Graphite annotations

- Common use cases
- Before you begin
- Annotation query modes
- Create an annotation query
- Metric query annotations
  - Example metric queries
    - Deployment markers
    - Service restarts
    - Error spikes above a threshold
    - Host down
- Events query annotations
  - Create events in Graphite
  - Example events queries
  - Correlate multiple event types
- Use template variables in annotations
- End-to-end example: correlate deployments with latency
  - Before you start
  - Step 1: Record a deployment event
  - Step 2: Build the latency panel
  - Step 3: Add the deployment annotation query
  - Step 4: Interpret the result
- Troubleshoot annotation issues
  - Annotations don't appear
  - Annotation title is empty

Annotations overlay event data on your dashboard graphs, helping you correlate events with metrics. You can use Graphite as a data source for annotations to display events such as deployments, incidents, or maintenance windows on your visualizations.

For general information about annotations in Grafana, refer to Annotate visualizations.

Common use cases

Graphite annotations help you explain changes in your metrics by marking when related events happened. Common use cases include:

- **Deployments:** Mark each release so you can see whether a deployment changed latency, error rate, or throughput.
- **Incidents and outages:** Mark when incidents start and resolve to correlate them with metric anomalies.
- **Maintenance windows:** Mark planned maintenance so expected dips aren't mistaken for problems.
- **Configuration changes:** Mark when you change a feature flag or setting to track its effect over time.
- **Scaling events:** Mark when you add or remove capacity to explain shifts in load or performance.

Each use case maps to one of the two query modes described in this page: a metric query when an existing metric already marks the event, or an events query when you record discrete events in Graphite.

Before you begin

Before creating Graphite annotations, ensure you have:

- A configured Graphite data source in Grafana.
- Metrics or events in Graphite that represent the events you want to annotate.
- Read access to the Graphite instance.

Annotation query modes

Graphite supports two ways to query annotations. You configure both in the annotation query editor:

Mode	Field	Description

Metric query	**Graphite Query**	Runs a standard Graphite metric query. Each returned data point with a non-zero value becomes an annotation, and the series name becomes the title.
Events query	**Graphite events tags**	Queries the Graphite events API and filters by one or more event tags. Each matching event becomes an annotation.

Use the metric query mode when a metric is non-zero only at the moments you want to mark. A counter that stays above zero after the first event marks every interval, because Grafana creates an annotation for every non-zero point. Use the events query mode when you record discrete events in Graphite and tag them.

Create an annotation query

To add a Graphite 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 the **Graphite** data source from the **Data source** drop-down list.
11. Configure one of the annotation query modes:
    - In the **Graphite Query** field, enter a metric query, for example `statsd.application.counters.*.count`.
    - In the **Graphite events tags** field, enter one or more event tags, for example `deploy`.
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**.

{{< admonition type="note" >}} Enter a value in either the **Graphite Query** field or the **Graphite events tags** field, not both. If you provide a metric query, Grafana uses the metric query mode and ignores the tags. {{< /admonition >}}

Metric query annotations

In metric query mode, Grafana runs the value in the **Graphite Query** field as a standard Graphite query over the dashboard time range. Grafana converts the results into annotations using these rules:

- Every data point with a non-zero value creates an annotation. Data points with a value of zero or `null` are skipped.
- The annotation timestamp is the timestamp of the data point.
- The annotation title is the series name returned by Graphite.

Because every non-zero data point creates an annotation, use a metric that only produces values at the moments you want annotated. Metrics that report continuous values produce an annotation at every step interval and flood your dashboard.

Example metric queries

The following examples show common ways to turn a metric into annotations.

Deployment markers

Record deployments as Graphite events and query them by tag. Refer to Events query annotations.

If you only have a counter, query the increase between points. `nonNegativeDerivative()` is zero while the counter is flat and non-zero only when it goes up, so annotations appear at each deploy:

```
nonNegativeDerivative(deploys.myservice.count)
```

Service restarts

Use the same pattern for restarts. Query a metric that is non-zero only when a restart happens, or record restarts as events:

```
nonNegativeDerivative(stats.myservice.restart.count)
```

Error spikes above a threshold

Because Grafana skips zero and `null` data points, use `removeBelowValue()` to annotate only the moments a metric crosses a threshold. The following query annotates every interval where the error count is 100 or greater:

```
removeBelowValue(stats.myservice.errors.count, 100)
```

Values below 100 become `null` and are skipped, so annotations appear only during error spikes.

Host down

Annotate while a host is down by inverting an up metric so that a down state produces a non-zero value:

```
offset(scale(hosts.web01.up, -1), 1)
```

This query returns `1` when `up` is `0`. Grafana skips the zero points from when the host is up, and creates an annotation for each interval the host stays down.

Events query annotations

In events query mode, Grafana queries the Graphite events API and filters events by the tags you provide in the **Graphite events tags** field. Grafana converts each matching event into an annotation using these rules:

- The annotation timestamp is the event's `when` value.
- The annotation title is the event's `what` value.
- The tags you query with are attached to the annotation.

Enter multiple tags separated by a space. Grafana sends those tags to the Graphite events API without `set=union`, so an event must include every tag.

