

GUROBI
OPTIMIZATION

Gurobi Instant Cloud Guide

Copyright © 2026, Gurobi Optimization, LLC

Nov 09, 2026

Revision: e14364076

CONTENTS

1 Getting Started **3**
- 1.1 First Solve in the Cloud . . . 3
- 1.2 First Tuning in the Cloud . . . 5
- 1.3 First Distributed Optimization in the Cloud . . . 6
- 1.4 First Distributed Tuning in the Cloud . . . 9
- 1.5 Using Instant Cloud in a Program . . . 11
- 1.6 Security and Network Settings . . . 11

2 Instant Cloud Manager **15**
- 2.1 Overview . . . 15

3 Instant Cloud REST API **33**
- 3.1 REST API v2 . . . 33

This is the guide for using the Gurobi™ Instant Cloud, which provides Gurobi Remote Services via cloud computing. Gurobi Instant Cloud is designed specifically to streamline the use of Cloud resources for the Gurobi Optimizer. This is the easiest way to use the Gurobi in the Cloud; no prior experience with cloud computing is needed. Gurobi Instant Cloud also provides additional features such as:

- Fully automated machine provisioning (AWS or Azure)
- Web interface to manage your machines and pools
- Dashboard to monitor your optimization jobs
- Machine and Job history
- REST API

If you are new to Gurobi Instant Cloud, you can quickly start your first solve or tuning sessions by following the *Getting Started* guide. You will also find detailed descriptions of the *Instant Cloud Manager* used to configure your Cloud environment. Finally, if you need to integrate Instant Cloud in a custom solution or framework, the *REST API* will give you all the tools to automate your processes.

Please check this document periodically to ensure you have the latest instructions for the Gurobi Cloud. Other cloud options exist. Please contact sales@gurobi.com to discuss other options.

CHAPTER

ONE

GETTING STARTED

You can get started with a few simple steps as Gurobi Instant Cloud comes with a predefined configuration. Let's review some typical use cases.

In order to access Gurobi Instant Cloud you need to register and request a Cloud license from your sales account manager, or sign up for a trial if you are eligible.

You also need to install the latest Gurobi Optimizer.

Finally, we recommend that you subscribe to updates from the Gurobi status page to receive notifications about scheduled maintenance and incidents.

1.1 First Solve in the Cloud

You can perform your first solve in a few simple steps:

1.1.1 1. Open the Instant Cloud Manager

Go to cloud.gurobi.com. If you are not logged in, you will be prompted for your credentials. If you do not have an account, please register and contact Gurobi to request an evaluation license.

1.1.2 2. Download the default license file

The list of licenses is displayed in the Instant Cloud Manager and your default license file is ready to be downloaded with the following button.



The license file contains the default access ID and secret key for the selected license. You just have to place this file in your home directory which takes precedence, or in one of the following shared locations:

- C:\gurobi\ on Windows
- /opt/gurobi/ on Linux
- /Library/gurobi/ on Mac OS X

In case you previously had a license file installed, please make sure to replace it, or set the environment variable `GRB_LICENSE_FILE` to point to the cloud license file, it will override the default locations.

1.1.3 3. Solve

You can try to solve any predefined MPS file provided with the Gurobi distribution. Here is an example on Mac OS X:

```
$ gurobi_cl /Library/gurobi1001/macos_universal2/examples/data/afiro.mps
Waiting for cloud server to start..........
Capacity available on '999999-default' cloud pool - connecting...
Established HTTPS encrypted connection with Compute Server

Gurobi Optimizer version 10.0.1 build v10.0.1rc0 (mac64[x86])
Copyright (c) 2023, Gurobi Optimization, LLC

Read MPS format model from file /Library/gurobi1001/macos_universal2/examples/data/afiro.
↪mps
Reading time = 0.12 seconds
AFIRO: 27 rows, 32 columns, 83 nonzeros
Optimize a model with 27 rows, 32 columns and 83 nonzeros
Coefficient statistics:
  Matrix range    [1e-01, 2e+00]
  Objective range [3e-01, 1e+01]
  Bounds range    [0e+00, 0e+00]
  RHS range       [4e+01, 5e+02]
Presolve removed 18 rows and 20 columns
Presolve time: 0.73s
Presolved: 9 rows, 12 columns, 32 nonzeros

Iteration    Objective       Primal Inf.    Dual Inf.      Time
       0   -4.8565680e+02   1.363638e+02   0.000000e+00      1s
       3   -4.6475314e+02   0.000000e+00   0.000000e+00      1s

Solved in 3 iterations and 0.74 seconds
Optimal objective -4.647531429e+02
```

As you can see in the log, the client automatically starts the pool and connects to it. If you wish, you can check the status of the machine using the Instant Cloud Manager. If you run again a new solve, you will notice that it can start right away because the machine is already available.

1.1.4 4. Terminate the pool (optional)

The machine will auto-terminate once it stayed idle for a duration limit called the idle shutdown. The default idle shutdown is 60 minutes, and it can be changed in the settings of the pools and your preferences. Otherwise, you can terminate the pool manually in the Instant Cloud Manager, by selecting the default pool and clicking on the terminate button.



1.2 First Tuning in the Cloud

You can perform your first tuning in a few simple steps. If you already installed your default cloud license file, you can go directly to step 3.

1.2.1 1. Open the Instant Cloud Manager

Go to cloud.gurobi.com. If you are not logged in, you will be prompted for your credentials. If you do not have an account, please register and contact Gurobi to request an evaluation license.

1.2.2 2. Download the default license file

The list of licenses is displayed in the Instant Cloud Manager and your default license file is ready to be downloaded with the following button.



The license file contains the default access ID and secret key for the selected license. You just have to place this file in your home directory which takes precedence, or in one of the following shared locations:

- C:\gurobi\ on Windows
- /opt/gurobi/ on Linux
- /Library/gurobi/ on Mac OS X

In case you previously had a license file installed, please make sure to replace it, or set the environment variable `GRB_LICENSE_FILE` to point to the cloud license file, it will override the default locations.

1.2.3 3. Tune

You can try to tune a MIP MPS file provided with the Gurobi distribution. Here is an example on Mac OS X:

```
$ grbtune /Library/gurobi1001/macos_universal2/examples/data/misc07.mps
Waiting for cloud server to start.............
Capacity available on '999999-default' cloud pool - connecting...
Established HTTPS encrypted connection

grbtune version 10.0.1 build v10.0.1rc0 (mac64[x86])
Copyright (c) 2023, Gurobi Optimization, LLC

Read MPS format model from file /Library/gurobi1001/macos_universal2/examples/data/
↪misc07.mps
Reading time = 0.56 seconds
MISC07: 212 rows, 260 columns, 8619 nonzeros

Solving model using baseline parameter set with TimeLimit=3600s

Solving with random seed #1 ...
Optimize a model with 212 rows, 260 columns and 8619 nonzeros
Variable types: 1 continuous, 259 integer (0 binary)
    [...]
```

As you can see in the log, the client automatically connects to the Instant Cloud server and checks for the pool status. As the machines are not launched yet, Instant Cloud starts the machines and the client reports that it is waiting until capacity is available. Then, it starts the tuning process.

1.2.4 4. Terminate the pool (optional)

The machine will auto-terminate once it stays idle for a duration limit called the idle shutdown. The default idle shutdown is 60 minutes, and it can be changed in the settings of the pools and your preferences. Otherwise, you can terminate the pool manually in the Instant Cloud Manager, by selecting the default pool and clicking on the terminate button.



1.3 First Distributed Optimization in the Cloud

The Gurobi Instant Cloud makes it easy to launch a cluster of machines for distributed optimization. This guide will walk you through the process of completing your first distributed solve in the Cloud.

