The TIMES Cloud Service

Webinar
20th May 2022
Open Energy System Modelling

Frederik Fiand
ffiand@gams.com

GAMS

Who are we?

What do we offer?

Why are we here?

Increasing Openness through the TIMES Cloud Service!

GAMS Who are we?

History

- **G**eneral **A**lgebraic **M**odeling **S**ystem
- Roots: World Bank, 1976
- Went commercial in 1987
- Locations: USA, Germany

Our Competencies & Activities

- Software system to model and solve optimization problems
- **First** algebraic modeling software
- **Customer support** for our software and all bundled solver products
- **Technical Consulting** services
- **Projects** with extensive partner network

Our Customers

- **Worldwide** installations
- Academic and commercial
- Energy, economics, logistics, operations research, and many more
- Big and small companies, government agencies

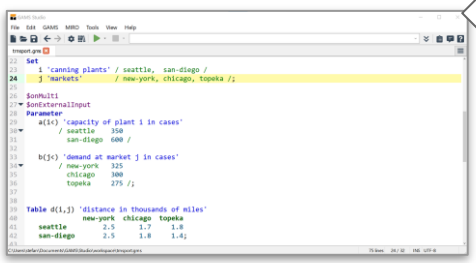
Our Staff

- 29 full time + students (US & EU)
- PhD level modeling and optimization experts with startup mentality
- Experienced management, generational change in 2015

GAMS What do we offer?

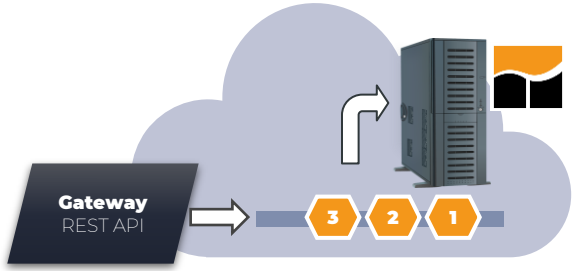
Modeling Platform

- **Platform Independent Algebraic Modeling Language for Domain Experts**
- **Commercial and Academic Solvers** (packaged)
- **APIs for major programming languages** (C++, Java, Python, Matlab and more)



Graphical UI Generator

- **Turns models into web applications for end users**
- **Extendable with custom code**
- **Local or Server Installation**



Deployment Solution

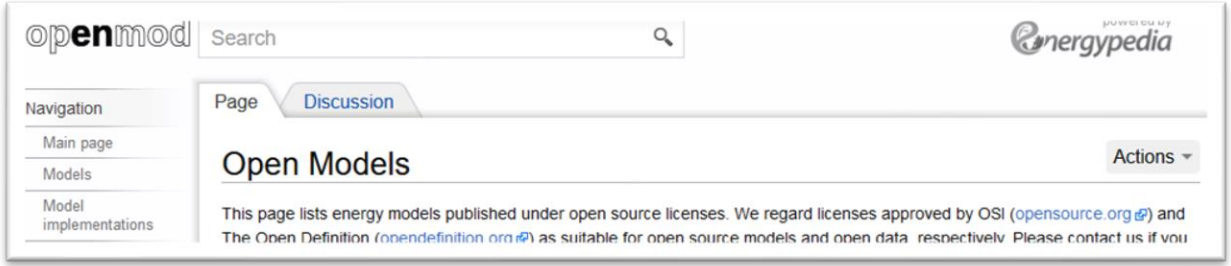
- **Solves GAMS models on centralized resources** (on-prem or cloud)
- **REST API** (user & job management)
- **GAMS job scheduling & Load balancing**

GAMS Why are we here?

Broad range of application areas

Agricultural Economics	Applied General Equilibrium
Chemical Engineering	Economic Development
Econometrics	Energy
Environmental Economics	Engineering
Finance	Forestry
International Trade	Logistics
Macro Economics	Military
Management Science/OR	Mathematics
Micro Economics	Physics

GAMS Why are we here?



Overview of models by type, software, implementation and processing

Model	Model class	Modelling software	Processing software
Balmorel	GAMS	GAMS	
EOLES elec	Electricity System Model	GAMS	
EOLES elecRES	Electricity System Model	GAMS	
Medea	Austrian and German electricity market	GAMS	Python
ELMOD	German and European Electricity Market	GAMS	
EMMA	Power market model	GAMS	
StELMOD	Optimization	GAMS	MS Excel
TIMES	Local National Regional Global models developed using TIMES	GAMS	EXCEL, VEDA, ANSWER

13 models listed on openmod are implemented in GAMS.

Source: https://wiki.openmod-initiative.org/wiki/Open_Models

TIMES Cloud Service Motivation

2020

- TIMES source code published on GitHub under an open source license
- GAMS license required
- Solver license required
- VEDA license required (front-end to TIMES)

Challenge: Increase the openness of the TIMES toolbox such that …

- … the entry barrier for new users is lowered
- … occasional users do not need to worry about licensing
- … the dissemination of models/results becomes license independent

TIMES Cloud Service / TIMES MIRO APP Proposals

→ Submitted joint ETSAP R&D project proposals in 12/2020

The TIMES Cloud Service

- allows to solve TIMES models “in the cloud”, license centrally administered by ETSAP
- is convenient to access (username/password, various clients)
- builds on state-of-the-art cloud technology (GAMS Engine)

The TIMES MIRO APP

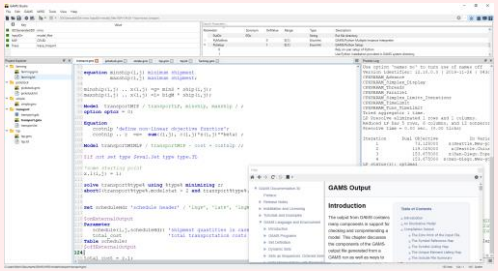
- is a free to use open source front end to TIMES
- builds on the (open source) graphical UI Generator GAMS MIRO

→ Both projects unfold synergies

TIMES
Cloud Service

Cloud Service Only

GAMS Studio

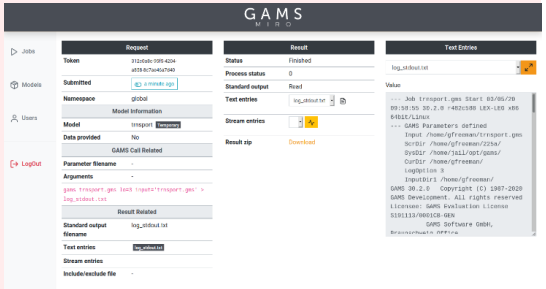


Other APIs
(Python, Java, etc.)

```
configuration = openapi_client.Configuration(
    host="http://localhost",
    username='admin',
    password='admin'
)

# Enter a context with an instance of the API client
with openapi_client.ApiClient(configuration) as api_client:
    # Create an instance of the API class
    job_api_instance = openapi_client.JobsApi(api_client)

    model = 'clad'  # str | Name of the main .gms file
    namespace = 'global'  # str | Namespace containing(or will contain) the model
    # str | Name of the file that captures stdout (default to 'log_stdout.txt')
```

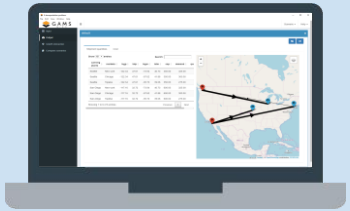
Veda

Web User Interface



MIRO Desktop

Everything local

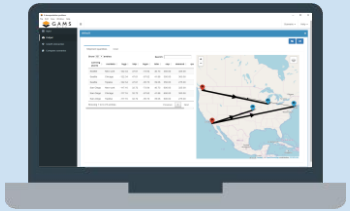


- GAMS & MIRO installed locally
- Synchronous job execution

MIRO APP only

MIRO Desktop

Everything local



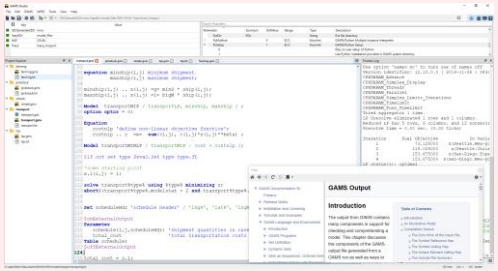
- GAMS & MIRO installed locally
- Synchronous job execution

MIRO APP only

TIMES Cloud Service

Cloud Service Only

GAMS Studio

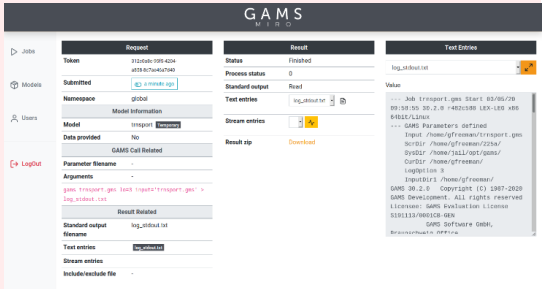


Other APIs (Python, Java, etc.)

```
configuration = openapi_client.Configuration(
    host="http://localhost",
    username='admin',
    password='admin'
)

# Enter a context with an instance of the API client
with openapi_client.ApiClient(configuration) as api_client:
    # Create an instance of the API class
    job_api_instance = openapi_client.JobsApi(api_client)

    model = 'clad'  # str | Name of the main .gms file
    namespace = 'global'  # str | Namespace containing(or will contain) the model
    # str | Name of the file that captures stdout (default to 'log_stdout.txt')
```

