Stackdriver Trace shows the details for a trace in the **Trace details** pane of the **Trace list** window. This view shows a summary of details about the request, a graphical timeline that shows the root span for the request and subspans for any RPC calls, and a detailed view of latency data collected for the spans.

A new menu-driven filtering model for the **Trace list** page is in beta (/products/#product-launch-stages) testing. pants are selected randomly for the beta UI. If you are selected a banner is displayed. To use the beta, you can ignore r or click **Dismiss**. To use the generally available **Trace List** page, click **Switch to classic mode**. If you dismiss the ban en want use the classic mode, click **Refresh** ⟳ in your browser, and then click **Switch to classic mode**.

cumentation on the beta UI, see Trace list window (beta) (/trace/docs/finding-traces#beta-ui). This page applies to I

1. In the Google Cloud Console, go to the **Trace list** window:

   Go to Trace List (https://console.cloud.google.com/traces/traces)

2. There are two ways to display detailed information about a trace:

   - Click its URI listed in table.

   - Click its dot in the graph.

The first time you choose a trace to examine, two new panes are displayed on the **Trace list** window:

- One of the new panes displays a graphical representation of latency data for the request using a waterfall graph. By default, the root span of the selected trace is highlighted.

- The other new pane displays detailed information about the row highlighted in waterfall graph.

These two panes are refreshed with new data each time you select a trace to examine:

Timeline
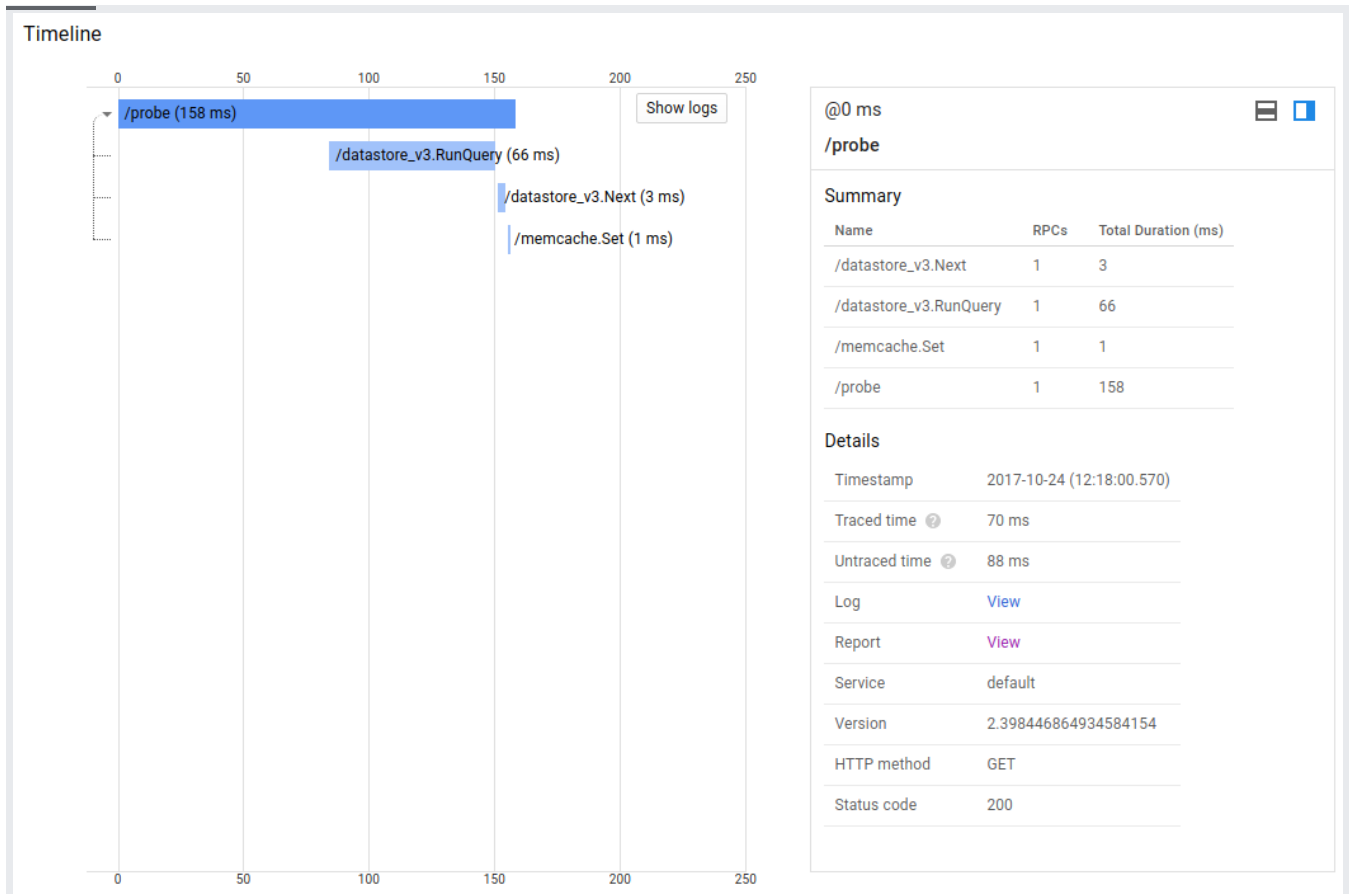
Each row in the waterfall graph corresponds to a span in the trace. Displayed on the span are two attributes:

- The name of the RPC call in the format *service_name.call_name*. For example, `datastore_v3.RunQuery`.

  For App Engine standard environment, the internal name reported here might not match the name from a language-specific service API.

- The time it took to complete the round-trip RPC call.

If you click **Show events**, then the waterfall graph is redrawn and event annotations are included as rows in the graph. For more information on annotations, see Annotating spans (/trace/docs/viewing-details#annotating_spans).

Adjacent to the waterfall graph is the span-details pane. This pane displays detailed information about the row highlighted in the waterfall graph. The data displayed varies depending on the element that is highlighted.

If you highlight a span, then the details include the URI name for the span and the relative start time and the name of the RPC call:

Property	Description
Relative start time	The first entry is the relative start time for the span. For root spans, this value is always @0 ms. For subspans, this is the time when the subspan started relative to the request start. That is, this value displays how long it takes from the start of the overall request, for this RPC call to begin.
Name	The next entry is the name of the RPC call. Names are formatted as *service_name.call_name*. For example, datastore_v3.RunQuery. For App Engine standard environment, the internal name reported here might not match the name from a language-specific service API

When a root span is highlighted, the next entry is the **Summary** table. There is a row in the table for the root span, and one row for each RPC called by a subspan:

Column	Description
Name	Name of the RPC call in format *service_name.call_name*. Displayed in the span label.
RPCs	The number of times the procedure was called.
Total Duration	The total time spent performing the RPCs.

The final entry in the pane is the **Detail** table. This table lists metadata about the span. The following properties are always displayed:

Property	Description
Timestamp	The time at which the application received the incoming request.
Log	A link to a log entry if log data is available. If log data isn't available, this row is omitted.
Report (Not available for beta)	A link to the most recent analysis report that includes this trace. If no reports include this trace, this row is omitted.

Property	Description
Labels	A table of labels and values in the request. Labels are application specific. A few common labels, and their meaning, are listed in the following table:

`http/host`	Hostname where the application is running.
`http/response/size`	Number of bytes in the HTTP response body.
`http/url`	Relative URL for the request.
`gae/request_log_id`	Internal App Engine request ID.

Additional properties might be listed when your request type is HTTP or when your application is running on App Engine. The following table displays a partial list of some commonly observed properties:

Property	Description
Traced time	(HTTP only) The aggregate time it took for all RPC calls to be completed.
Untraced time	(HTTP only) The time during which no RPC calls were made. That is, this measures the time spent locally in the application. Note that a longer than typical untraced time at the beginning of a request can often be caused by a new instance being created to handle the load.
HTTP method	(HTTP only) The HTTP method for the request.
Service	(App Engine only) The App Engine service that handled the request. For more information, go to App Engine (/appengine/docs/).
Version	(App Engine only) The version of the application that handled the request..