1.3.1 1. Open the Instant Cloud Manager

Go to cloud.gurobi.com. If you are not logged in, you will be prompted for your credentials. If you do not have an account, please register and contact Gurobi to request an evaluation license.

1.3.2 2. Create a pool with distributed workers

In the Instant Cloud Manager, go to the 'Pools' section and click on the create pool button:



Then, select the 'License' tab and set the number of workers to 2.



Finally, create the new pool. Note that a default name is assigned for you such as pool1.



1.3.3 3. Download the pool license file

The list of pools is displayed in the Instant Cloud Manager and your license file is ready to be downloaded with the following button.



The license file contains the default access ID and secret key for the selected pool. You just have to place this file in your home directory which takes precedence, or in one of the following shared locations:

- C:\gurobi\ on Windows
- /opt/gurobi/ on Linux
- /Library/gurobi/ on Mac OS X

In case you previously had a license file installed, please make sure to replace it, or set the environment variable `GRB_LICENSE_FILE` to point to the cloud license file, it will override the default locations.

1.3.4 4. Solve

You can try to solve a MIP MPS file provided with the Gurobi distribution. Here is an example on Mac OS X:

```
$ gurobi_cl /Library/gurobi1001/macos_universal2/examples/data/misc07.mps
Waiting for cloud server to start..........
Capacity available on '999999-pool1' cloud pool - connecting...
Established HTTPS encrypted connection

Gurobi Optimizer version 10.0.1 build v10.0.1rc0 (mac64[x86])
Copyright (c) 2023, Gurobi Optimization, LLC

Read MPS format model from file /Library/gurobi1001/macos_universal2/examples/data/
↪misc07.mps
Reading time = 0.47 seconds
MISC07: 212 rows, 260 columns, 8619 nonzeros
Optimize a model with 212 rows, 260 columns and 8619 nonzeros
Coefficient statistics:
  Matrix range    [1e+00, 7e+02]
  Objective range [1e+00, 1e+00]
  Bounds range    [1e+00, 1e+00]
  RHS range       [1e+00, 3e+02]

Starting distributed worker jobs...

Using Compute Server as first worker - running now
Started distributed worker on ip-52-91-137-123
Started distributed worker on ip-54-159-77-110

Distributed MIP job count: 3

    Nodes    |    Current Node    |     Objective Bounds      |     Work
 Expl Unexpl |  Obj  Depth IntInf | Incumbent    BestBd   Gap | ParUtil Time

H    0                          4155.0000000          -      -          3s
```

(continues on next page)

(continued from previous page)

```
H    0                          3610.0000000          -      -          3s
H    0                          3500.0000000 1415.00000  59.6%         3s
H    0                          2940.0000000 1415.00000  51.9%         3s
H    0                          2810.0000000 1415.00000  49.6%         4s
    24    22                      2810.00000 1544.28571  45.0%   99%    4s
  1114   475                      2810.00000 1926.66667  31.4%   99%    5s

Ramp-up phase complete - continuing with instance 1 (best bd 2175)

  7533   931 1492.85714    0   48 2810.00000 2175.00000  22.6%   99%    7s
 15311     0 2785.00000   21   13 2810.00000 2810.00000  0.00%   93%    9s

Cutting planes:
  Cover: 2
  Clique: 4
  MIR: 17
  Zero half: 10

Runtime breakdown:
  Active:   8.09s (88%)
  Sync:     0.81s (9%)
  Comm:     0.28s (3%)

Explored 15311 nodes (152346 simplex iterations) in 9.17 seconds
Distributed MIP job count: 3

Optimal solution found (tolerance 1.00e-04)
Best objective 2.810000000000e+03, best bound 2.810000000000e+03, gap 0.0%
```

Within this log, we have highlighted in bold some important steps. First, the client automatically connects to the Instant Cloud server and checks for the pool status. As the machines are not launched yet, Instant Cloud starts the machines and the client reports that it is waiting until capacity is available.

Then the Gurobi Optimizer detects that the pool is setup with 2 distributed workers. So it automatically starts the solve in distributed mode with 3 workers (the master compute server counts as one worker as well).

1.3.5 5. Terminate the pool (optional)

The machine will auto-terminate once it stays idle for a duration limit called the idle shutdown. The default idle shutdown is 60 minutes, and it can be changed in the settings of the pools and your preferences. Otherwise, you can terminate the pool manually in the Instant Cloud Manager, by selecting the created pool and clicking on the terminate button.



1.4 First Distributed Tuning in the Cloud

The Gurobi Instant Cloud makes it easy to launch a cluster of machines for distributed tuning. If you already installed a cloud license file for a pool with distributed workers, you can go directly to step 4.

1.4.1 1. Open the Instant Cloud Manager

Go to cloud.gurobi.com. If you are not logged in, you will be prompted for your credentials. If you do not have an account, please register and contact Gurobi to request an evaluation license.

1.4.2 2. Create a pool with distributed workers

In the Instant Cloud Manager, go to the 'Pools' section and click on the add new pool button:



Then, open the 'License' tab and set the number of workers to 2.



Finally, create the new pool. Note that a default name is assigned for you such as pool1.



1.4.3 3. Download the pool license file

The list of pools is displayed in the Instant Cloud Manager and your license file is ready to be downloaded with the following button.



The license file contains the default access ID and secret key for the selected pool. You just have to place this file in your home directory which takes precedence, or in one of the following shared locations:

- C:\gurobi\ on Windows
- /opt/gurobi/ on Linux
- /Library/gurobi/ on Mac OS X

In case you previously had a license file installed, please make sure to replace it, or set the environment variable GRB_LICENSE_FILE to point to the cloud license file, it will override the default locations.

1.4.4 4. Tune

You can try to tune a MIP MPS file provided with the Gurobi distribution. Here is an example on Mac OS X:

```
$grbtune /Library/gurobi1001/macos_universal2/examples/data/misc07.mps
Waiting for cloud server to start...........
Capacity available on '999999-pool1' cloud pool - connecting...
Established HTTPS encrypted connection

grbtune version 10.0.1 build v10.0.1rc0 (mac64[x86])
Copyright (c) 2023, Gurobi Optimization, LLC

Read MPS format model from file /Library/gurobi1001/macos_universal2/examples/data/
↪misc07.mps
Reading time = 0.26 seconds
MISC07: 212 rows, 260 columns, 8619 nonzeros

Distributed tuning: launched 3 distributed worker jobs

Solving model using baseline parameter set with TimeLimit=3600s

Solving with random seed #1 ...
Optimize a model with 212 rows, 260 columns and 8619 nonzeros
Variable types: 1 continuous, 259 integer (0 binary)
    [...]
```

Within this log, we have highlighted in bold some important steps. First, the client automatically connects to the Instant Cloud server and checks for the pool status. As the machines are not launched yet, Instant Cloud starts the machines and the client reports that it is waiting until capacity is available.

Then the Gurobi Optimizer detects that the pool is setup with 2 distributed workers. So it automatically starts the tuning in distributed mode with 3 workers (the master compute server counts as one worker as well).

1.4.5 5. Terminate the pool (optional)

The machine will auto-terminate once it stayed idle for a duration limit called the idle shutdown. The default idle shutdown is 60 minutes, and it can be changed in the settings of the pools and your preferences. Otherwise, you can terminate the pool manually in the Instant Cloud Manager, by selecting the created pool and clicking on the terminate button.



1.5 Using Instant Cloud in a Program

Using a cloud license file will work seamlessly with any program or supported environment: C++, Python, MATLAB, Java, .Net, C or R. The cloud license file can be easily downloaded from a license, or a pool using the Instant Cloud Manager.