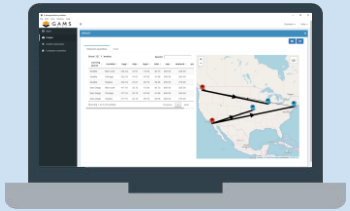
Veda

Web User Interface



MIRO Desktop

Everything local

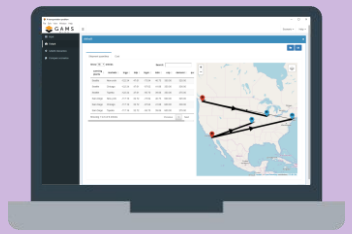


- GAMS & MIRO installed locally
- Synchronous job execution

MIRO APP only

MIRO Desktop

Boosted by GAMS Engine

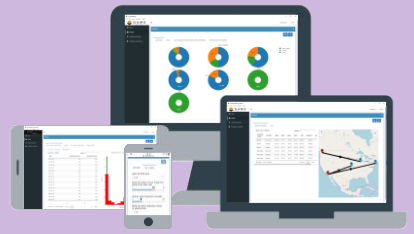


- MIRO installed locally
- Synchronous and asynchronous job execution

MIRO Server

Everything on a server

→ No software installed locally



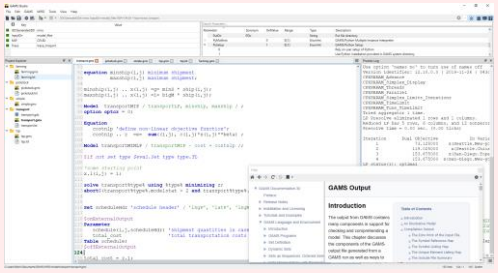
MIRO application

Both

TIMES Cloud Service

Cloud Service Only

GAMS Studio

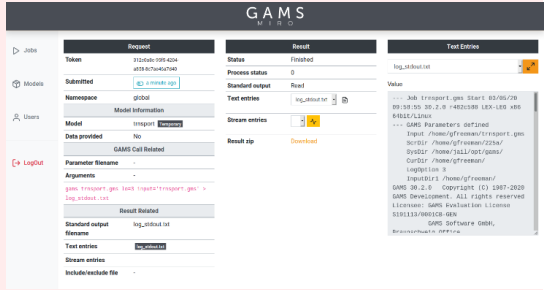


Other APIs (Python, Java, etc.)

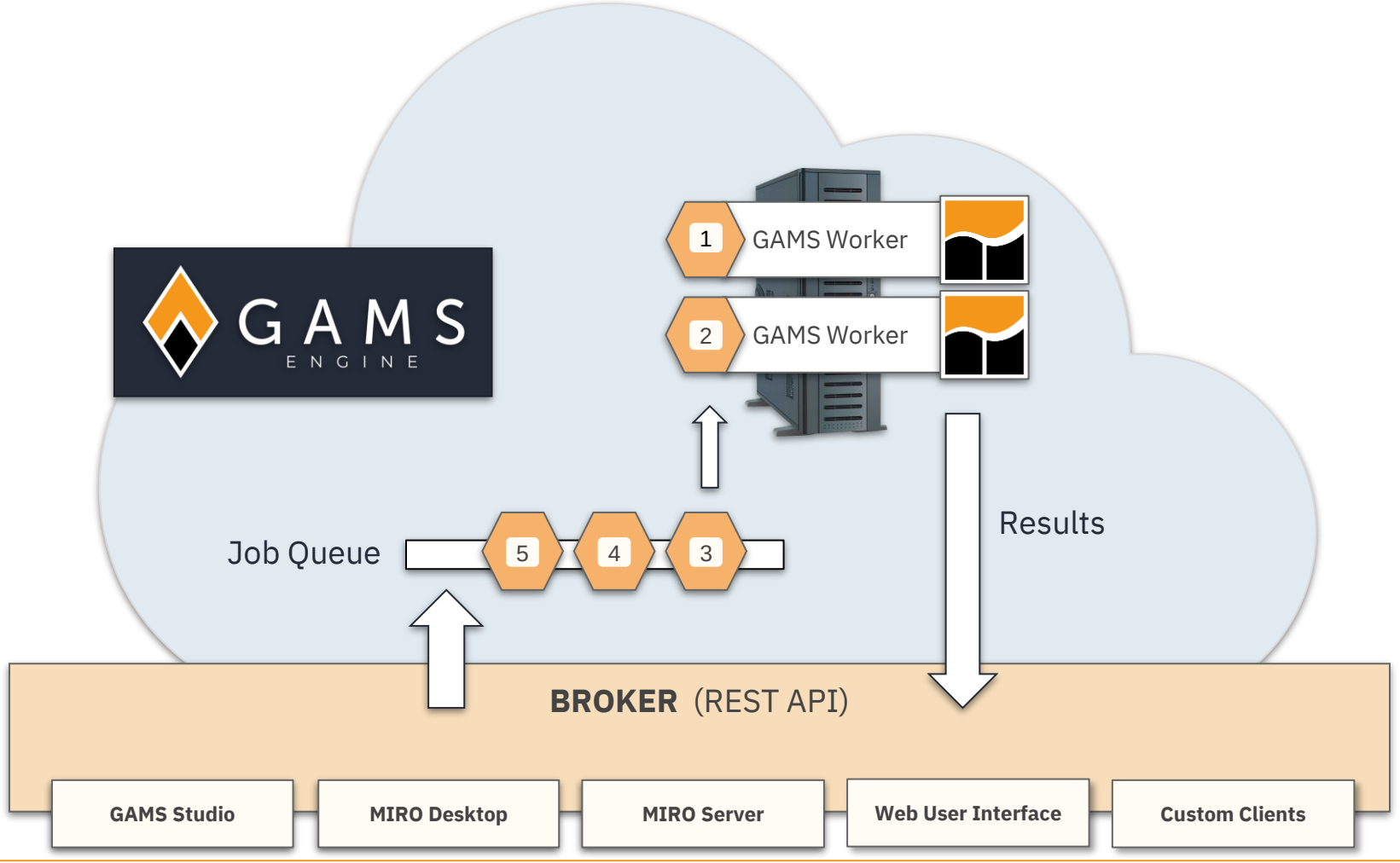
```
configuration = openapi_client.Configuration(
    host="http://localhost",
    username='admin',
    password='admin'
)

# Enter a context with an instance of the API client
with openapi_client.ApiClient(configuration) as api_client:
    # Create an instance of the API class
    job_api_instance = openapi_client.JobsApi(api_client)

    model = 'clad'  # str | Name of the main .gms file
    namespace = 'global'  # str | Namespace containing(or will contain) the model
    # str | Name of the file that captures stdout (default to 'log_stdout.txt')
```