If you highlight an event, then the event details are displayed in the details pane. By default, events aren't displayed in the waterfall graph. To include them in the waterfall graph, click **Show events**.

Insights on the performance of the request are shown below the timeline, when insights are available.

For more information, go to Insights (/trace/docs/trace-overview#view_insights).

You can annotate a trace by using Stackdriver Trace API v1 or Stackdriver Trace API v2. This section describes the different options available to you.

You can add annotations to spans by creating a `labels` object and attaching it to the TraceSpan (/trace/docs/reference/v1/rest/v1/projects.traces#tracespan) object when you use Stackdriver Trace API v1 `patchTraces`.

These annotations are displayed as labels when you view a traces' details. For more detail, go to Viewing trace details (#viewing_trace_details) on this page.

You can annotate spans by creating an `attributes` object and attaching it to the Span (/trace/docs/reference/v2/rest/v2/projects.traces/batchWrite#span) object when you use Stackdriver Trace API v2 `batchWrite`.

These annotations are displayed as labels when you view a traces' details. For more detail, go to Viewing trace details (#viewing_trace_details) on this page.

roduct or feature is in a pre-release state and might change or have limited support. For more information, see the pr _stages (/products/#product-launch-stages).

You can add annotations and message event to spans by creating a TimeEvents (/trace/docs/reference/v2/rest/v2/TimeEvents) object and attaching it to the Span (/trace/docs/reference/v2/rest/v2/projects.traces/batchWrite#span) object when you use Stackdriver Trace API v2 `batchWrite`.

The `TimeEvents` object is an array of `TimeEvent` objects, each one contains a message event and an annotation.

To view the `TimeEvents` for a trace, go to the traces' waterfall graph and click **Show events**. If you select a `TimeEvent`, then its details are displayed in the details pane (#span_information).

Trace associates logs with traces by using annotations in the log entries. When you use the Trace client libraries, you can associate a trace with a log entry by setting the `trace` field in the LogEntry object (/logging/docs/reference/v2/rest/v2/LogEntry).

The `trace` field must be set to a string with the following format:

where `[PROJECT_ID]` is your Google Cloud project ID and `[TRACE_ID]` is the trace identifier

You can associate a span with a log entry by setting the `span_id` field in the LogEntry object (/logging/docs/reference/v2/rest/v2/LogEntry). In this case, set the `span_id` field to the 16-character hexadecimal encoding of the span's ID. For example, a span with an ID of `74` would be represented as `000000000000004a`. When you are using trace sampling, it's possible that a log entry is created when the trace itself isn't captured.

You can view the log entry for a trace alongside the waterfall graph or in the Logs Viewer. When you use the Logs Viewer, it's automatically restricted to the timestamp range of the trace. If there are no log entries to display, the Logs Viewer displays the message `No entries found matching current filter`.

From the **Trace details** pane, to view the log entry for the trace, do one of the following:

- To display the trace log entries alongside the waterfall graph, go to the waterfall graph and click **Show logs**. When **Show logs** isn't displayed, no log entries are available.

- To view the log entry in the Logs Viewer, click **View** next to the label **Log** in the **Details** section. Note that when you have a Cloud Load Balancing trace, click **View** next to the label **VM Log**.

For more information about viewing log entries in Stackdriver Logging, see Viewing logs (/logging/docs/view/overview).

To view any log entries, you must have the `logging.logEntries.list` permission in your project. This permission is provided by the **Logs Viewer** and **Project Viewer** Cloud Identity and Access

Management (Cloud IAM) roles.

To view VM instance logs, you must have the `compute.instances.get` permission in your project. This permission is provided by the **Compute Engine Network Viewer** and **Project Viewer** Cloud IAM roles.

- [Creating analysis reports](/trace/docs/analysis-reports) (/trace/docs/analysis-reports)
- [Viewing traces across projects](/trace/docs/cross-project-traces) (/trace/docs/cross-project-traces)