Traces

- Add a panel with tracing visualizations
  - Before you begin
  - Steps {#add-the-traces-panel-query}
- Add TraceQL with table visualizations
  - Use a variable to add other links to traces
- Configuration options
  - Panel options
  - Span filters options

{{< shared id="traces-viz" >}}

Traces visualizations let you follow a request as it traverses the services in your infrastructure. The traces visualization displays traces data in a diagram that allows you to easily interpret it. Traces visualizations currently render one trace traversal based on the traceID used in TraceQL or using a variable.

{{< /shared >}}

For more information about traces and how to use them, refer to the following documentation:

- Tracing in Explore
- Tempo data source
- Getting started with Tempo

{{< figure src="/media/docs/grafana/panels-visualizations/screenshot-traces-visualization-11.5.png" max-width= "700px" alt="Screenshot of the trace view" >}}

{{< docs/play title="Traces Panel" url="https://play.grafana.org/d/edodkfmj0tce8f/" >}}

Add a panel with tracing visualizations

After you have tracing data available in your Grafana instance, you can add tracing panels to your Grafana dashboards.

Using a dashboard variable, `traceID`, lets you create a query to show specific traces for a given trace ID. For more information about dashboard variables, refer to the Variables documentation.

Before you begin

To use this procedure, you need:

- A Grafana instance
- A Tempo data source connected to your Grafana instance

Steps {#add-the-traces-panel-query}

To view and analyze traces data in a dashboard, you need to add the traces visualization to your dashboard and define a query using the panel editor. The query determines the data that is displayed in the visualization. For more information on the panel editor, refer to the Panel editor documentation.

This procedure uses dashboard variables and templates to allow you to enter trace IDs which can then be visualized. You'll use a variable called `traceId` and add it as a template query.

1. From your Grafana instance, do one of the following:
   - On a new dashboard: Click or drag a panel onto the dashboard.
   - On an existing dashboard: Click **Edit** in the top-right corner, click the **Add new element** icon, and then click or drag a panel onto the dashboard.
2. Click **Configure visualization** to open panel edit mode.
3. In the query editor, click the data source list and select the appropriate tracing data source.
4. In the top-right corner of the panel editor, select the **All visualizations** tab, search for, and select **Traces**.
5. Under the **Panel options**, enter a **Title** for your trace panel or have Grafana create one using generative AI features.

For more information on the panel editor, refer to the Configure panel options documentation.

6. In the query editor, click the **TraceQL** query type tab.
7. Enter `${traceId}` in the TraceQL query field to create a dashboard variable. This variable is used as the template query.
   {{< figure src="/static/img/docs/panels/traces/screenshot-traces-template-query.png" alt="Add a template query" >}}
8. Click **Back**.
9. Click the **Add new element** icon and click **Variable**.
10. Add a new variable called `traceId`, of variable type **Custom**, giving it a label if required.
    {{< figure src="/media/docs/grafana/panels-visualizations/screenshot-traces-custom-variable-v11.5.png" max-width="400px" alt="Add a template query" >}}
11. Click **Save**.
12. Enter an optional description of your changes, and click **Save**.
13. Click **Exit edit**.
14. Verify that the panel works by using a valid trace ID for the data source used for the trace panel and editing the ID in the dashboard variable.
    {{< figure src="/static/img/docs/panels/traces/screenshot-traces-traceid-panel.png" alt="Results of query in trace panel" >}}

Add TraceQL with table visualizations

While you can add a trace visualization to a dashboard, having to manually add trace IDs as a dashboard variable is cumbersome. It's more useful to instead be able to use TraceQL queries to search for specific types of traces and then select appropriate traces from matching results.

1. In the same dashboard where you added the trace visualization, click **Edit** in the top-right corner.
2. Click the **Add new element** icon, and then click or drag a panel onto the dashboard.
3. Click **Configure visualization**.
4. Select the same trace data source you used in the previous task.
5. In the top-right corner of the panel editor, select the **All visualizations** tab, search for, and select **Table**.
6. In the query editor, select the **TraceQL** tab.
7. Under the **Panel options**, enter a **Title** for your trace panel or have Grafana create one using generative AI features.
8. Add an appropriate TraceQL query to search for traces that you would like to visualize in the dashboard. This example uses a simple, static query. You can write the TraceQL query as a template query to take advantage of other dashboard variables, if they exist. This lets you create dynamic queries based on these variables.