Veda

Web User Interface

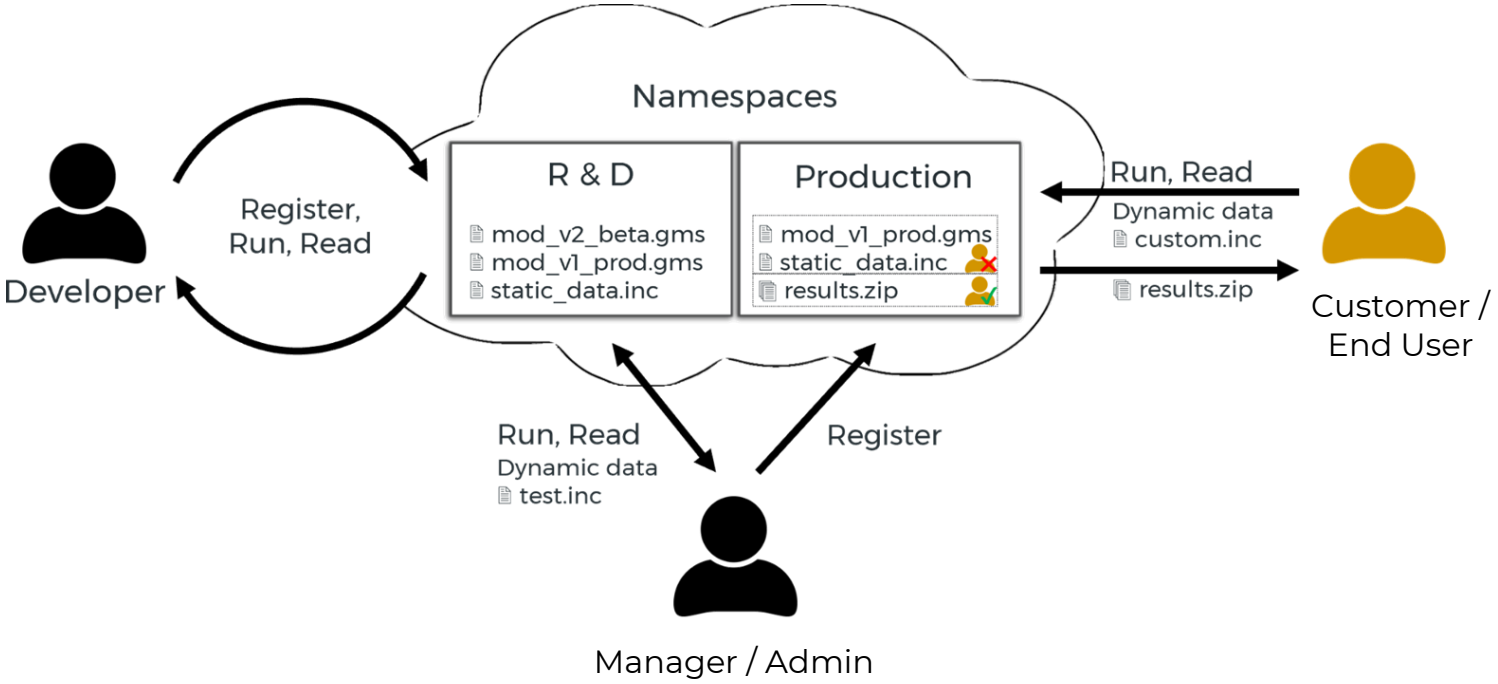


GAMS Engine The Technology behind the TIMES Cloud Service



GAMS Engine Summary of Features

User Management



GAMS Engine Summary of Features

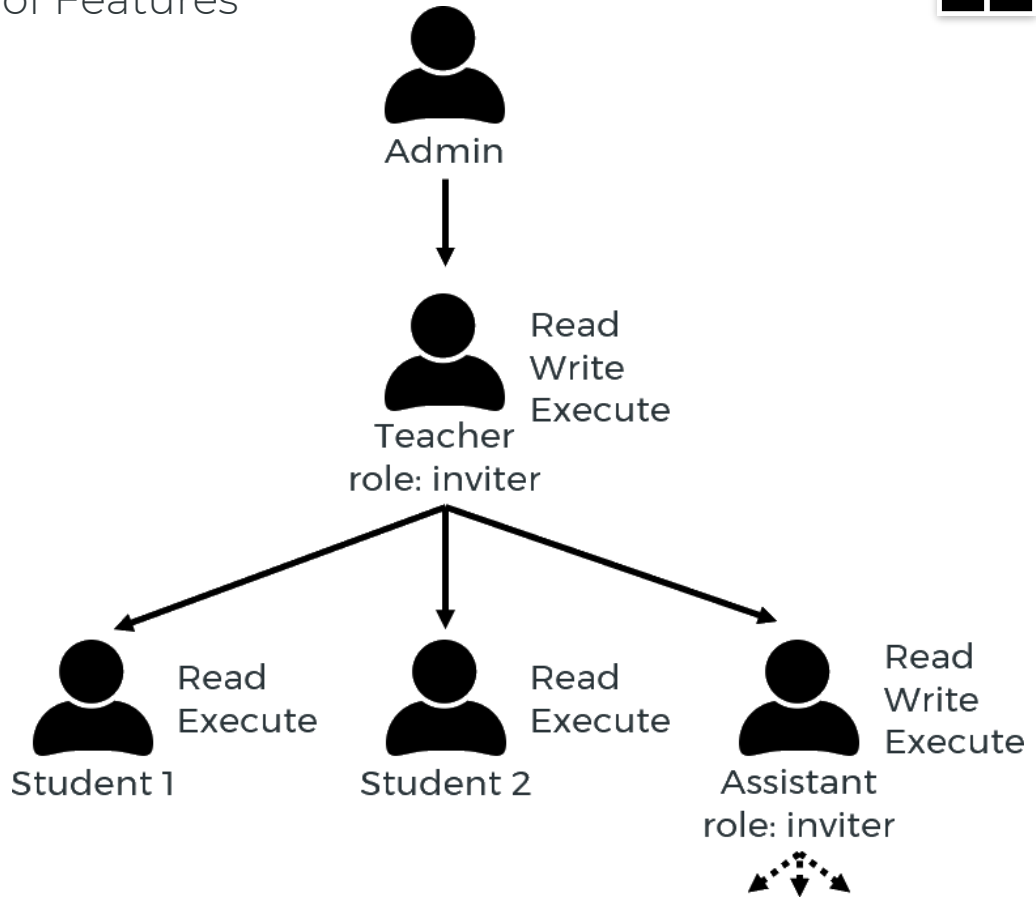
User Management

Hierarchical user tree(s):

Root node are called “admins”

Leaf nodes are called “users”

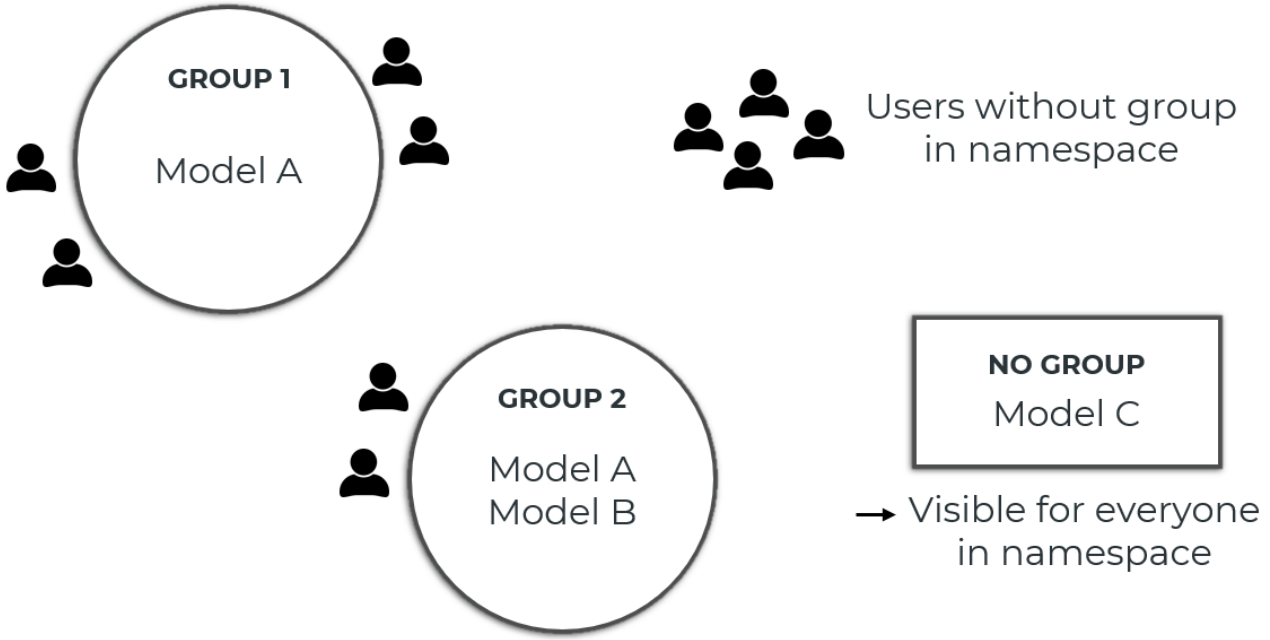
Other nodes are called “inviters”



GAMS Engine Summary of Features

User groups

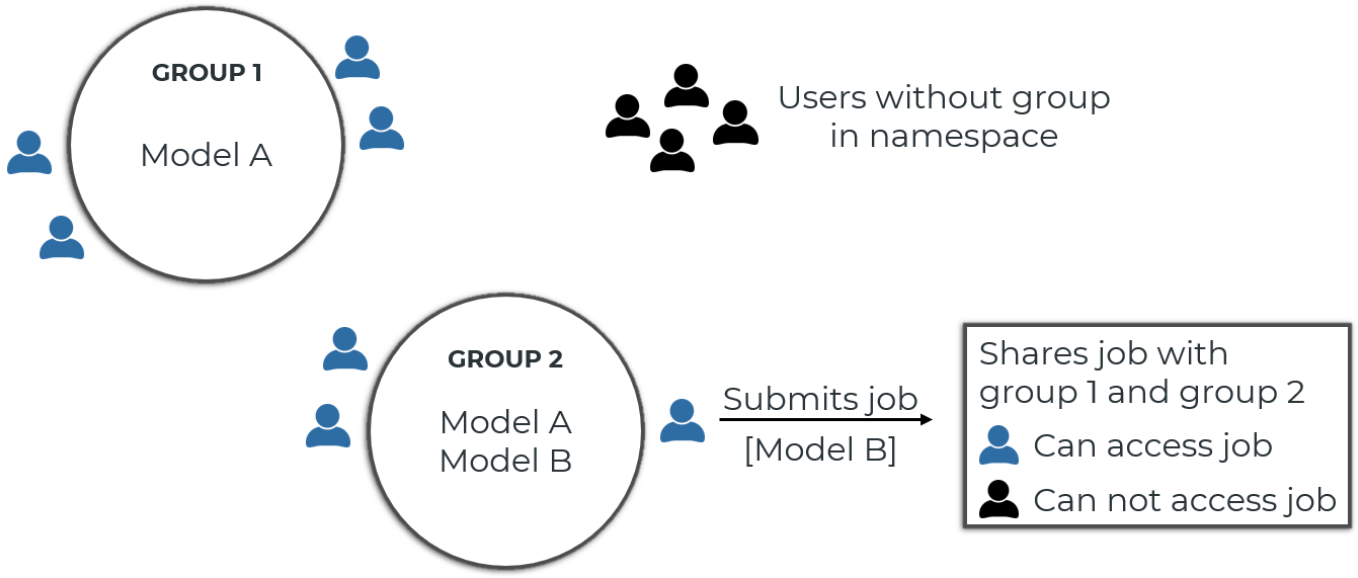
Models and jobs can be shared with groups



GAMS Engine Summary of Features

User groups

Models and **jobs** can be shared with groups



User Management

Users can be assigned **quotas**:

- Volume quota to limit total solve time user is allowed to use
- Disk quota to limit how much disk space user is allowed to use
- Parallel quota to limit how many jobs user is allowed to use in parallel