In addition, when programming in C, C++, Python, Java or .Net, the Gurobi client libraries provide you with dedicated environment constructors to specify the access ID, the secret key and optionally the pool. If the pool is not provided, your job will be launched in the default pool associated with your cloud license. Please refer to the Gurobi Optimizer Reference Manual.

Each license comes with a predefined pool called 'default'. You can edit the configuration of pools in the Instant Cloud Manager or create new ones. The updated configuration will be effective for newly launched machines only. So if the machines of a pool are already running, please make sure to terminate them so that new configuration will be taken into account.

One of the important configuration options is the idle shutdown time. When a client program requests a cloud server, it takes some time (usually 1-2 minutes) to launch that server. Rather than forcing client programs to incur this delay each time they run, the Gurobi Instant Cloud leaves a server running until it has been idle for the specified idle shutdown time. In this way, a second client program may find a cloud server already available. You can set this to a small value if you want your server to shut down immediately after your job finishes, or to a very large value if you want your server to always be available.

1.6 Security and Network Settings

Our goal is to provide a secure environment to our customers, and we are continuously monitoring and improving our architecture and processes. In this section, we will review the security features and the required network settings to operate Gurobi Instant Cloud.

1.6.1 Accessing the Cloud Manager

The Cloud Manager is designed to streamline the control of the Gurobi Optimizer on the Cloud. With the Cloud Manager, Gurobi manages AWS EC2 or Azure instances. The Cloud Manager consists of the website **cloud.gurobi.com** and a REST APIs. The main functions of the Cloud Manager are about configuring, controlling and monitoring Gurobi compute servers. No optimization model data is communicated with the Cloud Manager.

When accessing the website, users must be authenticated with their Gurobi accounts. When using the *REST API*, the clients are authenticated with the API key and API secret related to a user account. The communication is secure using the HTTPS protocol (minimum of TLS 1.2) and the Cloud Manager database is encrypted at rest. Access to **cloud.gurobi.com** is also protected by a Web Application Firewall. For security purposes, Gurobi records and monitors the metadata of HTTPS communication.

For better availability and scalability, the Cloud Manager is hosted in different regions of the world. The clients will be routed to the most appropriate available server using a latency based routing. Each region may also provide several instances of the servers. Clients should not hardcode IP addresses to access the Cloud Manager, and should always make sure to use the latest DNS resolution.

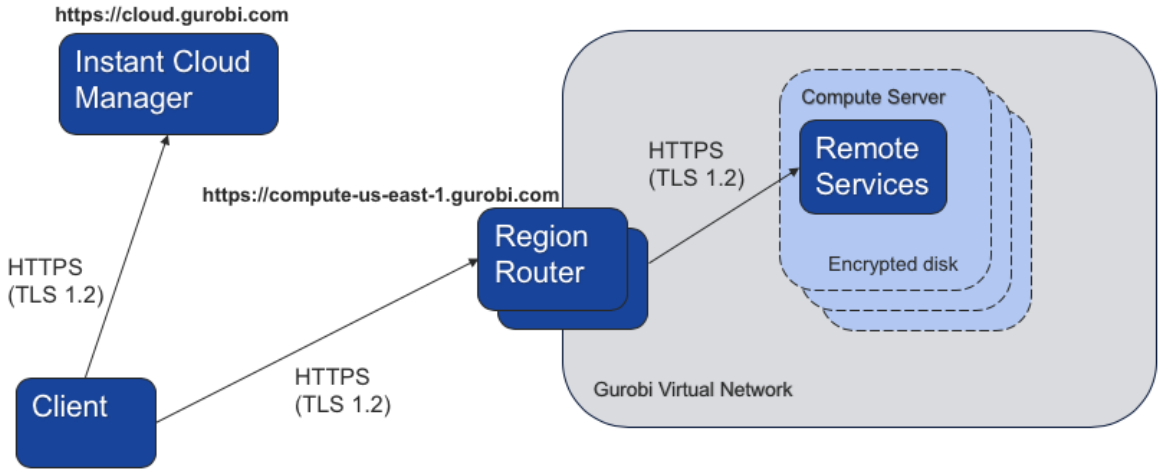
The Gurobi client (gurobi_cl, grbtune, Gurobi library…) will first connect to the Cloud Manager using the secure REST API to check the pool status and launch the compute servers as necessary. In order to enable this connection, the client firewalls must be configured to open the standard HTTPS port 443 to host **cloud.gurobi.com**.

1.6.2 Accessing the Compute Servers

When a Compute Server is started, you get a new EC2 or Azure virtual machine that is not shared with any other Gurobi customers, it is always dedicated. Access to each machine is authenticated with API keys and secured with end-to-end encryption. Machine disks are also encrypted. When the machine is terminated, all optimization data are discarded from memory and disk.

Once the compute server has been launched, optimization commands are exchanged between the client and the server. The communication is secure using end-to-end encryption with HTTPS (minimum of TLS 1.2). The region router consists of a load balancer and a region router. The region router is a reverse proxy that will forward the communication to the appropriate compute server within the Gurobi private virtual network. The load balancer, the region routers and the compute servers all use encrypted HTTPS communication.

The started machines are not accessible directly and passing through the region router is enforced. The diagram below summarizes the architecture with AWS.



As shown below, each region provides a different URL address to its router. Clients should not hardcode IP addresses to access the region routers, and should always make sure to use the latest DNS resolution. In order to enable this connection, the client firewalls must be configured to open the standard HTTPS port 443 to the following hosts depending on the region.

Provider	Region	Router
AWS	us-east-1	https://compute-us-east-1.gurobi.com
AWS	us-west-1	https://compute-us-west-1.gurobi.com
AWS	eu-central-1	https://compute-eu-central-1.gurobi.com
AWS	ap-northeast-1	https://compute-ap-northeast-1.gurobi.com
AWS	ap-southeast-2	https://compute-ap-southeast-2.gurobi.com
Azure	eastus	https://compute-eastus-azure.gurobi.com
Azure	westus2	https://compute-westus2-azure.gurobi.com
Azure	westeurope	https://compute-westeurope-azure.gurobi.com

1.6.3 Managing API keys

The Cloud Manager website is the only place where the API keys can be generated. Multiple API keys can be generated so that keys can be replaced in case one of them has been compromised. Each key is owned by a user. Before disabling a user by contacting the Gurobi support or before deleting an API key, please make sure that you have migrated your applications to new API keys.

1.6.4 Proxies

The architecture is compatible with standard proxy settings using environment variables HTTP_PROXY and HTTPS_PROXY. HTTPS_PROXY takes precedence over HTTP_PROXY for https requests. The values may be either a complete URL or a "host[:port]", in which case the "http" scheme is assumed.

CHAPTER

TWO

INSTANT CLOUD MANAGER

2.1 Overview

Gurobi Instant Cloud is specifically designed to streamline the use of Cloud resources for the Gurobi Optimizer. This is the easiest way to use Gurobi in the Cloud; no prior experience with cloud computing is needed. If you are new to Gurobi Instant Cloud, you can quickly start your first solve or tuning sessions by utilizing the getting started resources. This documentation will help you use the Cloud Manager, which is the web application that allows you to manage your Gurobi Cloud environment.

2.1.1 Accounts

Licenses

The Instant Cloud Manager allows you to list the active licenses associated with your account. The list provides the license id, the number of active machines, the rate plan, the remaining credit and the credit limit. It also indicates if the notifications have been enabled or disabled. The following icon may be displayed:



If an error icon is displayed next to the license id, you can move the mouse over and a tooltip will display the exact reason (typically the license has expired or does not have enough credit). In this case, please contact your sales representative or support.

