Search documentation

Appian

Tempo Report Design

Appian 7.7

Toggle Us

Tempo reports are a way of pulling in data from tasks, records, relational databases, and other third-party data sources and displaying it on a single SAIL in Tempo for end users to view. Through the use of rules, SAIL components and expressions, Tempo reports enable designers to quickly define what data to display and how to display it in a usable and concise way on both web and mobile clients.

The sections below detail design information for Tempo reports, such as what to expect in terms of report security, the options available when configuring a Tempo report, and how to manage them after their creation.

To walk through an example of creating your first Tempo report, see also: Grid Tutorial

Tu

Re

Frequently Asked Questions

Where do Tempo reports display?

Tempo reports display in the Work Platform interface. By default, they display on the Reports tab. If configured as a task report, they display on the Tasks tab.

How do Tempo reports differ from Portal reports?

Tempo reports can easily be modified through their expression, be viewed on mobile devices and the broad set of web browsers supported by Tempo, and report on data from tasks, records, and relational databases. Portal reports can only be viewed from the Portal and can only report on data from process models, processes, and tasks.

Ot

What happens to my Portal reports?

These continue to work as before in the legacy Portal interface as part of Appian 7.x. As you migrate your application to Tempo, you can design Tem reports for each Portal report, including Task Reports, and access these from the Tempo interface.

Can I Import/Export Tempo reports?

Yes, Tempo reports are like other Appian objects in that their definitions can be deployed between development and production systems using application containers.

Do I have to use Tempo reports when I upgrade?

No, you can update to the latest version of Appian without any required re-work. You can begin using Tempo reports when you choose.

Can I have multiple components on a Tempo report?

Yes, the report's expression displays arrays of components in a single-column layout, or you can display arrays within specific layout components, such as DashboardLayout, to display more than just a single chart or grid.

Tempo Report Concepts

Tempo Report

A Tempo report is an Appian object that displays charts and grids based on an expression. It holds the basic properties of the report (such as name and description), the expression that determines what SAIL components display, and the role map that determines its security.

Tempo reports can display data queried from tasks, records, and relational databases.

See also: Record Design and Records Tutorial

You can display the data through any of the SAIL components, including chart components and basic field components.

See also: SAIL Components

Task Report

Task reports are a subset of Tempo reports. They are accessible from the Tasks tab alongside the system's existing task views. You can configure any Tempo report to be a task report while creating the Tempo report object, but Appian recommends only configuring those reports that display a grid wi process task links.

See also: Grid Field and Process Task Links

Security Model

Security for Tempo reports can be defined at two levels. Access to the data displayed, and access to the report in Tempo.

Data Security

Data security is based on the security of the underlying data source. The expression runs under the context of the user viewing the report. Users must have at least viewer rights to the data to view it on the report, but Tempo report security alone will not stop them from viewing the data in other areas the system.

For example, not giving a user viewer rights to a Tempo report prevents the user from seeing it on the Reports tab in Tempo, but the data might still b visible in a Portal report or process dashboard. On the other hand, if a user has viewer rights to a Tempo report, but they do not have access to the da used in a component on the report, the user will see the report in the list, but the report may fail to render or just the component with the restricted da will fail to render. Again, this depends on the expression and the source security.

See also: Configuring Process Security and Configuring Data Store Security

Tempo Report Security

Each Tempo report has a rolemap specifying its viewers, auditors, editors, and administrators. When creating the Tempo report, only one group can be added to each role. Users must also have at least viewer rights to the different pieces of data to view it on the report.

The rights for each role include the following:

Actions / Roles	Administrator	Editor	Auditor	Viewer
View in Tempo	Yes	Yes	Yes	Yes
View the Tempo report configuration	Yes	Yes	Yes	No
Update properties	Yes	Yes	No	No
Update security	Yes	No	No	No
Delete the Tempo report	Yes	No	No	No

To access the Tempo report object's definition and rolemap, users must also be in the designer role.

Notes

When importing a report into an environment where the report already exists (e.g. to update it), the imported report's rolemaps will be merged with the existing report's rolemaps. If the imported report has different rolemaps from the existing report, you will need to reconfigure the security on the environment after the import using the report designer interface. The report designer interface only displays one group per role, so it may not accurate reflect the rolemaps after import if the import resulted in multiple groups for a single role. Reconfiguring the security for all of the roles will ensure that each role has the correct rolemap set and displayed. This behavior may change in a future release.

See also: Designer Role

Creating a Tempo Report

Tempo reports can be created by any user in the designer role.

Creating a report with a single chart SAIL component involves four main steps:

1. **Create a rule that returns the values to display.** Done by creating a rule that results in a `series` value for the chart SAIL component.
2. **Configure the SAIL dashboard.** Done by using SAIL functions to return component values that will display on the Tempo report.
3. **Create the security groups.** Done as you would create any other group in Appian and allows you to easily change the viewer, auditor, editor, and administrator users on the report.
4. **Create the Tempo report.** Done through the Reports view in the Designer interface. Once complete, the Tempo report displays in Tempo.

