BLOB-less Class data. For details see Optimizing a property from the user interface.
- The BIX extraction process occurs from an external Postgres database. For details see Creating and updating external relational database instances with JDBC URLs.
- The BIX extraction process uses a CSV format. For details, seeCreating and running an Extract rule.

When you use the high-throughput data extraction option, the following functionality is not available:

- You cannot input new property names for the CSV files when creating the *Extract* rule.
- You cannot append the `-c` parameter to include child classes.
- You cannot append the `-x` parameter to include header and footer information in the output. For details see BIX parameters and command-line arguments and Using the Job Scheduler rule to extract data with BIX.

PEGA ©2025 Pegasystems Inc

- You cannot use the checksum in the manifest summary. For details see Creating and running an Extract rule.

Create a dynamic system setting with keystring `bix/useHighThroughputDownloadForCSV` and set **value** field to `false` to disable high-throughput extractions for CSV extract files from an external Postgres databases with BLOB-less Class data.

- **Extracting large data by using BIX**
- Configuring optional prconfig.xml settings in on-premises systems

Extracting large data by using BIX

Pega Platform™ enables you to divide Business Intelligence Extract (BIX) operations into smaller parts to speed up the extraction process. In this way, you can conveniently extract specific classes, ensuring a more efficient and streamlined workflow.

Extracting datasets with large number of records might cause performance issues, hence dividing the extracts into several parts can improve the extract's performance.

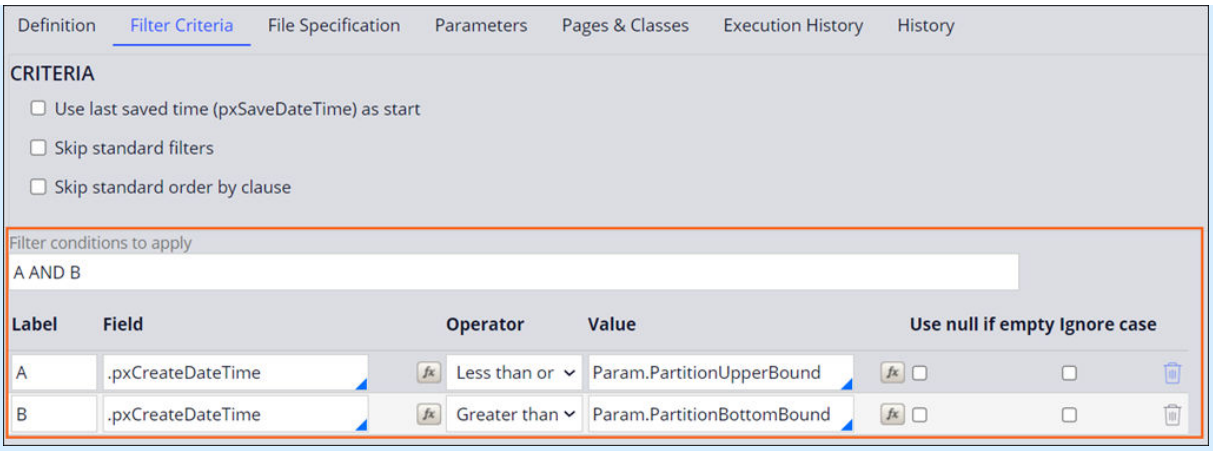
Before you begin:

Create an Extract rule with custom parameters to extract the required data only. For more information, see Creating and running an Extract rule.

> **NOTE:** In the **Filter Criteria** and **Parameters** tabs, specify filter criteria to extract only the data that match your specific requirements. Ensure that the filters are optimized in the database.

The following figure shows a sample Extract rule with specified filters that use the *Create date/time* property:

©2025 Pegasystems Inc



Sample filters

Creating the activity responsible for scheduling of extracts

1. In the header of Dev Studio, click **Create** > **Technical** > **Activity**.
2. In the **New** tab, in the **Activity Record Configuration** section, specify the details of the activity, and then click **Create and open**.
   For more information, see Creating an activity.

Updating the activity responsible for scheduling of extracts

3. In the **Method** field, enter `Property-Set`.
4. Click **Expand to see method parameters**, and specify parameters that constitute the boundaries of the extracted data.