From the list of licenses, you can perform the following actions:

	Edit and save the notification settings.
	Download the default license. The default license file can be used right away to start machines in the Cloud by using the dedicated default pool of this license. The license file contains the default Access ID and Secret Key for the selected license. You just have to place this file in your home directory which takes precedence, or in one of the following shared locations: • `C:\gurobi\` on Windows • `/opt/gurobi/` on Linux • `/Library/gurobi/` on Mac OS X In case you previously had a license file installed, please make sure to replace it, or set the environment variable `GRB_LICENSE_FILE` to point to the cloud license file, it will override the default locations.
	Display the statements. From the list of licenses, you can also access the latest statement detailing the billing events on the selected license.

License Sharing

Note that you can ask Gurobi support to share a license among users of the same organization. In this case, machines and pools related to a shared license will be accessible by multiple users. Any user of a shared license can create, launch, or terminate machines and pools with the same access rights.

Notifications

Notifications can be automatically generated when a specific situation may require your attention:

- a license remaining credit is low,
- a license is about to expire,
- or a machine is running out of memory (usage over 90%).

Notifications are enabled by default for Silver, Gold and Platinum licenses with default thresholds. You may need to review the settings of your licenses to enable or disable the notifications, and adjust the thresholds according to your needs.

In the license panel, you can open the detail page and edit the notification settings in the notifications tab. Notifications can be enabled or disabled for each license. You can also provide several thresholds to trigger notifications:

- A warning credit notification will be generated if the current credit minus the credit limit is lower than or equal to the warning threshold.
- An urgent credit notification will be generated if the current license credit minus the credit limit is lower than or equal to the urgent threshold.
- A license expiration notification will be generated if the current license expires within the specified remaining days.

You can also set the behavior to receive emails. You can also create a test notification to verify that you are able to receive the emails.

When a notification is generated, it is then accessible in the notification panel. In this panel, you can list the open and closed notifications. Credit notifications are automatically closed when the license has been credited with enough funds. You can filter the notifications by license ID and see the details of each notification. You can also check if the emails were sent and to which recipients.

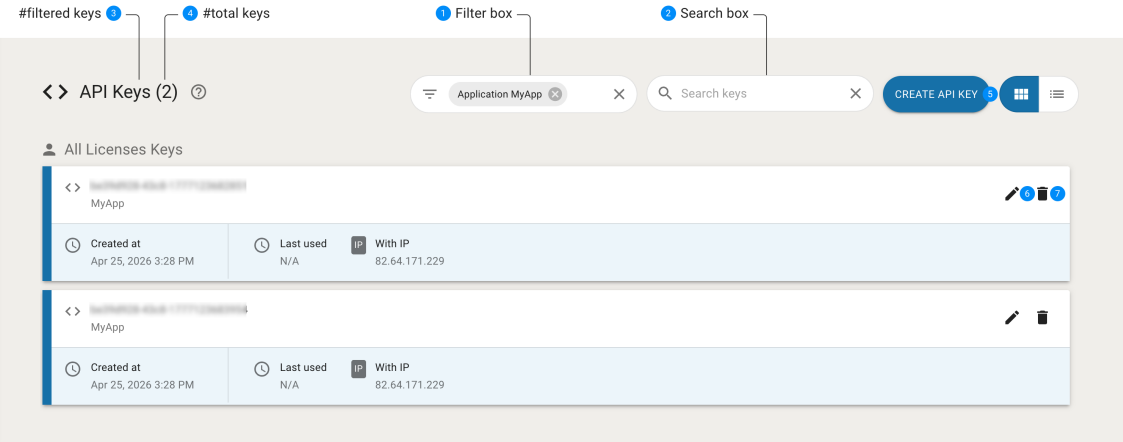
Note that each user has a preference setting to unsubscribe from notifications. If a user unsubscribes for his/her account, the user will no longer receive any notifications. However, notifications will remain listed in the notification panel.

API keys

An API key consist of an access ID and a secret key. Different types of API keys can be managed:

API keys for all licenses	The API keys can access all the licenses, machines and pools associated with your account. Multiple keys can be generated.
API keys for a specific license	The API keys can only access the specified license. Only machines and pools created for this license can be managed. Multiple keys can be generated.

The API Keys page displays all of the keys associated with your account:



1	API keys can be queried using the Filter box.
2	API keys can be filtered using the Search box.
3 4	The number of tiles / rows of the table. When the page has been filtered, two numbers are displayed to the right of the page title showing respectively the number of API keys matching the current filtering and the total number of API keys. If no API keys have been filtered, only the total number of API keys is displayed.
5	Button opens a dialog to create a new API key.
6	Button to edit the application name and description for the API key.
7	Delete a key. Once deleted, any application using this key will be immediately blocked. Note that the default key of a specific license cannot be deleted.

Multiple Keys

The API supports multiple simultaneous API keys per license or for all licenses. This is useful when there is a need to replace an API key that is already in use by an application or some users. With this feature, you can generate a new key, update the applications or notify the users, and finally delete the old key when necessary. During this process, you will not incur downtime of the applications or experience user access issues. You can also generate multiple keys for tracking purposes.

Preferences

The preferences section enables you to define and store default values and options.

Idle Shutdown	Idle shutdown specifies a duration limit in minutes after which the machine will auto-terminate when there is no job running or in queue. New pools and one-time servers will take this value as default.
Idle Job Timeout	This timeout specifies a duration limit in minutes after which the job will auto-terminate when there is no command sent to the server. This is useful to avoid using resources when some clients may leave their connection open (for example in an interactive python shell) while not being active. If the value is 0, it will be disabled. New pools and one-time servers will take this value as default.
Job Limit	When a machine license is 'full compute server', a maximum number of concurrent jobs can be set. New pools and one-time servers will take this value as default.
Provider	The cloud provider, Amazon Web Services (AWS) or Microsoft Azure.
Region	The region references the location of the data center where the machines are provisioned. Select a region that is closer to your operations to minimize latency. New pools and one-time servers will take this value as default.
Machine type	Different machine types can be provisioned depending on the requirements (mainly memory and CPU). New pools and one-time servers will take this value as default.
Job History	Changes the default selection to enable the job history with new pools and machines. Existing pools are not impacted. To enable the job history for an existing pool, please edit the pool options. When selected, job metadata and engine logs will be archived for up to 90 days and accessible in the job history view.
Max compute servers per pool	When a pool is created, a size must be specified which represents the number of compute servers part of the pool. The default maximum size can be changed to go over the predefined limit.
Max distributed workers per compute server	When a machine license is 'full compute server', a set of distributed workers can be defined. The default maximum number of workers can be changed to go over the predefined limit.
Subscribe to notifications	If disabled, notifications will not be sent by email to this account even if the notifications has been enabled on a specific license. You can still list notifications in the notifications panel.

2.1.2 License Core Limit

Overview

Gurobi Instant Cloud licenses include limits on the number of concurrent cores you can use. The **concurrent core limit** controls how many CPU cores can be running across all your machines at the same time. This limit applies to all machines of a given license, across different pools.

Understanding Concurrent Cores

What Are Concurrent Cores?

Concurrent cores are the total number of CPU cores actively running across all your compute machines at any given moment. Your license includes a maximum limit for concurrent cores.

Example:

If your plan allows 160 concurrent cores, you could run:

- Two 64-core machines + one 32-core machine (64 + 64 + 32 = 160 cores)
- Five 32-core machines (5 × 32 = 160 cores)
- Any other combination totaling up to 160 cores

What Counts Toward Your Limit?

Only **compute server nodes** running optimization jobs count toward your concurrent core limit.

What Does NOT Count?

