Manage quotas

The Google Calendar API has quotas to make sure that it's used fairly by all users. There are three important limitations to consider when using the Calendar API:

- **API usage quotas** are enforced per project and per user. See the next section for more information.
- **General Calendar usage limits**: Avoid Calendar use limits (https://support.google.com/a/answer/2905486).
- **Operational limits:** you might be rate limited at any time. For example if you attempt to write to a single calendar in quick succession.

Types of Calendar API usage quotas

Two types of quotas are enforced:

- **Per minute per project:** This is the number of requests made by your Google Cloud project.
- **Per minute per project per user:** This is the number of requests made by any one particular user in your Cloud project. This limit aims at helping you ensure a fair distribution of usage among your users.

Quotas are calculated per minute using a sliding window, so a rapid burst of traffic which exceeds your per-minute quota during one minute will result in rate limiting during the next window to ensure that, on average, your usage remains within the quotas.

If either quota is exceeded, you are rate limited and receive a `403 usageLimits` status code (https://developers.google.com/workspace/calendar/api/guides/errors#403_rate_limit_exceeded) or a `429 usageLimits` status code (https://developers.google.com/workspace/c…ur queries. If this happens, here's what you …

1. Make sure you follow all the best p…ize traffic patterns (#randomize_traffic_pa…s).
2. If your project is growing and you h…

per-project quota (#quota_increase).

3. If the per-user quota limit is hit, you can do the following:

    - If you use a service account, allocate the load to users (#proper_accounting_with_service_accounts) or split it between multiple service accounts.
    - While you can request an increase in the per-user quota, in general it is not recommended to increase it above the default value as your application may start to hit other types of limits, for example general calendar use limits (https://support.google.com/a/answer/2905486), or operational limits.

Request for quota increase

To view or change usage limits for your project, or to request an increase to your quota, do the following:

1. If you don't already have a billing account (//cloud.google.com/billing/docs/how-to/manage-billing-account) for your project, then create one.
2. Visit the Enabled APIs page of the API library (https://console.cloud.google.com/apis/enabled) in the API Console, and select an API from the list.
3. To view and change quota-related settings, select **Quotas**. To view usage statistics, select **Usage**.

Use exponential backoff

When we want you to slow down your rate of requests, we will return a 403 "usageLimits" response or a 429 response (see the full error documentation (/workspace/calendar/api/guides/errors#403_calendar_usage_limits_exceeded)). This is not a fatal error and we expect you to retry the request after a short interval. If requests are still arriving too quickly, we will ask again, and so on. For this to work correctly, it is important that the delays between requests increase over time.

Generally, you should use *truncated exponential backoff*; the Cloud Storage documentation

(https://cloud.google.com/storage/docs/retry-strategy) has a good explanation of how this works and the preferred algorithm. If you're using a Google client library, this will normally be handled for you; consult your library documentation. Normally, you should use the library implementation rather than write your own.

Randomize traffic patterns

Calendar clients are prone to spiky traffic patterns caused by multiple clients performing operations at the same time. For example, a common bad practice for a Calendar client is to perform a full sync at midnight. This almost certainly would lead to exceeding your per-minute quota and result in rate limiting and backoffs.

To avoid this, make sure that your traffic is spread throughout the day wherever possible. If your client needs to do a daily sync, have the client determine a random time (different for each client). If you need to perform an operation on a regular basis, vary the interval +/- 25%. This will distribute the traffic more evenly and provide a much better user experience.

Use push notifications

A common use case is to want to perform an action whenever something changes in the user's calendar. An anti-pattern here is to repeatedly poll every calendar of interest. This will very quickly use up all your quota—for example, if your application has 5,000 users and polls each user's calendar once a minute, then this will require a per-minute quota of at least 5,000 even before any work is done.

Server-side applications can register for push notifications, which allows us to notify you when something of interest happens. These require more work to set up, but allow for dramatically more efficient use of your quota, and provide a better user experience. Make sure you specify the `eventType` for which you want to be notified. For more information, see Push notifications (/workspace/calendar/v3/push).

Proper accounting with service accounts

If your application is performing requests using domain-wide delegation

(/identity/protocols/oauth2/service-account#delegatingauthority), by default the *service account* is charged with regard to "per minute per project per user" quotas, and not the user you're impersonating. This means that the service account will likely run out of quota and be rate-limited, even though it might be operating on multiple users' calendars. You can avoid this by using the **`quotaUser`** URL parameter (or **`x-goog-quota-user`** HTTP header) to indicate which user will be charged. This is used only for quota calculations. See Limiting requests per user (https://cloud.google.com/apis/docs/capping-api-usage#limiting_requests_per_user) in the Cloud documentation for more information.

Test quota limit handling

To ensure that your application can gracefully handle reaching quota limits in practice (e.g. by doing retries with exponential backoff (#backoff)) and to minimize any potential disturbances to your users we strongly recommend testing this scenario out in a real environment.

In order for such a test not to interfere with your real application usage, we recommend registering a separate test-only project in Google API Console (https://console.cloud.google.com/) and configuring it (https://developers.google.com/workspace/calendar/api/guides/auth) in a way similar to your production project. You can then set (https://console.cloud.google.com/iam-admin/quotas) artificially low quotas for this project and observe the behavior of your application.

Pricing

All use of the Google Calendar API is available at no additional cost. Exceeding the quota request limits doesn't incur extra charges and your account is not billed.

Last updated 2025-07-30 UTC.