> **NOTE:**
>
> Use immutable data such as properties that are not updated over time. For example, use the *Create date/time* properties.
>
> Repeat the queueing process for each partition.

5. Click **Add a step**, and in the **Method** column, enter `Queue-For-Processing`.

©2025 Pegasystems Inc

6. Click the Expand to see method parameters, and perform the following actions:
   a. In the Type of queue list, select Dedicated.
   b. In the Queue processor field, enter the name of your custom queue processor, and then open it by clicking the Target icon.

      For example:

      Enter `PartitionedExtraction`.

   c. In the **Create Queue Processor** section, click Create and open.
7. In the **New** tab that appears, customize the created queue processor by performing the following actions:
   a. In the **Processing activity** section, in the Class field, specify the class to which the queue processor applies.
   b. In the Activity field, enter the label of the activity that is responsible for extracting the partitions, and then open it by clicking the Target icon.

      For example:

      Enter `ExtractPartition`.

   c. In the **Create Activity** section, click Create and open.

Customizing the activity responsible for extracting partitions

8. In the Method field, enter `Property-Set`.
9. Click Expand to see method parameters, and specify the details.

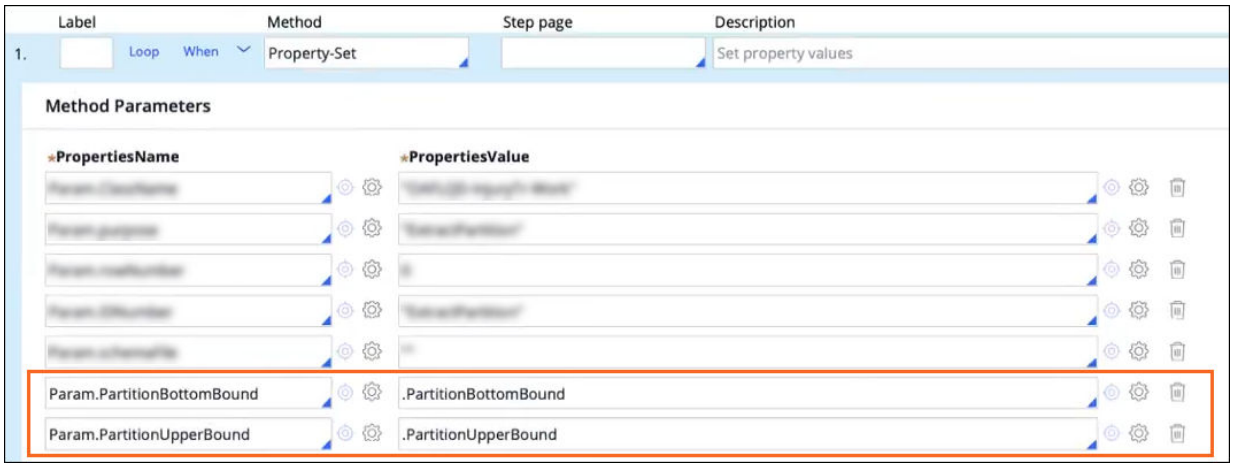
NOTE: Provide details of the Extract rule that you specified at the beginning. In this way, the system identifies the correct rule.

©2025 Pegasystems Inc

For example:

Enter parameters from your scheduled item to define boundaries of the partition.

The following figure shows sample parameters that might be set as boundaries:



Sample parameters

10. Click **Add a step**, and in the **Method** column, enter `Call Rule-Admin-Extract.runExtract`.
    Invoking this rule performs the extract.
11. In the upper-right corner of the rule, click **Save**.

What to do next:

Pega Platform enables you to define any number of partitions based on your specific requirements. If you want to extract data by using more partitions, add them to the activity responsible for scheduling of extracts.

Business Intelligence Exchange - FAQ

Business Intelligence Exchange (BIX) is a ruleset, which extraction processes can be used to extract Class data from your application. BIX provides the extract functions of an ETL (extract, transform, and load) utility by using the *Pega-BIX* ruleset, which supports the *Rule-Admin-Extract* rule type.

©2025 Pegasystems Inc

The following list includes frequently asked questions about BIX:

What is real-time data extraction in BIX?

Real-time data extraction is a capability that complements the existing scheduled batch extracts, enables clients to usge the same BIX extraction rules to events in real-time. Pega Platform™ provides the capability to publish change events along with data to a Kafka topic/queue for downstream processing, and also offers the ability to listen for events via an HTTP interface.