The following do NOT count toward your concurrent core limit:

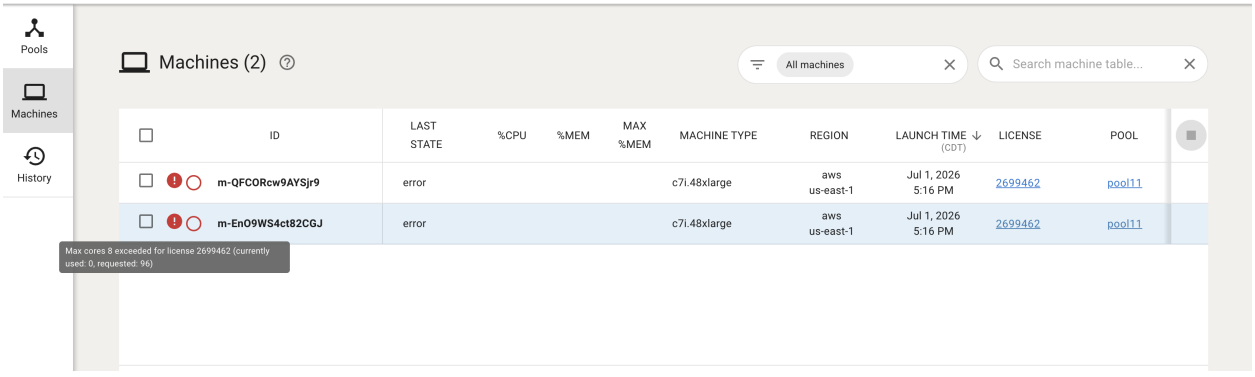
- **Distributed worker nodes** used in distributed algorithms
- **Stopped or terminated machines** (only running machines count)

How Limits Work

When You Reach Your Limit

When you attempt to start a new machine that would exceed your license's concurrent core limit, the machine may be rejected. However, the system includes a retry mechanism - if the core count is about to decrease (for example, because another machine is being terminated), the request will be retried up to 4 times and may eventually be accepted.

If all retries fail, you will see an error message indicating you've reached your concurrent core limit. The error appears on machines that fail to start due to the core limit.



What to do:

- Stop one or more running machines to free up cores, then try again

- Contact your account manager to discuss upgrading your license or increasing your limit

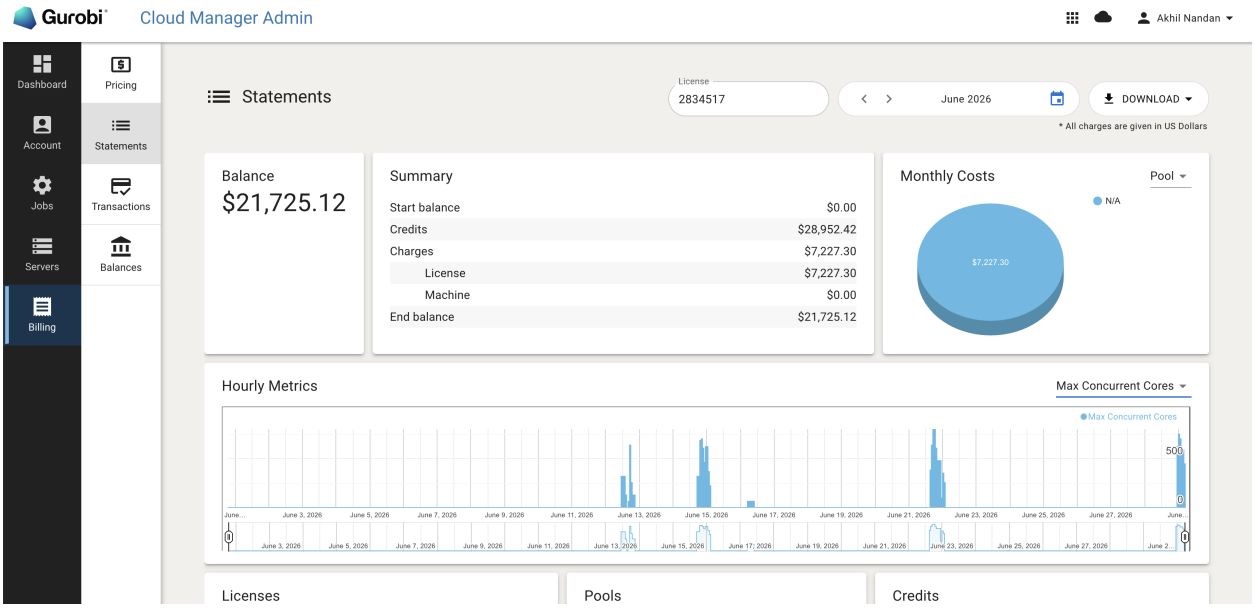
Monitoring Your Usage

Real-Time Monitoring

To view your current concurrent core usage:

1. Log in to the Gurobi Cloud Portal
2. Navigate to **Instant Cloud Manager**
3. Go to **Billing** → **Statements**
4. Click on a statement to view running machines
5. In the **Hourly Metrics** section, select **Max Concurrent Cores** to see core usage over time

The hourly metrics chart shows your concurrent core usage throughout the selected billing period, helping you identify peak usage times and patterns.



You can also view currently running machines by clicking on the statement, which shows each machine and its core count. Add up the cores from all running machines to see your current total.

API Monitoring

You can also monitor usage programmatically via the Instant Cloud *REST API*:

Endpoint: `GET /machines`

Response: Returns all active machines

Note: The API response does not include a core count field for each machine. The response includes machineType (e.g., c5.4xlarge, Standard_F16s_v2), but you will need to map machine types to their core counts separately to calculate total concurrent core usage.

Analyzing Historical Usage

Understanding Past Usage Patterns

The **Billing** section provides detailed historical data to help you understand your usage patterns and make informed decisions:

What You Can See:

- **Hourly core usage** — How many cores were running at any given hour
- **Peak usage times** — When you used the most concurrent cores
- **Average usage** — Your typical core consumption
- **Machine types used** — Which machine sizes you deployed

How to Use This Information:

1. **Identify Peak Demand Periods:**
    - Look for hours when you hit or came close to your limit
    - Note the days and times when this happens
    - Example: If you see peaks every Monday morning, plan accordingly
2. **Calculate Your 95th Percentile Usage:**
    - This represents your typical high-water mark
    - Choose a plan that accommodates this level to avoid disruptions
    - Example: If 95% of the time you use under 120 cores, a 160-core plan provides good headroom
3. **Understand Seasonal Variations:**
    - Identify if your usage spikes during certain months
    - Plan for these peaks by optimizing your workload scheduling or consider adjusting your license if increased capacity is consistently needed
4. **Analyze Machine Type Efficiency:**
    - Compare costs and performance of different machine sizes
    - Example: Running one 64-core machine may be more efficient than four 16-core machines for certain workloads

Frequently Asked Questions

Q: How do I know how many cores a machine type has?

A: Machine core counts are shown when you create or view pools. Common machine types include 8-core, 16-core, 32-core, and 64-core configurations. The exact types available depend on your cloud provider (AWS or Azure) and region.

Q: What happens to running jobs if I hit my limit?

A: Running jobs are never interrupted. The limit only prevents new machines from starting. Your existing jobs will continue to completion.

Q: Do distributed workers count toward my limit?

A: No. Distributed worker nodes used in distributed MIP or concurrent optimization do not count toward your concurrent core limit. Only compute server nodes count.

Q: Can I temporarily exceed my limit for critical work?

A: The system enforces the concurrent core limit. If you need additional capacity, contact your account manager to discuss options for your license.

Q: How can I see my historical concurrent core usage?

A: The **Billing** section provides historical machine usage data, including the number of cores used over time. This helps you understand your usage patterns and plan capacity needs. You can view hourly metrics, peak usage times, and calculate your 95th percentile usage.