Quotas are inherited, but can be further restricted

Resource requests

- Resources (CPU/RAM/ephemeral disk) a job is allowed to use can be specified: Engine makes sure to schedule the job on a node with sufficient resources
- Resources can be bundled as so-called “instances”: a label is assigned to a set of resource requests (CPU/RAM/ephemeral disk)
- Instances that users are allowed to use can be restricted
- Allowed instances are inherited, but can be further restricted

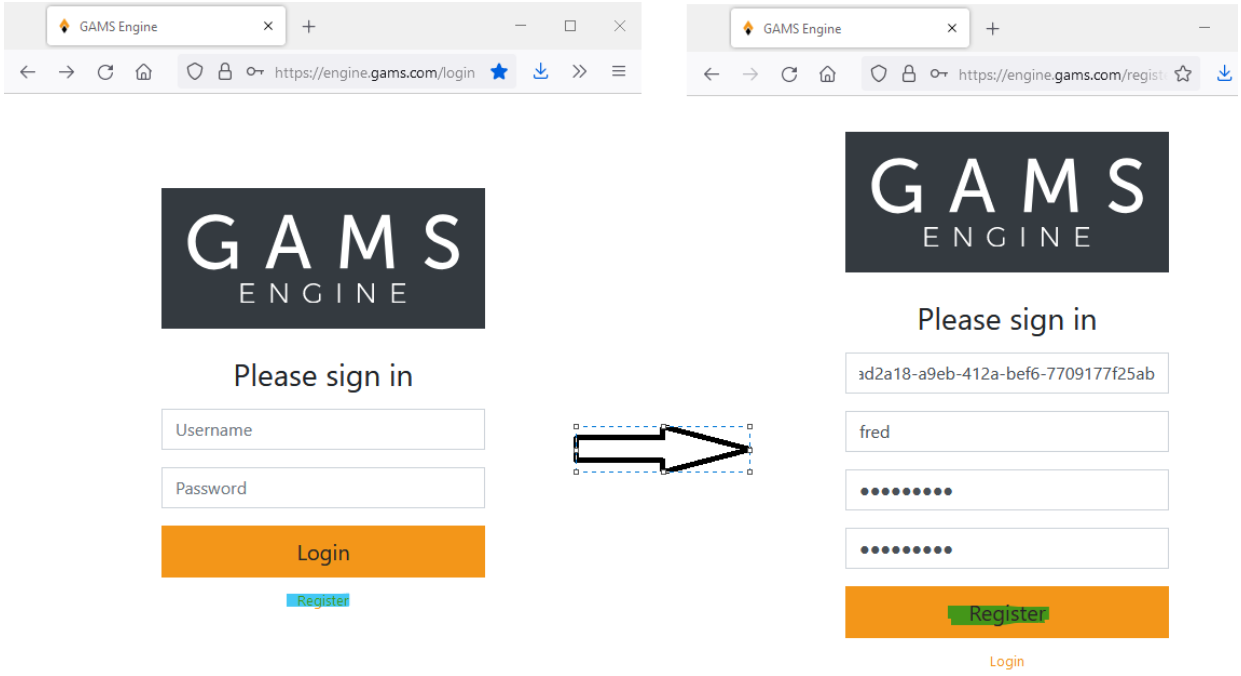
GAMS Engine Summary of Features

- System to run potentially long-running and computationally expensive tasks on centralized compute resources.
- The only interface to communicate with GAMS Engine is REST API
- Various clients
- User management system
- User quotas
- Resource requests
- User groups
- Hypercube Jobs
- Job dependencies
- Webhooks
- High Security standards
- ...

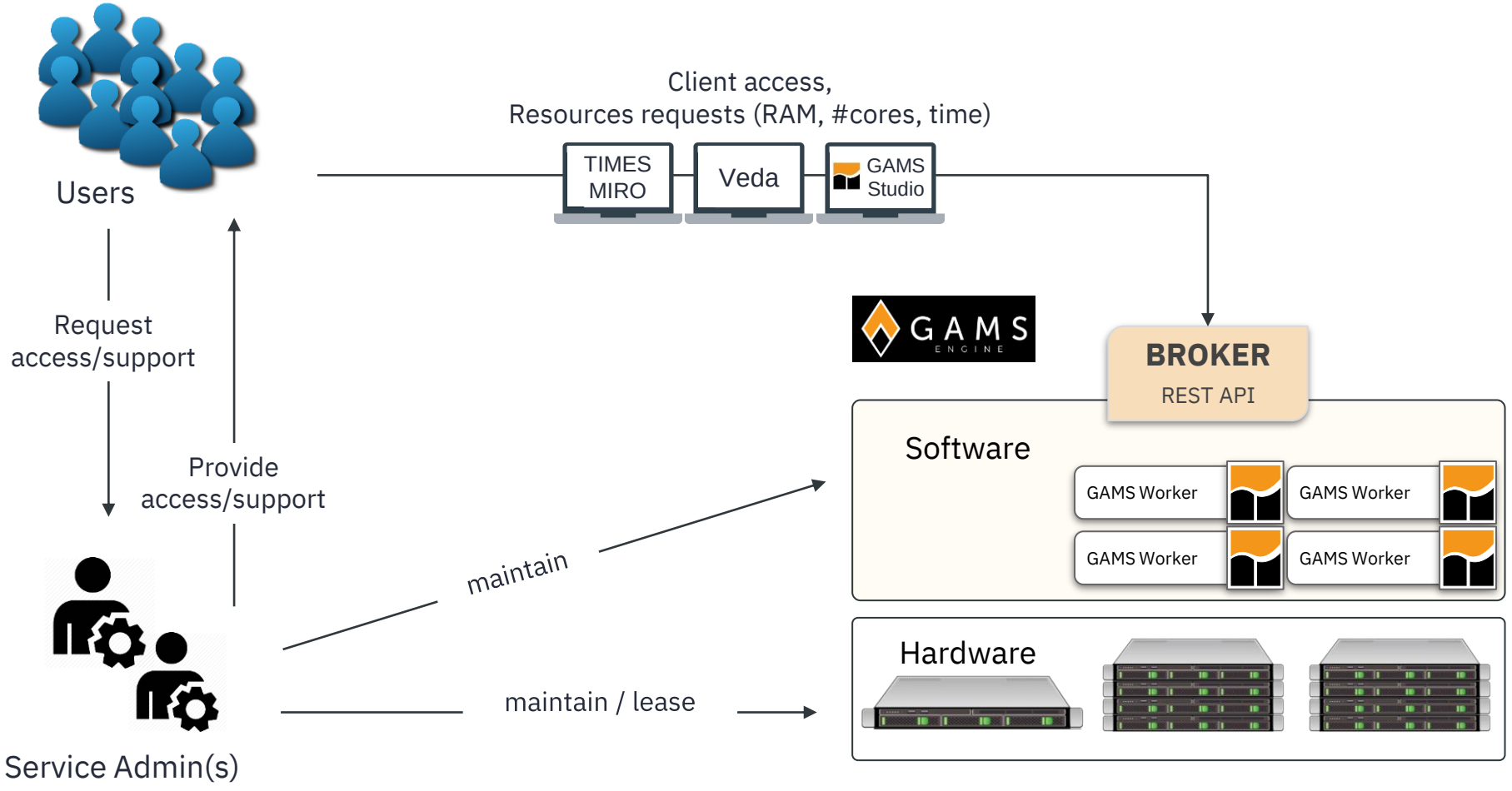
→ Read more

How to get access?