For more information, see Configuring BIX real-time data extraction.

Does Pega Platform™ support BIX?

Pega offers BIX as an optional add-on product, but the Ruleset is shipped with Pega Platform™. Contact your Pega Account Executive to ensure that your license includes BIX access. The real-time data extraction capability is available from Pega Infinity release '24.2.

What are the use cases for adopting real-time BIX capability?

The business need involves making data available in real-time for external Insights and analytics tools, or establishing dependencies between disparate systems.

Will the channels (API or Data Set) make a difference to the published changes?

The published data is driven by the Extract rule and formatted in JSON. There are no dependencies on the channels

Are all Data Set types supported?

Types in Data Set are categorized into Database Management, File System, General, Social, and Stream sources. In Pega Infinity release '24.2, the real-time data extraction supports Kafka Stream only.

PEGA ©2025 Pegasystems Inc

How to configure BIX real-time data extraction?

Configuration is done through the Extract Rule. Pega Platform™ is shipped with the *EnableDataExtraction* release toggle that needs to be enabled.

For more information, see Configuring BIX real-time data extraction.

How to restrict what events gets published to the consumer?

The events are captured by using the Declare Trigger. The instance/record commit to Database is watched for, and the ClipboardPage is queued for processing. The *pyPerformRealTimeDataExtraction* condition in the Declare Trigger is provided for extension.

What data gets published?

The properties mapped in the Extract Rule are extracted and published to Kafka. The message includes metadata to assist the consumers in processing the message.

Is data published only when it changes?

All database commit events are processed and published to Kafka. The messages include data, which consists of the properties and their values configured in the Extract rule, as well as metadata. The messages are published regardless of whether there are changes to the values of the properties configured in the Extract Rule.

What is BIX real-time data extraction performance impact?

The BIX real-time architecture uses three main components: Declare Trigger, Queue Processor, and Data Set (Kafka). The impact on end-users (on the web nodes) is negligible, because all performance operations are handled in the background or by stream nodes. These background or stream nodes will consume additional memory as messages are queued for processing.

PEGA ©2025 Pegasystems Inc

How to stop changes getting streamed to the Data Set?

You can disable the EnableDataExtraction release toggle, which prevents messages from being streamed to the configured Data Set.

Will traditional or batch-mode or flat-file approach of BIX be deprecated?

Batch Data Extraction is a required capability in the journey of real-time data extraction. The initial load of data or the reconciliation can be done using the batch data extractions.

Should I migrate from batch-mode BIX to real-time BIX?

Yes, begin adopting this capability in phases. This approach helps address issues with large volumes of data extraction by enabling continuous data extraction.

How does BIX real-time data extraction capability differ from Data Flows?

Data Flows read from a source and push to a target, hence they do not watch or monitor for events. In contrast, BIX monitors database commits, extracting and publishing information in real-time to downstream systems. BIX's Extract Rule offers in-model capability, allowing the configuration of properties needed for extraction. On the other hand, in Data Flows, extraction is configured using Data Transforms, which can complicate discoverability. This complexity can impact performance, especially when the data includes reference fields, as all reference fields must also be read from the source and processed.

What happens if the consumer or target is not available? Will the data be lost?

The extracted data is published to Kafka, where it is retained until the message is read or until the topic's message retention configuration is met. When the system is back online, the subscriber, consumer, or target can read the messages from Kafka.

PEGA ©2025 Pegasystems Inc

Can multiple real-time extracts be configured to provide different data from the same Case Type or Data Type to each consumer in real-time?

Only one real-time enabled Extract is allowed per Case Type or Data Type. Therefore, the Extract rule must include all the data required by all consumers. However, you can create and use multiple Extract rules for a Case Type or Data Type, as long as only one is real-time enabled.

How can the data be transformed before it is published?

It is currently not possible. The Extract Rule determines what data gets published, and the message is constructed based on this Rule.

How to use the Data Extraction APIs?

Use Postman to try the Data Extraction APIs.

Why are there no methods or APIs listed for Data Extraction on the Pega API landing page?

In the extract Service Package rule, update the Service Access Group. The configured Access Group must have the *Pega-BIX* ruleset in its stack or the application it refers to.