Q: What should I do if I frequently hit my limit?

A: Review your usage patterns in the Billing section. If you consistently need more capacity, contact your account manager to discuss adjusting your license to better match your workload. Provide historical usage data to help them recommend the right plan.

Q: How much headroom should I have in my plan?

A: A good rule of thumb is to choose a plan where your 95th percentile usage is about 75-80% of your limit. This gives you headroom for occasional spikes while avoiding paying for unnecessary capacity. For example, if you typically use up to 120 cores, a 160-core plan provides appropriate buffer.

Q: Can I mix different machine sizes?

A: Yes, by using different pools. Each pool defines a single machine type, so all machines within a pool are the same size. To use different machine sizes, create multiple pools with different machine types.

2.1.3 Jobs

Jobs

You can get the list of jobs running or recently ended on machines and pools. Note that when a machine is terminated, the job executed on this machine will disappear from this list. If you need to keep visibility on all executed jobs, see the Job History feature.

A job can be in one of the following states:

State	Description
running	The job is currently running.
completed	The job has recently completed successfully.
aborted	The job was aborted.
disconnected	The job lost the connection with the client due to network issues or because the client was terminated.
failed	The job itself experienced an unexpected termination due to a lack of memory or an internal problem.

From the job list, you can perform the following actions. Some actions are specific to one job and some others can be applied to multiple jobs using the checkboxes.

	Abort selected jobs.
	Open the job dashboard.
	Download the job log.

Queued Jobs

Each machine in the pool has a job limit that indicates how many jobs can be running concurrently. If the capacity of all the machines of the pool is reached, jobs will be placed in a queue.

With the queued job list, you know when jobs were added to the queue, the priority, and the index in the queue. You can also perform the following actions (for one or multiple jobs):

	Abort selected jobs.

History Jobs

While the active job list provides a view of jobs currently queued, running or recently ended, the job history provides a persistent list of ended jobs. The job history menu option lets you browse and filter these jobs. The engine log can also be downloaded. When using this feature, the job metadata and the engine logs are stored on the Instant Cloud secured servers for a maximum of 90 days. This feature can be disabled on selected pools or machines in the options panel.

From the job list, you can perform the following actions:

	Download the complete log file.

Tabs below the table provide details on the selected job, including the job description, client attributes, status, model information, and algorithm information.

The maximum number of jobs retrieved is 2000. Filtering of jobs will only be performed locally based on the provided results.

Job Dashboard

The job dashboard provides different views about a given job with the purpose of allowing an optimization expert to monitor the job execution. It displays a set of tiles about the job status, the model information, and the parameters. It also provides a tile to display current charts, and another tile to display the engine log. If the job has processed several models, then only information about the current one is displayed.

From the dashboard view, you can perform the following actions:

	Select different types of charts if available.
	Download the complete log file.
	Display the log from the start. When opening the dashboard, only the last 50 lines of the log will be shown and then the log will fill out incrementally if the job is still running. This button lets you display the entire log if available.

2.1.4 Servers

Pools

A pool defines the configuration of one or more machines. When the Gurobi client library connects to the Instant Cloud to get an environment of Cloud machines, the pool is used to reference the group of machines to use. If the machines are already started, they will be used right away. If some of the machines are not running, Instant Cloud will then launch them automatically so that the client will be able to start the optimization as soon as they become available. The pools fully automate the process of starting machines and waiting for them to be ready.

A pool can be shared by multiple applications and users. The first application accessing the pool will trigger the launch of the required machines, and the subsequent solves by any other applications will be able to execute without waiting for the machines.

A pool can be created to distinguish configurations used in different contexts: large or small optimization problems, development vs production deployments, different regions (data centers) to minimize latency…

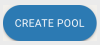
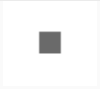
A pool has a name (alphanumeric characters only), and its size indicates how many compute servers are part of the pool. The pool configuration can also specify the number of distributed workers to associate with each compute server. The compute servers and the distributed workers (if any) have the same configuration: machine type, region, idle shutdown, idle job timeout, job limit, and Gurobi version.

A default pool is automatically created for each license. A default pool cannot be deleted, but its configuration can be changed. When the license file does not specify a pool, the default pool is used. The name of a default pool is `default`.

The status of the pool is displayed with a colored icon:

	The pool is not ready, some machines have not been launched yet.
	The pool is not ready, some machines have been launched but are not available yet.
	The pool is ready, all the machines specified in the pool are available.
	The pool has an error, you can move the mouse over and a tooltip will display the exact reason.

In addition to the status, you can perform the following actions. Some actions are specific to one pool and some others can be applied to multiple pools. In this case, you can toggle the selection using the checkboxes.

	Add a pool. When adding a pool, you are able to specify its name, its license, and all the configuration parameters.
	Scale up or down the pool.
	Edit a pool. The pool configuration can be modified depending on the needs without making changes to the clients or deployed applications. Note that, the name of the pool and its license cannot be changed. Changing the pool configuration will take effect only when new machines are launched. To avoid possible inconsistent configurations among the machines of the same pool, it is highly recommended to terminate the machines first.
	From the list of pools you can also download the pool license file. The license file contains the default access ID and secret key for the selected pool. Place this file in your home directory or in one of the following shared locations: • C:\gurobi\ on Windows • /opt/gurobi/ on Linux • /Library/gurobi/ on Mac OS X If you previously had a license file installed, replace it, or set the environment variable `GRB_LICENSE_FILE` to point to the cloud license file.
	Delete selected pools. When a pool is deleted, all the machines are also terminated even if they were running.
	Terminate pools. The machines launched for a pool will typically auto-terminate based on the idle shutdown parameter. However, it may be useful in some cases to terminate the machines manually.
	Launch pools. A pool will typically be launched automatically by a client when an optimization problem is ready to be processed. However, it may be useful to launch the pool manually. When a pool is launched, the missing machines part of the pool are launched.

Create or Edit Pools

When creating or editing a pool, you will have access to the following properties:

Name	A pool has a name that is unique for a given license. The name must be composed of alphanumeric characters only and the name `default` is reserved. The name of a pool cannot be changed.
Description	An optional description of the pool.
Size	The number of compute servers that must be launched for this pool.
License	The license used for this pool. The license of a pool cannot be changed.
Workers	The number of distributed workers to launch for each compute server.
Idle Shutdown	Idle shutdown specifies a duration limit in minutes after which the machine will auto-terminate.
Idle Job Timeout	This timeout specifies a duration limit in minutes after which the job will auto-terminate when there is no command sent to the server. This is useful to avoid using resources when some clients may leave their connection open while not being active (for example in an interactive python shell). This timeout can also be specified client side by using the property CSIdleTimeout. The default value client-side is 30 minutes. The actual value will be the maximum between the value specified by the client and the value specified by the pool.
Job Limit	A maximum number of concurrent jobs for each compute server. This option can be set only for 'full compute server' license type.
Provider	The cloud provider, Amazon Web Services (AWS) or Microsoft Azure.
Region	The region references the location of the data center where the machines are provisioned. Select a region that is closer to your operations to minimize latency.
Machine type	Different machine types can be provisioned depending on the requirements (mainly memory and CPU). Depending on the machine types and license, the cost of a machine and distributed workers can differ. Instant Cloud can give you more details and an estimate of the costs.

Pool Scaling

It is possible to scale up or down your pools using the Instant Cloud Manager or the REST API. A pool defines a number of compute servers that is actually the minimum number of servers. So when a client starts a pool, the pool becomes ready when the number of servers is ready. Then, you can scale up by requesting additional servers to be added to the pool. When servers are added, they automatically join the cluster of compute servers of the pool and start processing new jobs or jobs already queued in existing servers.

