InfluxDB annotations

- Before you begin
- Create an annotation query
- InfluxQL annotations
  - Field mappings
  - InfluxQL annotation query example
  - Range annotation example
- Flux annotations
  - Flux annotation query example
- SQL annotations
  - SQL annotation query example
  - SQL range annotation example
- Use template variables in annotations

Annotations overlay event data on your dashboard graphs, helping you correlate events with metrics. You can use InfluxDB as a data source for annotations to display events such as deployments, alerts, or other significant occurrences on your visualizations.

For general information about annotations, refer to Annotate visualizations.

Before you begin

Before creating InfluxDB annotations, ensure you have:

- An InfluxDB data source configured in Grafana
- Data in InfluxDB containing event information with timestamp fields
- Read access to the InfluxDB database or bucket containing your events

Create an annotation query

To add an InfluxDB annotation to your dashboard:

1. Navigate to the dashboard you want to update and click **Edit**.
2. Click the **Add new element** icon (blue plus sign).
3. Click **Annotation query**.
4. Enter a name for the annotation query.
5. If you don't want to use the annotation query right away, clear the **Enabled** checkbox.
6. Select a color for the annotation event markers.
7. Select an option in the **Show annotation controls in** drop-down list to control where on the dashboard the annotation is displayed.
8. Select an option in the **Show in** drop-down list to control the panels in which the annotation is displayed.
9. Click **Open query editor** to open the **Annotation Query** dialog box.
10. Select your **InfluxDB** data source from the **Data source** drop-down list.
11. Configure the annotation query and field mappings.
12. (Optional) Click **Test annotation query** to ensure that the query is working properly.
13. Click **Close** when you've completed the query setup.
14. Click **Save**.
15. (Optional) Enter a description of the changes you've made.
16. Click **Save**.
17. Click **Exit edit**.

InfluxQL annotations

For InfluxQL-configured data sources, write an InfluxQL query in the **InfluxQL Query** field. Your query **must** include `WHERE $timeFilter` to scope the results to the dashboard's time range.

Field mappings

Field mappings tell Grafana which InfluxDB columns contain the annotation data. If your query returns only one column, you don't need to enter anything in the field mapping fields.

Field	Description
Text	The column containing the annotation description displayed when you hover over the annotation.
Tags	The column containing tags for the annotation. The value can be a comma-separated string. Tags help categorize and filter annotations.
TimeEnd	The column containing an end time for range annotations. Range annotations display as a shaded region on the graph instead of a single vertical line.

InfluxQL annotation query example

The following query retrieves deployment events and displays them as annotations:

```
SELECT title, description
FROM events
WHERE $timeFilter
ORDER BY time ASC
```

Range annotation example

To display events with duration as shaded regions:

```
SELECT description, tags, end_time
FROM maintenance_windows
WHERE $timeFilter
ORDER BY time ASC
```

Configure the field mappings as follows:

- **Text:** `description`
- **Tags:** `tags`
- **TimeEnd:** `end_time`

Flux annotations

For Flux-configured data sources, annotations use the standard Flux query editor. Write a Flux query that returns data frames with time, text, and optional tag fields.

Flux annotation query example

```
from(bucket: "events")
  |> range(start: v.timeRangeStart, stop: v.timeRangeStop)
  |> filter(fn: (r) => r["_measurement"] == "deployments")
  |> pivot(rowKey: ["_time"], columnKey: ["_field"], valueColumn: "_value")
```

Grafana processes the resulting data frames as annotation events. Ensure the query returns a `_time` column and at least one text column.

SQL annotations

SQL-configured data sources (InfluxDB 3.x) use the standard Grafana annotation query editor. Write an SQL query that returns a time column and at least one text column. Use the `$__timeFilter(time)` macro to scope results to the dashboard time range.

SQL annotation query example

```
SELECT time, title, description
FROM events
WHERE $__timeFilter(time)
ORDER BY time ASC
```

SQL range annotation example

To display events with duration as shaded regions, return both a start time and an end time:

```
SELECT time, end_time, description, tags
FROM maintenance_windows
WHERE $__timeFilter(time)
ORDER BY time ASC
```

Configure the field mappings:

- **Text:** `description`
- **Tags:** `tags`
- **TimeEnd:** `end_time`

Use template variables in annotations

You can use template variables in your annotation queries to filter annotations based on dashboard variable selections. For example:

```
SELECT title, description
FROM events
WHERE $timeFilter AND environment = '$environment'
ORDER BY time ASC
```

If your annotations aren't appearing or you encounter errors, refer to Troubleshoot InfluxDB data source issues.

□□□ #2
□ □□□ □□□ 2026-10-31 15:57:23 UTC
□ □□□ □□□ 2026-10-31 15:57:25 UTC