Do I need to write custom code or a service to read messages from Kafka and insert or update them into the data warehouse?

According to the requirements, custom code or a service should be written to read messages from Kafka and insert or update them into the data warehouse.

Is it possible to push data to a database table using real-time data extraction?

It is possible to send data to Kafka only. DataFlow or other custom codes can be implemented to read data from Kafka and push to the database table.

PEGA ©2025 Pegasystems Inc

How does real-time extraction support retries in the event of any failures?

Currently, there is limited support available for handling errors and exceptions. If failures occur while pushing data to Kafka, messages are sent to a Dead Letter Topic. This capability is being enhanced to support message retries from the Dead Letter Topic in upcoming releases.

How is data security ensured during transit in real-time data extraction?

Data is written to Kafka using the Dataset rule, and the security configuration for the Dataset is applied to the data transmitted to Kafka.

Is the Declare Trigger not created or deleted when the extract rule is saved?

The Declare Trigger is created when real-time data extraction is enabled. However, the Declare Trigger is not created or deleted when changes to the extract rule are saved. After saving, the changes need to be checked in to create or delete a Declare Trigger.

Service package: 'extract' does not list any Service APIs?

By default, service access group points to *PRPC:Administrators*. Update the service access group to the one with proper access group of application to view the Service APIs.

When data loss happens and data fails to reach external Kafka, is any exception or alert generated?

An exception will be logged in the event of any failures encountered while writing data to Kafka.

PEGA ©2025 Pegasystems Inc

Troubleshooting BIX for on-premises environments

When you run a Business Intelligence Exchange (BIX) extraction process from a Pega Platform Extract rule, a Pega Platform Job Scheduler rule, or the command-line, learn to recognize the most common errors in the extraction process in on-premises environments. Refer to the following tables to troubleshoot the root causes and resolve or prevent problems with extractions.

Common issues

When experiencing an issue with a BIX extraction process in an on-premises environment, verify that you have performed the following tasks:

- Verify that you have specified the configuration settings correctly. For more information, see Specifying configuration settings for BIX in an on-premises environment.
- Ensure that you properly configure the elements in `prbootstrap.xml` and `prconfig.xml` when you run a command-line extract. For more information, see Running a BIX command-line extraction in on-premises systems.

Do not use agents to automate the BIX extraction process. For Pega Platform 8.4 and later, Job Scheduler rules replace agents for automating BIX extraction processes. Clients with agents running BIX extractions should replace the agents with a Job Scheduler rule. For more information, see Replacing an legacy Rule-type-agent with a job scheduler Rule and Using the Job Scheduler rule to extract data with BIX.

You can find a record of errors in the BIX extraction processes in the BIX log files. For more information about BIX logging, see BIX logging. For information on common BIX errors, refer to the following sections in this article:

- Query timeout exception error
- Generated CSV extract or manifest file cannot be parsed
- BIX extraction process does not generate a manifest file containing data description language for SQL databases

PEGA ©2025 Pegasystems Inc

- Java SQLException for schema generation
- System time zone ignored

Query timeout exceptions

Symptom	Explanation	Solution
An extraction process fails with a query timeout exception error.	Query timeout exception errors can occur when the rule runs against a database table with a significantly larger amount of records than in the previous extract.	Increase the query timeout value in the dynamic system settings (DSS). • DSS name: `BIX/selectQueryTimeout` • Ruleset: *Pega-RULES* **NOTE:** Restarting Pega environment is not required. For more information about editing a DSS value, see Editing a dynamic system setting.

©2025 Pegasystems Inc

Generated CSV extract or manifest file cannot be parsed

Symptom	Explanation	Solution
When you run a command-line extraction, Pega Platform cannot parse a generated CSV extract or CSV manifest file through a downstream process, such as a file listener using a parse delimiter rule.	CSV parsing failures can occur when the parse delimiter in Pega Platform interprets the CSV extract and misinterprets the file usage of commas.	Add the `-l` parameter when running the command-line extraction. For more information, see the following topics. • BIX parameters and command-line arguments • Running a BIX command-line extraction in on-premises systems

BIX extraction process does not generate a manifest file containing data description language for SQL databases