The pool will scale down automatically by using the existing idle shutdown parameter. This means that any server being idle for this time limit will be terminated, up to completely terminate the pool. The pool can also be explicitly scaled down by reducing the number of servers. In this case, if a machine must be stopped but it is already running a job, it will be moved into a draining mode where new jobs will not be processed while running ones will be processed normally. When these jobs have completed, the machine will be automatically terminated.

For better monitoring, the pool view displays the number of machines that are starting, stopping and the ones that are ready.

Machines

Machines are Gurobi compute servers or distributed workers provisioned in the Cloud. A compute server can accept new problems to solve while distributed workers are used as nodes to run distributed optimization or tuning. Machines are launched automatically as part of a pool or manually using the Instant Cloud Manager.

A machine can be in one of the following states:

launching	The machine request has been recorded, and the provisioning will start soon.
pending	The provisioning process has started.
configuring	The machine has been provisioned and the network is being configured.
starting	The machine has been configured and the Gurobi remote services are starting.
idle	The Gurobi remote services are ready to accept new optimization jobs. No optimization jobs are running.
running	The Gurobi remote services are ready to accept new optimization jobs. At least one optimization job is running.
killing	The machine termination has been recorded, and the shutdown process will start soon.
shutting down	The machine is shutting down. Once a machine is fully shutdown, it will be removed from the list. The machine history can be accessed to list recent terminated machines.
error	An error occurred while the machine was being provisioned, running or shutting down. A machine in error state will remain in the list for around 2 minutes and then will be removed. The machine history can be accessed to list recent terminated machines.

In addition to the state name, a colored icon may be displayed:

	When the machine is not idle or running, a progress icon indicator is displayed.
	The machine has an error, you can move the mouse over and a tooltip will display the exact reason.

You can perform the following actions. Some actions are specific to one machine and some others can be applied to multiple machines, in this case you can toggle the selection using the checkboxes.

	Terminate machines. The machines will typically auto-terminate based on the idle shutdown parameter. However, it may be useful to terminate the machines manually. If a compute server has one or more distributed workers, they will be terminated as well.

Tabs below the table provide details on the selected machine, including the machine description, its configuration as well as its actual status.

Machine Metrics

The machines are reporting the current CPU usage, current memory consumption and maximum memory consumption.

History

The history panel lists recent terminated machines. The list enables you to sort and search for machines.

The last machine state is also displayed. If the last state was `idle`, then the machine was terminated because the idle shutdown time limit was reached. If the last state was `shutting down`, then the machine was terminated explicitly using the Instant Cloud Manager or the REST API.

The table also displays the maximum memory consumption by the machine. This is important information so that you can adjust the machine configuration accordingly.

Tabs below the table provide details on the selected machine, including the machine description, its configuration as well as its last known status.

2.1.5 Billing

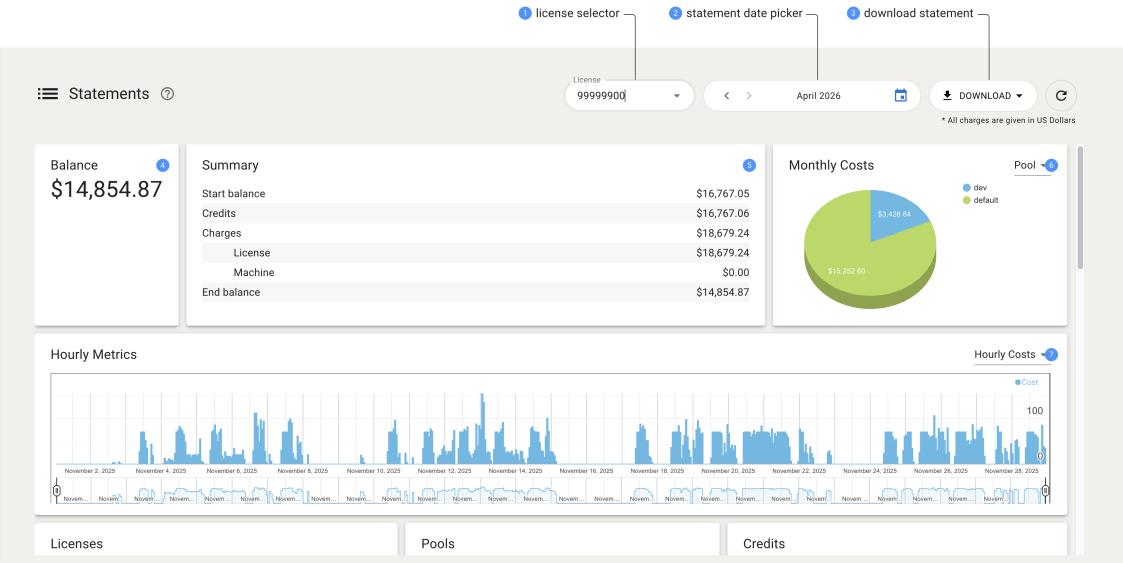
Overview

The main responsibility of the billing system is to generate monthly statements for all active license holders. The billing module presents an interactive user interface with the statement details such as machine usage, charges, transactions, balances and license related information.

Statements

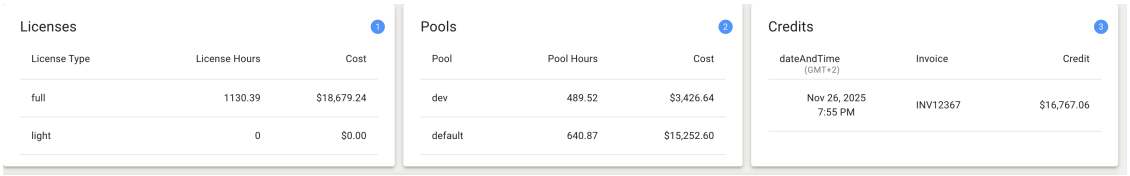
Billing statements are generated in monthly basis for each active license. In order to keep statements up-to-date, the billing manager checks for new events initiated by the license every two minutes or less. On the first day of every month the statements get closed for the prior month.

The statement panel consists of the following sections as shown in the screenshot below:



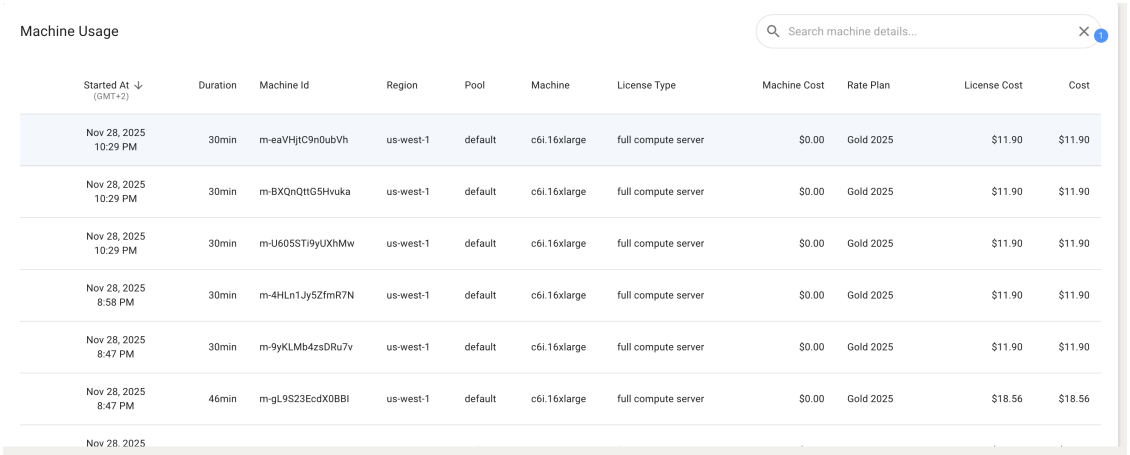
1	**License selector** If you have more than one license in your account, a statement is generated for each of the licenses, thus the license selector allows to switch between different license statements.
2	**Statement date picker** Allows switching between statements by month and year.
3	**Download statement** Allows downloading the statement in CSV and HTML format.
4	**Balance** Shows the ending balance for a particular license in dollar amount.
5	**Summary** Shows Starting balance, total credits, license and machine charges, and ending balance for a particular license.
6	**Monthly costs** The pie chart by default shows monthly cost by pool, however you can change it to get the costs by machine type or by region.
7	**Hourly metrics** The default graph shows hourly cost, however you can change it to see the cost by pool, the number of machines per hour, the costs by machine type or the costs per region.