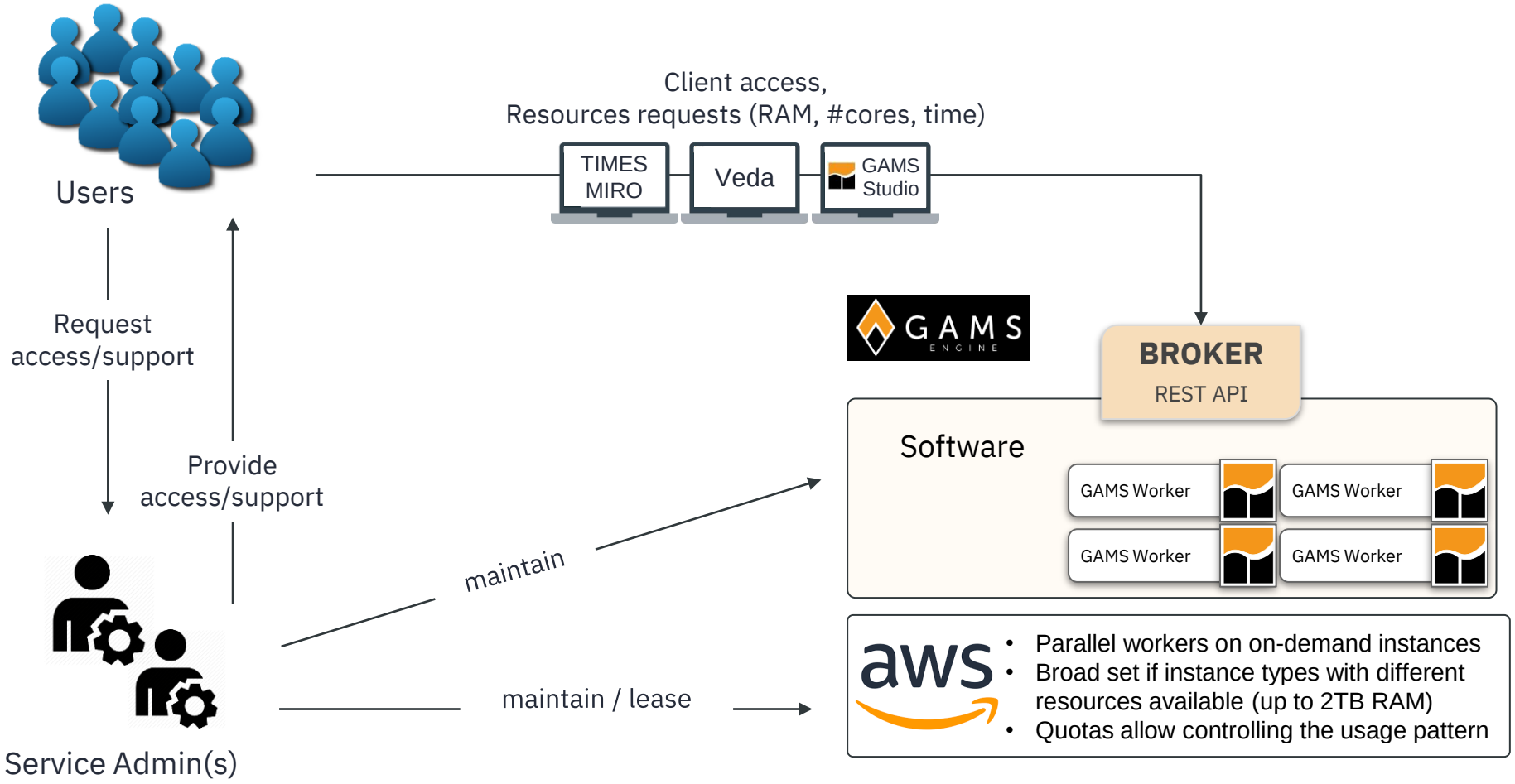
1. Request registration code by sending an email to timescloud@etsap.org (currently restricted to ETSAP members)
2. Go to the TIMES cloud login page https://engine.gams.com/login