You can filter events in the following ways:

Value	Result
Empty	Returns all events in the dashboard time range.
A single tag	Returns events that include that tag, for example `deploy`.
Multiple tags	Returns events that include every tag. `deploy myservice` matches events tagged with both.
A wildcard	Returns events whose tags match the pattern, for example `web*`.

Create events in Graphite

Before you can query events, record them in Graphite. Graphite exposes an HTTP endpoint for creating events:

```
curl -X POST http://<GRAPHITE_HOST>/events/ \
  -H "Content-Type: application/json" \
  -d '{
    "what": "Deployed myservice v2.5.0",
    "tags": ["deploy", "myservice"],
    "data": "Deployed new feature flags and performance improvements"
  }'
```

Replace `<GRAPHITE_HOST>` with your Graphite server address. The `what` field becomes the annotation title, and the `tags` field is what you filter on in the **Graphite events tags** field.

Example events queries

The following examples show what to enter in the **Graphite events tags** field for common scenarios.

Goal	Value	Result
Show all events	Empty	Displays every event in the dashboard time range.
Show all deployments	`deploy`	Displays events tagged `deploy`.
Deployments for one service	`deploy myservice`	Displays events tagged with both `deploy` and `myservice`.
All production events	`prod*`	Displays events whose tags match the `prod*` pattern.
Incidents	`incident`	Displays events tagged `incident`.

Correlate multiple event types

To compare different kinds of events on the same dashboard, add more than one annotation query and give each a distinct color. For example:

1. Add a **Deployments** annotation query with the tag `deploy`.
2. Add an **Incidents** annotation query with the tag `incident`.
3. Add a **Maintenance** annotation query with the tag `maintenance`.

Each query renders in its own color, so you can see how deployments, incidents, and maintenance windows line up with your metrics.

Use template variables in annotations

You can use template variables in both annotation query modes to filter annotations based on dashboard variable selections. Grafana resolves the variables at query time using the current dashboard variable values.

Metric query with a variable:

```
nonNegativeDerivative(deploys.$service.count)
```

Events query with a variable:

```
deploy $environment
```

For more information about template variables, refer to Graphite template variables.

End-to-end example: correlate deployments with latency

This walkthrough shows how to combine a Graphite events annotation query with a metric panel so you can see whether deployments affect request latency. The result is a time series panel that plots latency with a vertical marker at each deployment.

Before you start

Ensure you have the following:

- A Graphite data source configured in Grafana.
- A latency metric in Graphite, for example `stats.timers.myservice.request.latency.p95`.
- Permission to create events in Graphite.

Step 1: Record a deployment event

When your pipeline deploys the service, post an event to Graphite tagged `deploy` and `myservice`:

```
curl -X POST http://<GRAPHITE_HOST>/events/ \
  -H "Content-Type: application/json" \
  -d '{
    "what": "Deployed myservice v2.5.0",
    "tags": ["deploy", "myservice"],
    "data": "Rolled out connection pool changes"
  }'
```

Replace `<GRAPHITE_HOST>` with your Graphite server address.

Step 2: Build the latency panel

1. Create a dashboard and add a **Time series** panel.
2. Select the **Graphite** data source.
3. In the query editor, build the latency query, for example `stats.timers.myservice.request.latency.p95`.
4. Confirm the panel plots your latency data over the selected time range.

Step 3: Add the deployment annotation query

1. In the dashboard toolbar, click the **Add new element** icon (blue plus sign), then click **Annotation query**.
2. Enter `Deployments` as the name and select a color that stands out against the latency line.
3. Click **Open query editor** and select the **Graphite** data source.
4. In the **Graphite events tags** field, enter `deploy myservice`.
5. (Optional) Click **Test annotation query** to confirm the query returns events.
6. Click **Close**, then **Save**.

Step 4: Interpret the result

The panel now shows a vertical marker at each deployment. Hover over a marker to see the deployment title from the event's `what` value. If latency rises immediately after a marker, the deployment is a likely cause. If latency is steady across markers, deployments aren't degrading performance.

To extend the dashboard, add a second annotation query tagged `incident` in a different color, so you can tell whether latency changes follow deployments or unrelated incidents.

Troubleshoot annotation issues

The following section addresses common issues you might encounter when using Graphite annotations.

Annotations don't appear

Possible causes and solutions:

Cause	Solution
Time range doesn't include the events	Expand the dashboard time range to include the event timestamps.
Metric query returns only zero values	Annotations are created only for non-zero data points. Verify the metric produces non-zero values at event times.
No matching events	Confirm events exist in Graphite for the supplied tags and time range.
Both fields are populated	Clear one of the fields. Grafana uses the metric query mode whenever the **Graphite Query** field has a value.

Annotation title is empty

In events query mode, Grafana uses the event's `what` value as the title. Ensure the `what` field is populated when you create events in Graphite.

For more troubleshooting guidance, refer to Troubleshoot Graphite data source issues.

□□□ #2
□ □□□ □□□ 2026-10-31 15:56:49 UTC
□ □□□ □□□ 2026-10-31 15:56:52 UTC