Symptom	Explanation	Solution
An extraction process that uses database schema as the extract file format, that has enabled a manifest (DDL) file to extract to a target database, produces the following error in the log files: `Error: 2020-06-02 19:23:24,783 [ PegaRULES-Batch-4] [ STANDARD] [ ] [ CLM:01.01.27] (ernal.bix.BIXManifestWriterDDL ) ERROR G01292608`	The Extract rule for SQL database extractions uses dbo as the default schema name. Pega Platform does not provide the option to change the default schema name for SQL extractions in a target database. If your target SQL database schema does not use dbo as the schema name, Pega Platform produces this error when attempting to generate a manifest file in the schema.	In the `prconfig.xml` file default schema name fro which Pega Platform extr `<env name="database defaultSchema" valu` • Replace `<DBName>` in which you want P to generate the mar • Replace `<SchemaNa`

©2025 Pegasystems Inc

Symptom	Explanation	Solution
		to which Pega Platfo the manifest file in t For more information, se Configuring the target da on-premises systems.

Java SQLException for schema generation

Symptom	Explanation	Solution
Schema generation for the target database fails with a Java SQLException, and produces the following error in the log files: `Error: "java.sql.SqlException: A PRPC-supported database platform was not found:`	When using a database schema as an output, Pega Platform must support the database version that you are using.	Do not extract data to a database that Pega Platform does not support. For more information about supported databases, see the Pega Platform Support Guide.

System time zone ignored

Symptom	Explanation	Solution
When you run a command-line extraction or an extraction using the Job Scheduler rule, the BIX extraction process does not	The time zone information must also be passed with the parameters `-u <start date>`, `-U <end date>`, `-d <start date>` and `-D`	Include time zone information in parameters for `-u <start date>`, `-U <end date>`, `-d <start`

©2025 Pegasystems Inc

Symptom	Explanation	Solution
consider the time zone when you use the following parameters: `-u <start date>`, `-U <end date>`, `-d <start date>` and `-D <end date>`.	`<end date>`. The time zone must reflect the thread from which the extraction process runs.	`date>` and `-D <end date>`. For more information, see BIX parameters and command-line arguments

Specifying configuration settings for BIX in an on-premises environment

Ensure that you have specified the following BIX configuration settings to your meet your extraction process requirements:

Location	Setting name	Description
`prconfig`	`Compatibility/BIXUseOptimizedClipboardXML`	Optimizes the Clipbo large number of Clas For more informatior prconfig.xml settings
`prconfig`	`Compatibility/BIXdisableForwardChaining`	Disables forward-cha properties when loac extraction process. For more informatior prconfig.xml settings
`prconfig`	`Compatibility/BIXDisableBackwardChaining`	Disables backward-ch properties when loac extract.

©2025 Pegasystems Inc

Location	Setting name	Description
		For more informatio prconfig.xml settings
prconfig	bix/retainOriginalFileNameForCSV	Keeps the original fil The default value (fa ') in the original file n
prconfig	bix/removeAdditionalColumnsInCSV	Removes the specifie files. For example, <env r removeAdditional value="pxExtract teTime"↗
prconfig	bix/useHighThroughputDownloadForCSV	Enables high-through extracts from a BLOB Postgres database fo
prconfig	bix/getClearTextWhileExtraction Ruleset: *Pega-RULES*	Displays encrypted d BLOB as clear text wh The default value (fa keeps data encrypted
DSS	BIX/generateXSDForAllProperties Ruleset: *Pega-RULES*	Generates XSD when for the XML type outp (false) disallows the GetAllProperties.

©2025 Pegasystems Inc

Location	Setting name	Description
DSS	`BIX/selectQueryTimeout` Ruleset: *Pega-RULES*	Specifies the time in s fetch records from th out.
DSS	`BIX/setCommandLineDelimiterForCSVManifestFile` Ruleset: *Pega-RULES*	By default, the comm applicable to the CSV default value of this D default setting and er Manifest as well. For more informatior • BIX parameters arguments • Running a BIX c on-premises sys • Using the Job Sc with BIX

- Configuring an FTP listener to access BIX data extracts in Pega Cloud environments
- BIX parameters and command-line arguments

- Using the Job Scheduler rule to extract data with BIX
- Running a BIX command-line extraction in on-premises systems

©2025 Pegasystems Inc