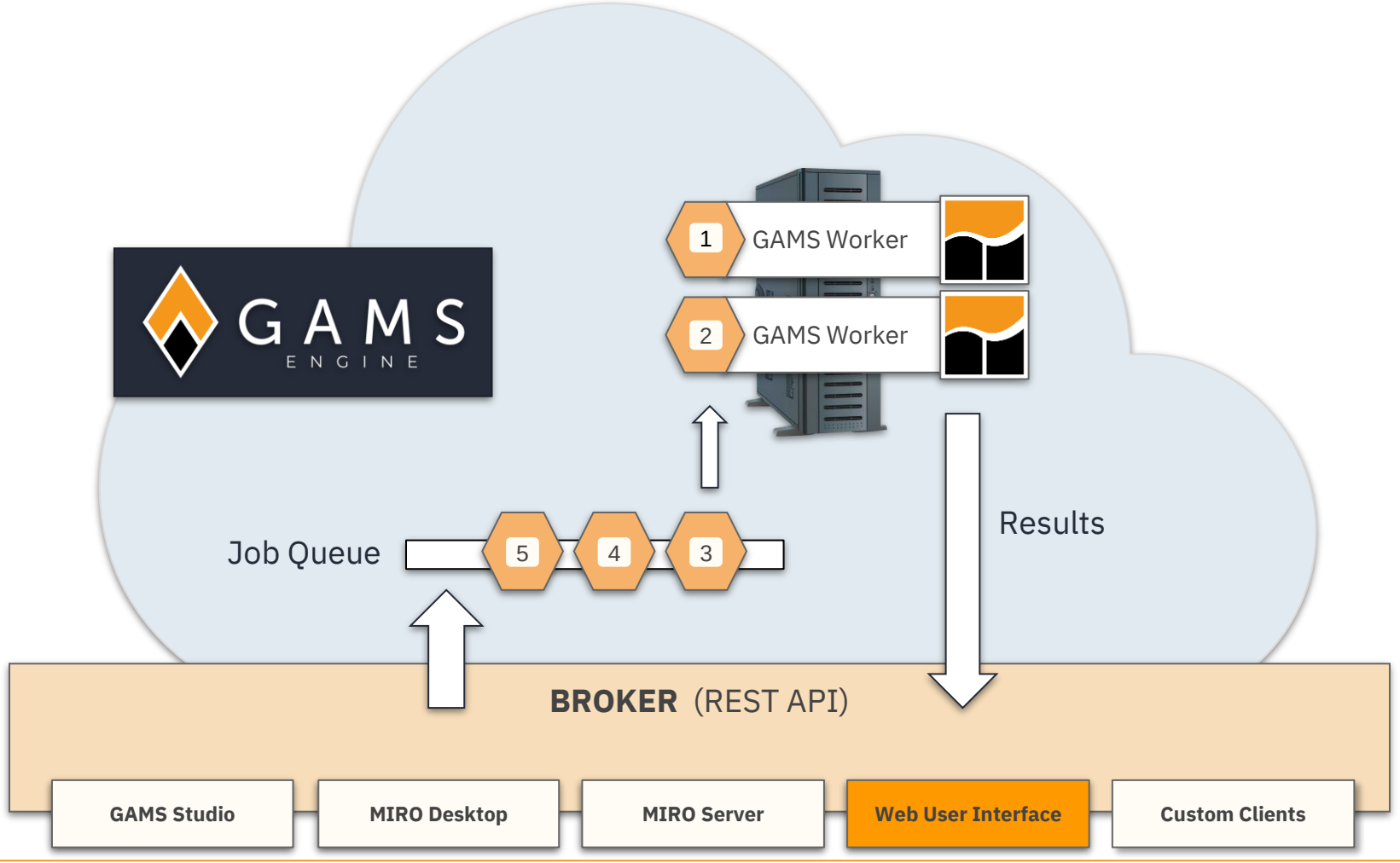
TIMES Cloud Service Roles & Components



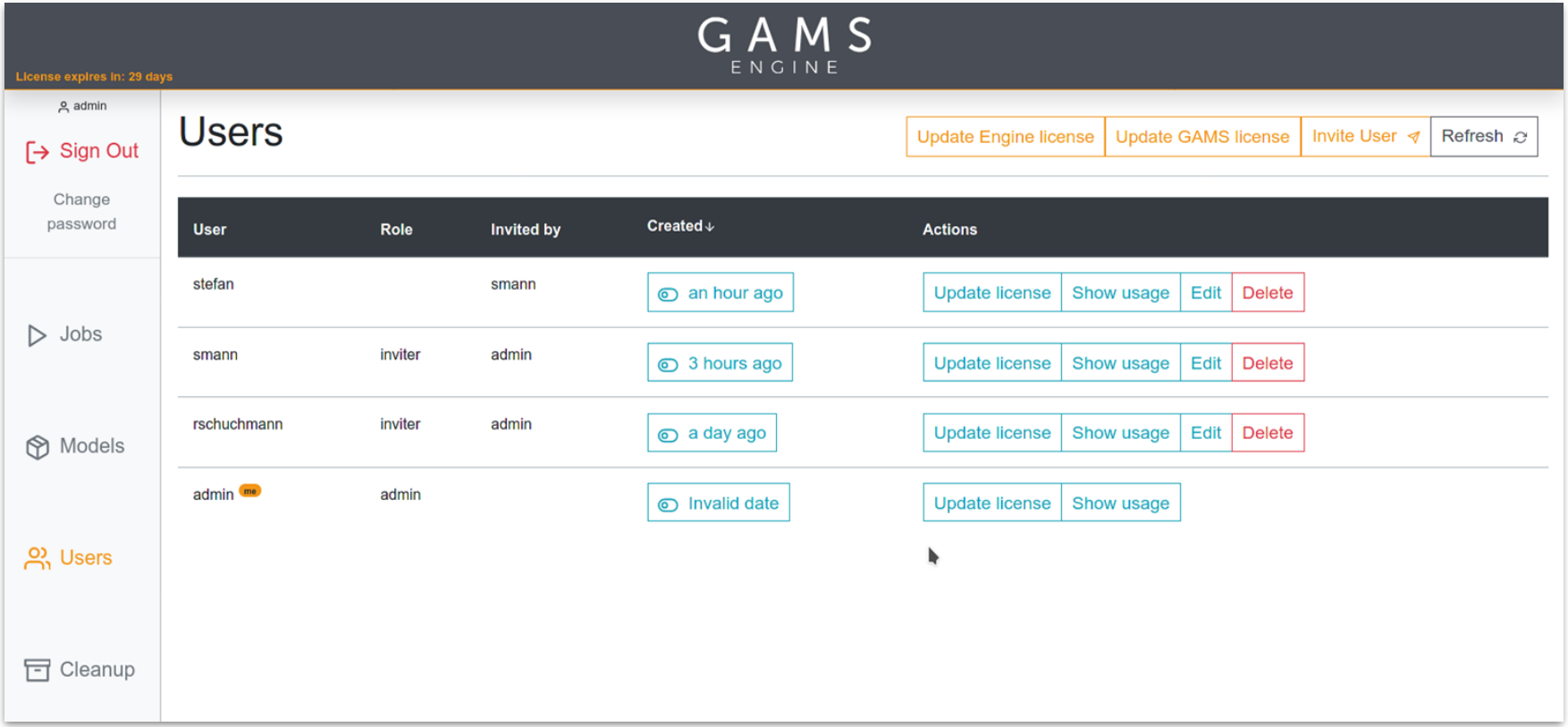
TIMES Cloud Service Roles & Components



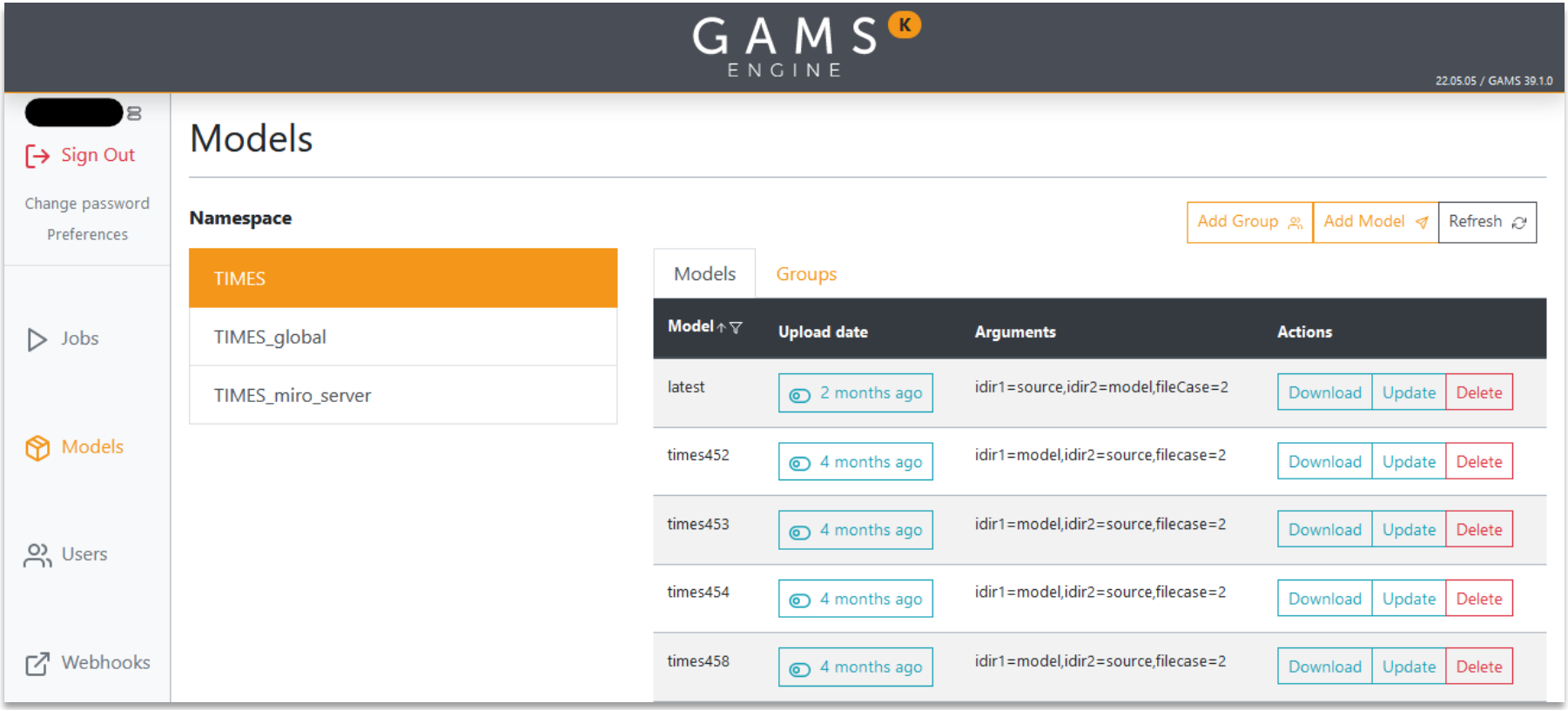
GAMS Engine Clients



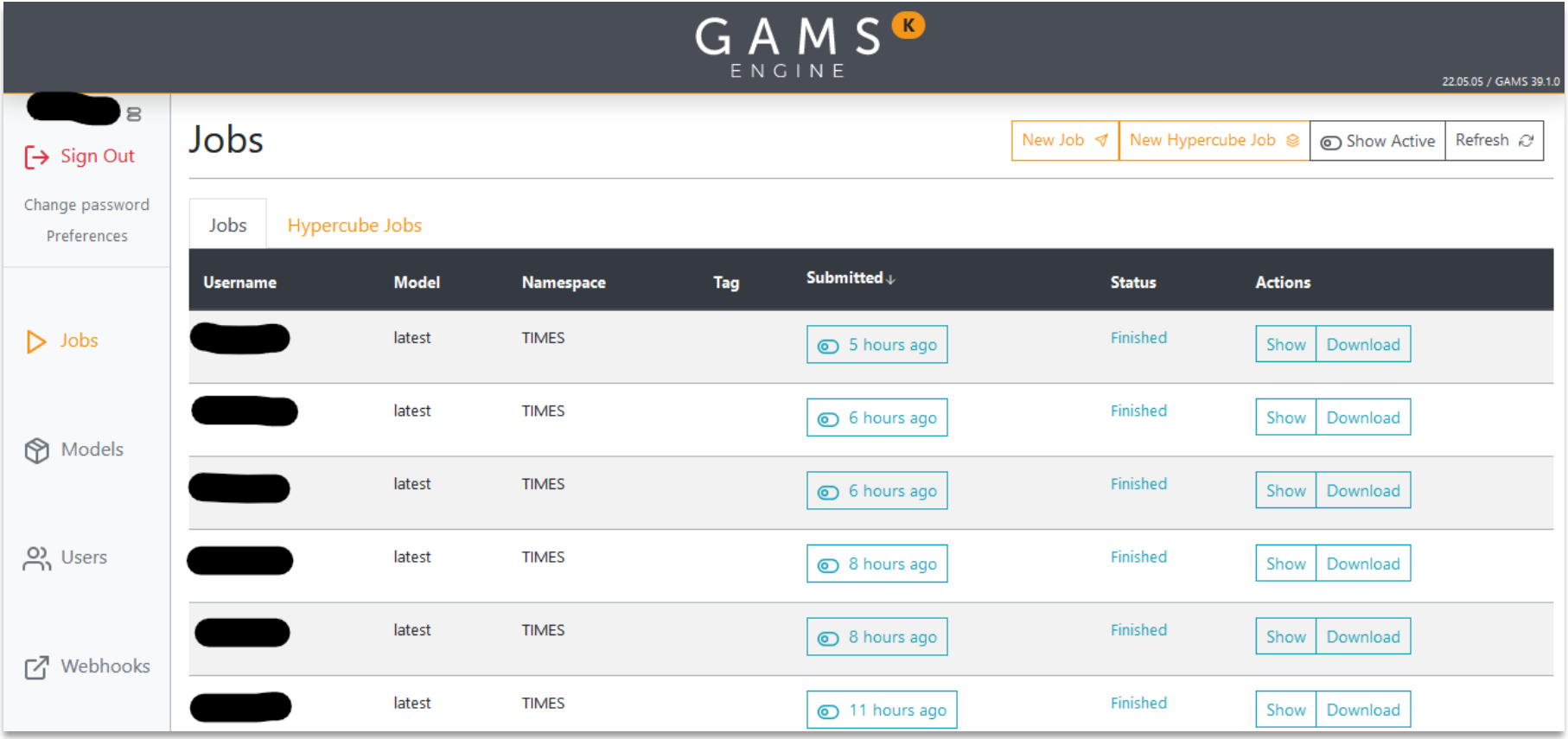
GAMS Engine UI User Management



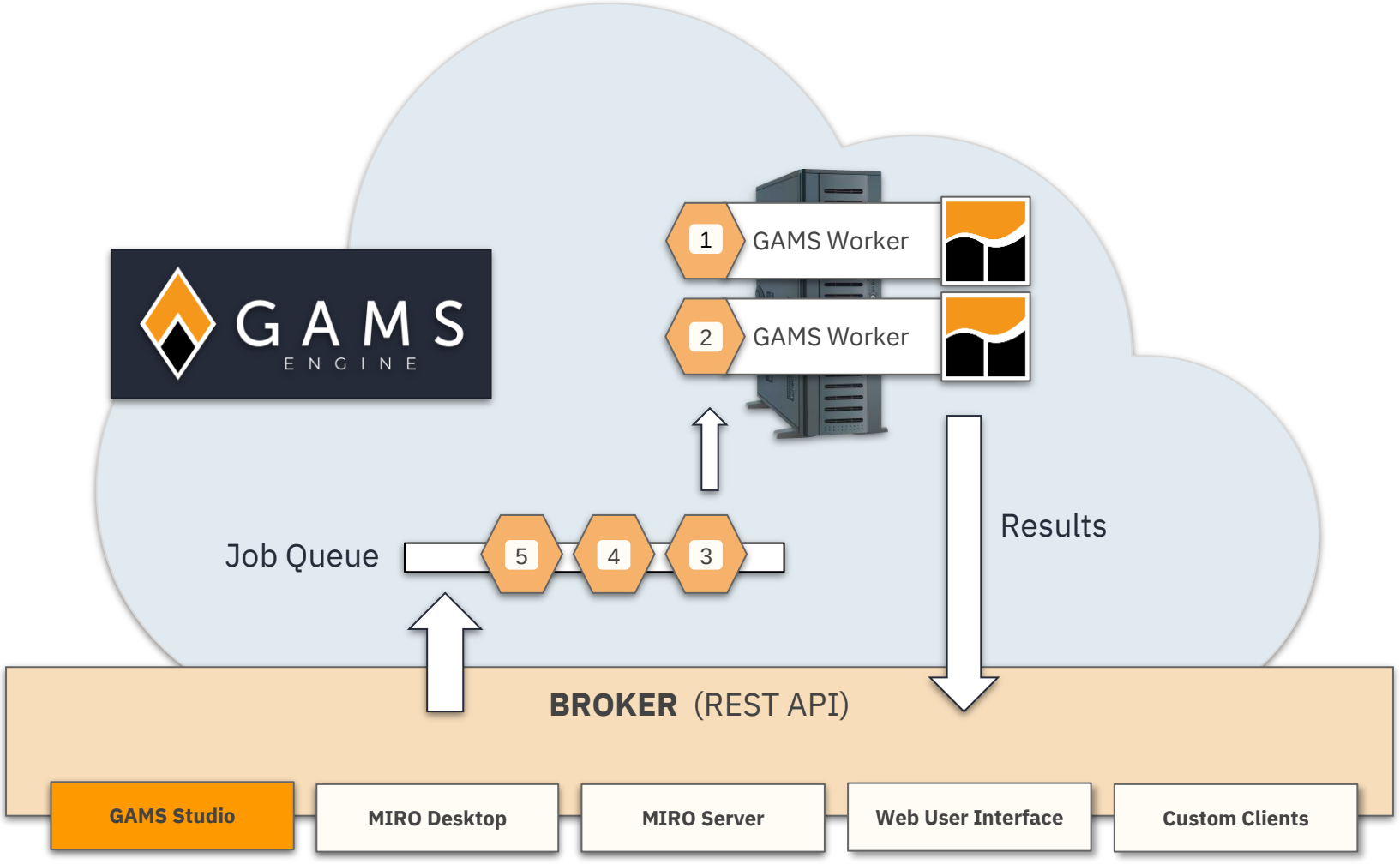
GAMS Engine UI Registered Models



GAMS Engine UI Jobs

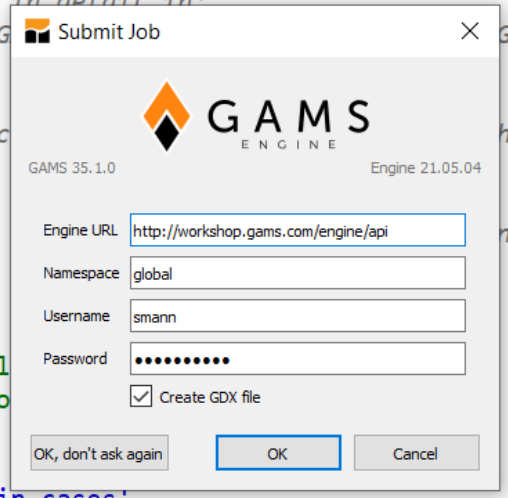


GAMS Engine Clients

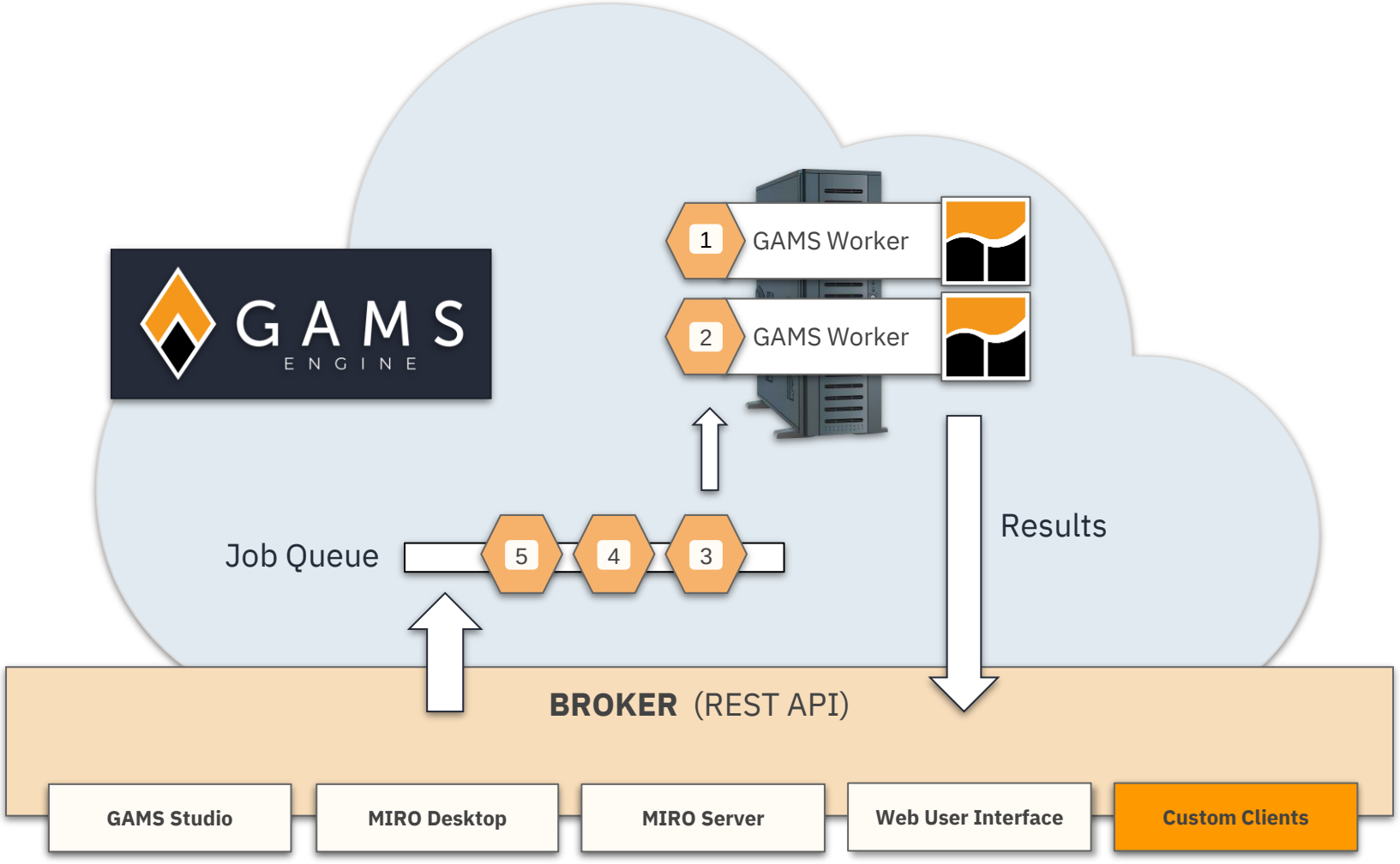


GAMS Engine Clients - GAMS Studio

```
tzig, G B, Chapter 3.3. In Linear Programming and Extensions.
nceton University Press, Princeton, New Jersey, 1963.

s formulation is described in detail in:
enthal, R E, Chapter 2: A G                                    Guide.
 Scientific Press, Redwood

 line numbers will not matc                                    hese
ments.