The next section of a statement contains the license charges, the pool charges and the credits for that month.



1	**Licenses** Shows cost and license hours by license types.
2	**Pools** Shows cost and license hours by pools.
3	**Credits** Shows all the transaction credits for that month.

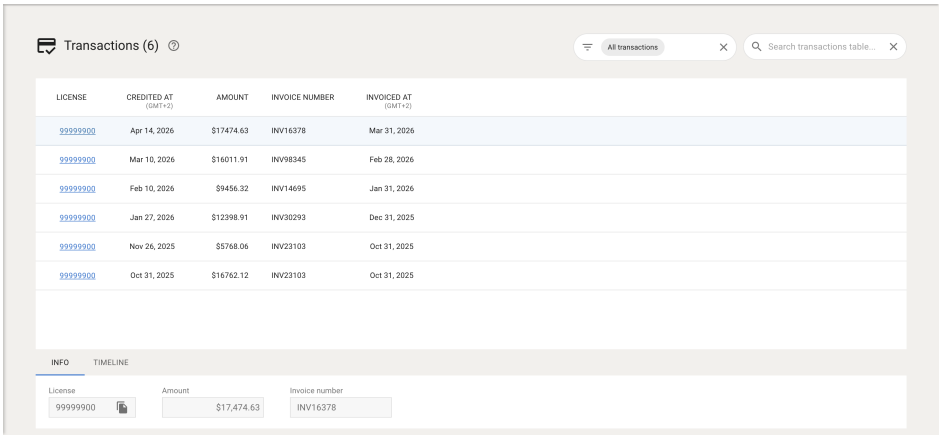
The last section includes the details of the machine usage.



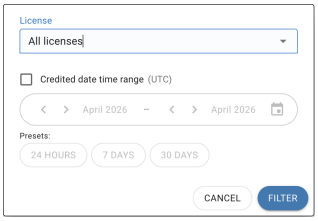
1	**Machine Usage** Shows machine usage details such as the machine start time, duration, id, region, pool, type, license type, cost, license cost.

Transactions

Transactions panel lists all the transactions for all licenses belonging to a user. The table consists of the following fields total amount credited, credited date, invoice number and date.



Transaction Filters



The default filter lists every transaction made for all licenses. Below are listed all transaction filters that you can use.

- **License**

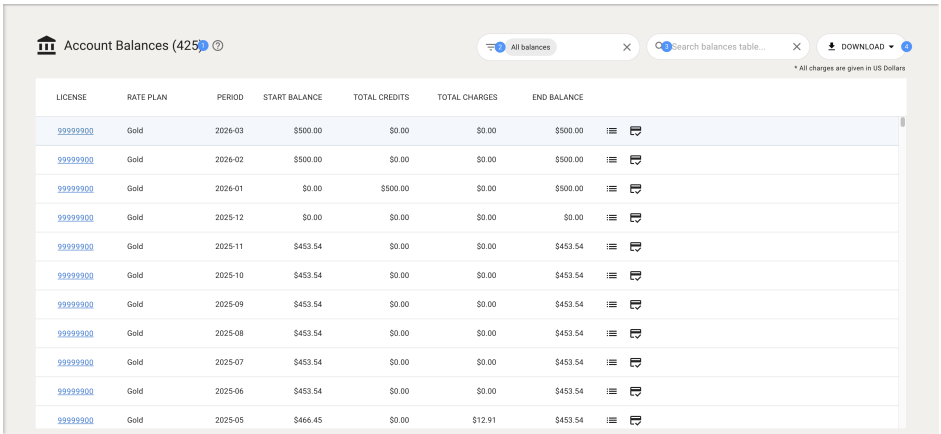
  If a user has multiple licenses this allows to filter on a license, this way you only see transactions for a specific license.

- **Date time range**

  This allows you to filter on a date range, this way you see transactions made between two different months or use the preset filters like last 24 hours, 7 or 30 days.

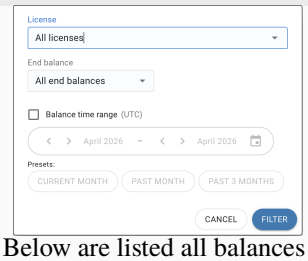
Account Balances

With the account balance table you can monitor the current and past balances for all your licenses.



1 **Number of rows in the results table.**

2 **Balance Filters**



The default filter lists all the balances for every license. Below are listed all balances filters that you can use.

- **License**: If a user has multiple licenses this allows to filter on a license, this way you only see transactions for a specific license.
- **End Balance**: This allows you to filter based on positive or negative ending balance.
- **Date time range**: This allows you to filter on a date range, this way you see balances between two different months or use the preset filters for current month, past month and past 3 months.

3 **Search balances**

The Search box enables users to further refine the results by specifying an arbitrary search string. In order for a balance to be displayed, the search string must be present in at least one of the columns in the table for that balance.

```
It is important to note that filtering with the Filter box is usually applied
server side, while searching with the Search box is always performed client side.
That means that the searching is done on results that were already filtered by the
server, so it is preferable to use the Filter box first and then use the Search
box to refine the results. Using the Search box as an alternative to the Filter␣
↪box
could result in missing items.
```

4 **Download Balances**

We also allow downloading the balances into a CSV file from the DOWNLOAD button in the upper right section.

5 **Balance results**

Balance results table allows listing of information about statement balances for each month. It includes information about start balance, total credits, total charges, ending balance for a license. The two icons at the end of each row are shortcuts which allow you to jump to the statement or to list the transactions of that month.

CHAPTER

THREE

INSTANT CLOUD REST API

The Instant Cloud REST API lets you build custom solutions or frameworks when you need to create, start and stop machines or pools automatically. The API follows standard REST principles and can be used in various languages and tools (Java, Python, Node.js, curl…)

3.1 REST API v2

The API v2 provides a set of endpoints to perform the following actions:

- list licenses,
- list, launch and terminate machines,
- list, create and delete machine pools,
- launch and terminate machine pools.

3.1.1 API keys

In order to use the API, you will need an API key consisting of an **API access ID** and a **secret key**. Please refer to the API key documentation to review the steps to retrieve the keys in the Instant Cloud Manager.

If a key has been created for a specific license, only the machines and pools related to this license will be accessible. If it is a global account key, all the information related to the account will be accessible.

The secret key is a like a password and it should never be shared with others and should not be sent by emails.

The access ID and the secret key must be passed with each request in the following HTTP headers:

- `X-GUROBI-ACCESS-ID`
- `X-GUROBI-SECRET-KEY`

3.1.2 Base URL

The API can be accessed at the following base URL:

```
https://cloud.gurobi.com/api/v2
```

HTTPS must be used to ensure that the communication is encrypted. Using HTTP will return an error.

3.1.3 Reference Documentation

The reference documentation provides all the details about the API by listing the endpoints and specifying the parameters as well as the input and output data. It is also an interactive API playground to let you try out each endpoint.