To create a Tempo report that contains more than one chart component, as well as any of the other SAIL components, follow the best practices for creating a SAIL interface. Your expression can then be used in place of the single chart component detailed in step 2 to display a more robust Tempo report.

See also: SAIL Tutorial

Create a Rule that Returns the Values to Display

To create a Tempo report that displays a chart SAIL component, you need to first create a rule that returns the data values and related labels to display in your chart.

The returned value must be of either type BarChartSeries, ColumnChartSeries, LineChartSeries, or PieChartSeries depending on the chart. You can use the `a!chartSeries()` function to return a value of this type.

See also: SAIL Chart Components

Creating a query rule for this function can be done in one of three ways depending on your data source:

- For record types, you can use the `queryrecord()` function to return a value of type DataSubset, then returns a value in the `series` syntax. See also: queryrecord()
- For data stores, you can use the `a!queryEntity()` function to return a value of type DataSubset, then returns a value in the `series` syntax. See also: a!queryEntity()
- For data stores or variables, you can create a query rule that directly queries the data, then use that return value within an expression rule to format the data into the `series` syntax.

See the following SAIL Recipes for examples of creating charts using `a!queryEntity()` :

- Aggregate Data from a Data Store Entity and Display in a Chart: Walks you through how to aggregate data and display it in a pie chart.
- Aggregate Data from a Data Store Entity using a Filter and Display in a Chart: Walks you through how to aggregate data from a data store entity and display it in a bar chart.
- Aggregate Data from a Data Store Entity by Multiple Fields and Display in a Chart: Walks you through how to aggregate data from a data store entity by multiple fields and display it in a column chart with multiple series.
- Aggregate Data from a Data Store Entity and Conditionally Display in a Chart or Grid: Walks you through how to aggregate data and display it in either a pie chart or a grid based on the user's preference.
- Configure a Chart Drilldown to a Grid: Walks you through creating a pie chart with dynamic links that will display a grid with data that depends on which pie slice the user clicks.
- Filter the Data in a Grid Using a Chart: Walks you through creating a pie chart with dynamic links that will filter the data in a grid depending on which pie slice the user clicks.

Configure the SAIL Dashboard

To display your data on a Tempo report, you need to create an expression that calls the rule you created in the previous step.

For a report that displays a single component, your expression only needs to return a single SAIL component data type value. To do so, use the SAIL functions provided with each SAIL component.

See also: SAIL Components

For example, an expression that returns a PieChart value may resemble the following where `rule!chartData` is the rule you created in the previous step:

```
=a!pieChart(
  label: "Opportunity Sources",
  instructions: "Leads are calculated based on the initial mode of contact",
  series: rule!chartData,
  showDataLabels: true,
)
```

Save your expression as an interface rule to make it easy to modify later. This rule will be used as the SAIL Dashboard value when creating the Tempo report object described later.

See also: Creating an Interface and SAIL Tutorial

Configure the Security Groups

As discussed in the Security Model section, only one group can be added to each security role for your Tempo reports rolemap.

Appian recommends creating groups specifically for the Tempo report.

Create a group for the administrators and add users or groups to it as needed.

Optionally, you can create a group for each of the following and add users or groups to them as needed:

- Viewers
- Auditors
- Editors

After creating the Tempo report, you can always add or remove users and groups from your Tempo report security groups without needing to modify the report directly.

Create the Tempo Report Object

Once you have configured the rules to determine what data to display and how it should display, as well as the security groups that make up the Tempo report's rolemap, you can create your Tempo report object.

Once created, it will be listed on the Reports tab and users within the security groups created in the previous step can view it.

To create the Tempo report object, access the Reports view in the Designer interface, and click **Create a Tempo Report** from the left-hand toolbar.

Fields

Name: The name of the Tempo report. Only accepts a Text value. For example, Yearly Earnings, Tickets by Status, or Tickets by Priority.

Description (Optional): The description of the Tempo report to be displayed in the Reports list view. Only accepts a Text value. For example, Summary Yearly Earnings Broken Down by Quarter, Number of Tickets Broken Down by Status, or Number of Tickets Broken Down by Priority.

Save as Task Report (Optional): Enables the Tempo report as a task report.

Viewer Group (Optional): A single group whose members have viewer rights to the report.

Auditor Group (Optional): A single group whose members have auditor rights to the report.

Editor Group (Optional): A single group whose members have editor rights to the report.

Administrator Group: A single group whose members have administrator rights to the report.

SAIL Dashboard: Expression that calls the rule you created for the interface earlier. See above: Configure the SAIL Dashboard

Notes

The system automatically generates a URL stub when you save the Tempo report object. For example, "lstM7Q".

Once the Tempo report is created, the report is available for all users with at least viewing rights. The URL stub created for the report can be shared between users but will only work for those that can already view the report.

Editing Tempo Reports

The following basic properties of a Tempo report can be edited:

- Name
- Description
- Viewer Group
- Auditor Group
- Editor Group
- Administrator Group
- SAIL Dashboard

To edit these basic properties, complete the following:

1. Open the **Reports** view in the Designer interface.
2. From the toolbar on the left-hand side, select **View All Tempo Reports**.
   - A list of existing Tempo reports displays.
3. Select the Tempo report you want to edit.
   - The Edit Report window displays.
4. Modify the values as needed.
5. Click **Update Report**.