words: linear programming,                                     ng
fText


i 'canning plants' / seattl
j 'markets'        / new-yo

ameter
a(i) 'capacity of plant i in cases'
     / seattle    350
       san-diego  600 /

b(j) 'demand at market j in cases'
     / new-york   325
```



GAMS Engine Clients



GAMS Engine Custom Clients

OpenAPI version 2.0

- language agnostic interface
- autogeneration for serveral languages (Python, Java, C, C#, C++, Node.js, Ruby…)

REST API

- only point of contact for clients
- job scheduling, queuing, GAMS calls are all transparently handled by Engine

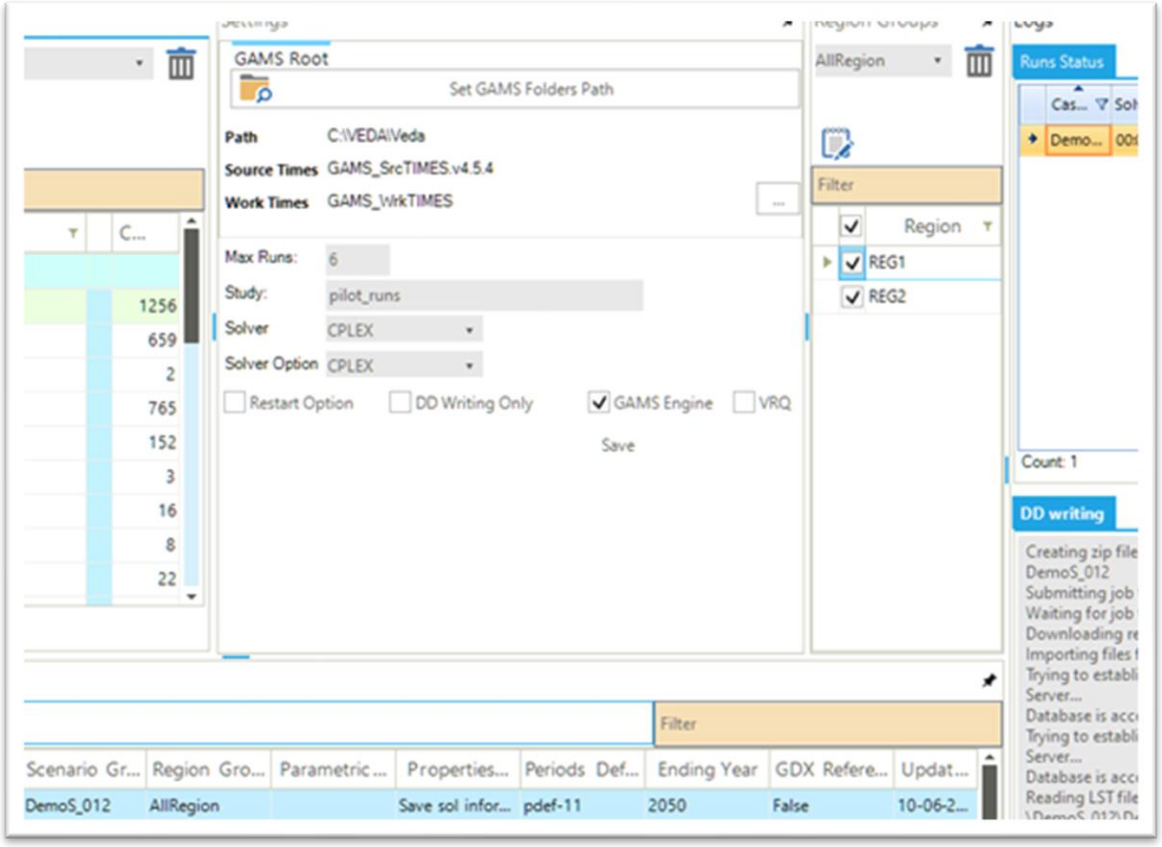
Job submission

```
create_job_response = job_api_instance.create_job(model,
                                                  namespace,
                                                  model_data=model_data,
                                                  arguments=arguments)
```