{{< figure src="/static/img/docs/panels/traces/screenshot-traces-dynamic-query.png" alt="Create a dynamic query" >}}
9. Click **Save**.
10. Enter an optional description of your changes, and click **Save**.
11. Click **Back** and **Exit edit**.

When results are returned from a query, the results are rendered in the panel's table.

{{< figure src="/static/img/docs/panels/traces/screenshot-traces-returned-query.png" alt="Results of a returned query in the panel table" >}}

Use a variable to add other links to traces

The results in the traces visualization include links to the **Explore** page that renders the trace. You can add other links to traces in the table that fill in the `traceId` dashboard variable when selected, so that the trace is visualized in the same dashboard.

To create a set of data links in the panel, use the following steps:

1. In the panel editor menu, under **Data links**, click **Add link**.
2. Add a **Title** for the data link.
3. Find the UUID of the dashboard by looking in your browser's address bar when the full dashboard is being rendered. Because this is a link to a dashboard in the same Grafana instance, only the path of the dashboard is required.
   {{< figure src="/static/img/docs/panels/traces/screensnot-traces-uuid-url.png" alt="Unique identifier for the dashboard" >}}
4. In the **URL** field, make a self-reference to the dashboard that contains both of the panels. This self-reference uses the value of the selected trace in the table to fill in the dashboard variable. Use the path for the dashboard from the previous step and then fill in the value of `traceId` using the selected results from the TraceQL table. The trace ID is exposed using the `traceID` data field in the returned results, so use that as the value for the dashboard variable.
   {{< figure src="/static/img/docs/panels/traces/screenshot-traces-edit-link.png" alt="Edit link and add the Trace link" >}}
5. Select **Save** to save the data link.
6. Enter an optional description of your changes, and click **Save**.
7. Click **Back** and **Exit edit**.

You should now see a list of matching traces in the table visualization. While selecting the **TraceID** or **SpanID** fields will give you the option to either open the **Explore** page to visualize the trace or following the data link, selecting any other field (such as **Start time**, **Name** or **Duration**) automatically follows the data link, filling in the `traceId` dashboard variable, and then shows the relevant trace in the trace panel.

{{< figure src="/static/img/docs/panels/traces/screenshot-traces-trace-link.png" alt="Selecting the trace link" >}}

{{< figure src="/static/img/docs/panels/traces/screenshot-traces-trace-link-follow.png" caption="Follow the trace link populates the trace ID and displays the traces view" >}}

Configuration options

{{< docs/shared lookup="visualizations/config-options-intro.md" source="grafana" version="<GRAFANA_VERSION>" >}}

Panel options

{{< docs/shared lookup="visualizations/panel-options.md" source="grafana" version="<GRAFANA_VERSION>" >}}

Span filters options

The **Span filters** options control the initial state of the span filters when the visualization loads, allowing you to customize your trace analysis view.

Option	Description
Filters	Add free-form filters that set the initial span filter state. Use **Text search** for text queries, `duration` for duration filters, or span attributes such as `service.name` and `span.name`. Supports variable interpolation. For example, you can set a filter value to `$var`, and the visualization replaces it with the value for the dashboard variable named `$var`.
Show matches only	Toggle the switch on to display only spans that match the defined filter criteria. This helps simplify trace interpretation.
Select critical path	Toggle the switch on to highlight spans in the critical path, which helps identify performance bottlenecks and their impact on overall latency.

Duration filters use the `duration` key with operators such as `=`, `>=`, `<=`, `>`, and `<`. In the visualization, users can toggle **Show all spans** to switch from the filtered view back to all spans.

□□□ #2
□ □□□ □□□ 2026-10-31 16:39:31 UTC
□ □□□ □□□ 2026-10-31 16:39:33 UTC