You can also edit the rules associated with the SAIL Dashboard value or add and remove users or groups from the report's security groups.

See also: Editing Rules and Adding Users to Groups

To change whether or not a Tempo report is a task report, you must create a new report.

Version Control

Only the latest version of the Tempo report is saved in the system.

Any changes you make to the rules created as part of the SAIL Dashboard, however, are saved as a new version just as any other rule.

See also: Version Control for Rules

Deleting Tempo Reports

Deleting a Tempo report removes the object from the system. It will no longer appear in Tempo.

To delete a Tempo report, complete the following:

1. Open the **Reports** view in the Designer interface.
2. From the toolbar on the left-hand side, select **View All Tempo Reports**.
   - A list of existing Tempo reports displays.
3. Click the orange **X** next to the Tempo report you want to delete.
   - A confirmation window displays.
4. Click **Yes**.

Tempo Report Design Best Practices

Report Definition

Save Report Configuration Expressions as Rules

Appian recommends saving any expressions created for a report as a rule for version control and testing purposes.

This also allows you to easily copy parts or all of the rule to create similar reports for different data.

Keep Report Names Simple and Concise

Each Tempo report's name should be easy for users to scan through.

Report Performance

Limit the Number of Series Values and Categories

The number of series values for a chart component affects performance. The fewer values that need to be evaluated, the better the performance.

Limit the Number of Chart SAIL Components on the UI

Each SAIL component added to a Tempo report also impacts the report's performance. This especially applies to chart and grid components. The more complex each component, the higher the strain on performance. Keep this in mind when adding more than one complex component to your report.

Limit Queries to External Databases

Store external data or the result of expressions that do heavy data manipulations in local variables defined using load() to avoid re-executing them upon every reevaluation. Use local variables to avoid redundant queries when configuring one or more components that use the same data.

Report Usability

Review and Adjust the Chart

As discussed, many factors affect the size, layout, and performance of a chart. Be sure to review the chart as it displays in Tempo and apply the best practices above to tweak the end result. Modifying your rule for the SAIL Dashboard in one browser while viewing the Tempo report in another is a great way to see the impact of your changes easily.

Keep Report Names Concise and Descriptive

Users should easily be able to understand the purpose and content of each Tempo report from its name. Since reports are sorted alphabetically, related reports should use names that start with the same text (such as the name of the record type being reported on) so they are grouped together on the report list, e.g. "Customers by Industry" and "Customers by Region".

Use Appropriate Column Layouts for Report Content

Two-column layouts provide greater density for smaller charts and grids, allowing users to see more data without scrolling and facilitating side-by-side comparisons. One-column layouts provide additional space to show charts and grids that contain more data points. A one-column layout is recommended for grids that include more than 5 columns and/or lengthy text content. Charts that show more than 7 data points are generally best-shown in one-column layouts.

To create visually-balanced reports, it is recommended to use the same layout (one-column or two-column) for all report sections. When mixing different layouts within a report, make sure that the height of content in each column of two-column sections is similar in order to minimize white space. For the same reason, it is preferable to place a one-column section above a two-column section; this reduces the likelihood of a shorter column creating empty space above the start of the next section. An example of an effective mixed-layout report is a grid in a one-column section above a two-column section containing a variety of smaller charts.

Keep Chart Labels Short

Long label text may reduce the amount of space available for the plot area of charts. When there is limited room to show chart labels, some text may overlap and certain labels may be hidden to preserve space.

Minimize the Number of Chart Series and Categories

Design charts with as few dimensions and data points as needed to optimize display. Simpler charts are easier to comprehend and also load more quickly

Line charts or column charts with a large number of data points will require users to scroll horizontally to view all the data.

Pie charts with a large number of slices, particularly if some of the slices are very thin (values with small magnitude compared to the overall data represented), may display with some data labels not visible. If this cannot be avoided, enable tooltips so users can see data values by hovering over pie slices.

Limit the Number of Grid Columns and Keep Text Short

Grids with a large number of columns, long column header labels, and/or long text data values will require users to scroll horizontally to view all columns. Use a one-column layout to allow more grid columns to be seen without scrolling. Consider reformatting or abbreviating label and data values to reduce their character length.

Accessible Reports Must Provide a Plain-Text Representation of Data

While charts are an effective medium for visually presenting data, they are primarily intended for consumption by sighted users. To make the same data available to non-sighted users who interact with Appian through a screen reader, provide a grid view of report content.

See Also

Grid Tutorial: Walks you through an example of creating your first grid and then configuring different ways to manipulate the data and enable paging an sorting

SAIL Tutorial: Walks you through an example of creating your first UI, which is the basis of Tempo report.

SAIL Components: Lists the supported SAIL components and the data structure required for adding them to a Tempo report.

SAIL Recipes: Includes several examples of how to create charts.