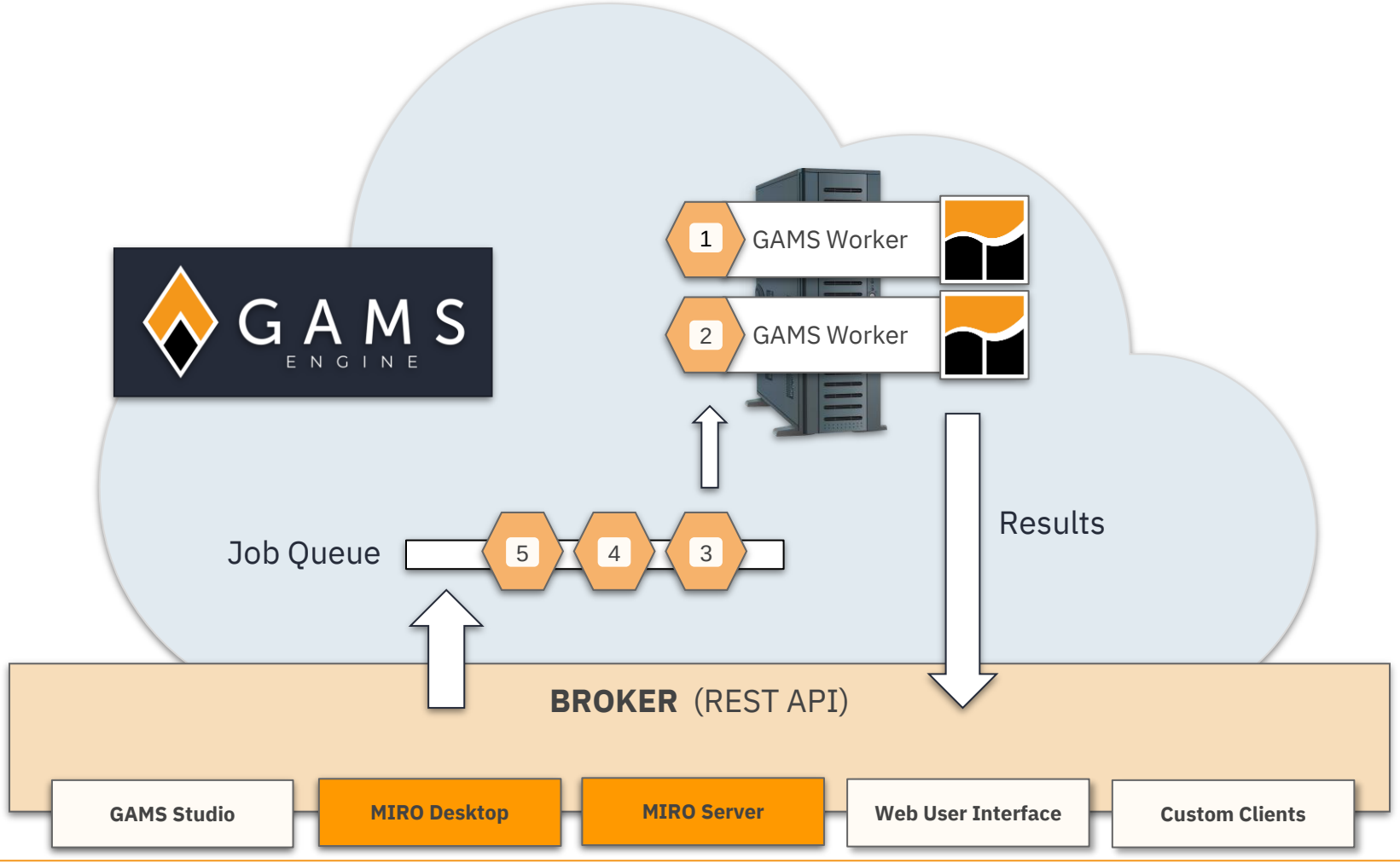
Downloading results

```
token = create_job_response.token
result_zip = job_api_instance.get_job_zip(token)
```

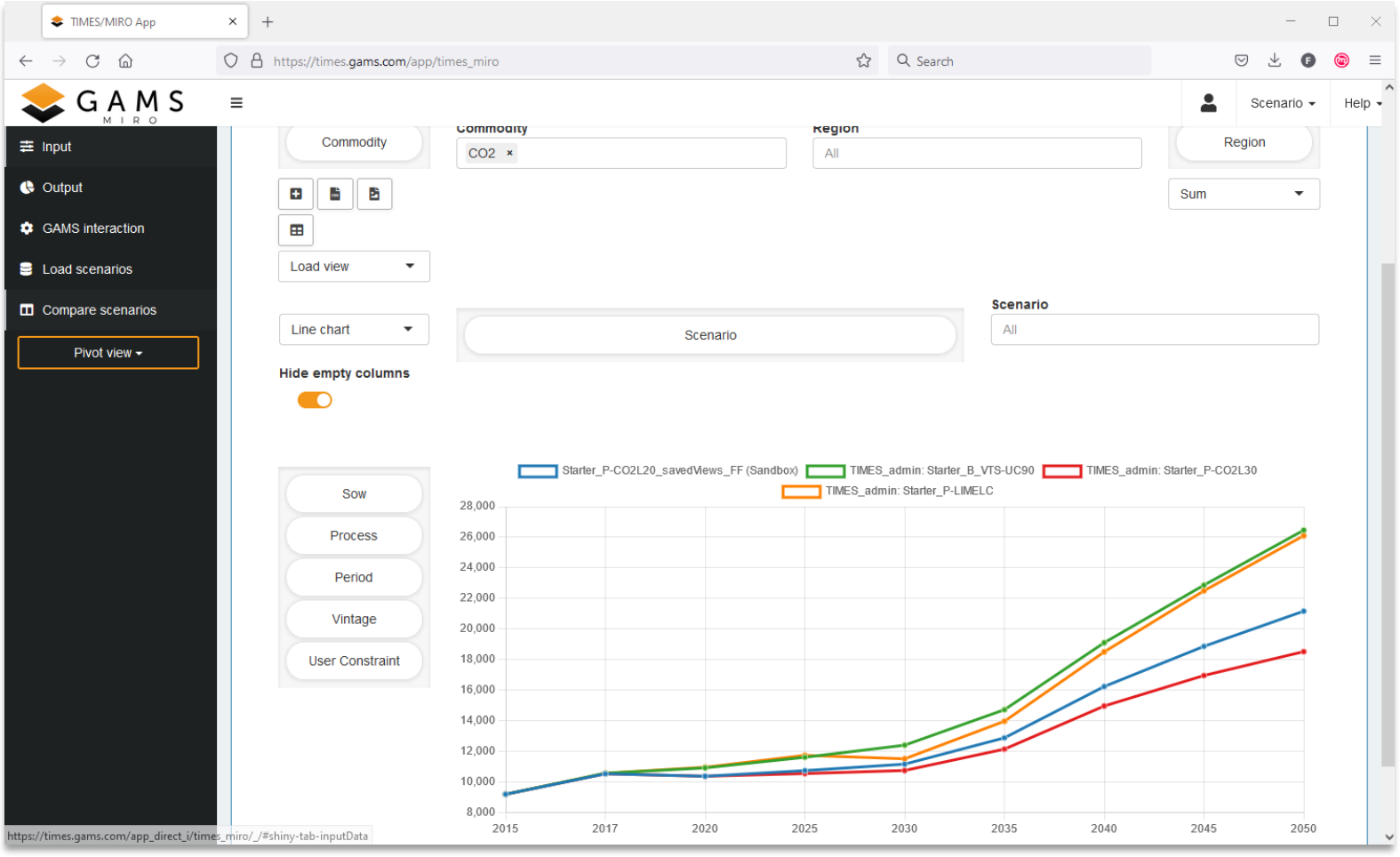
GAMS Engine Custom Clients – VEDA



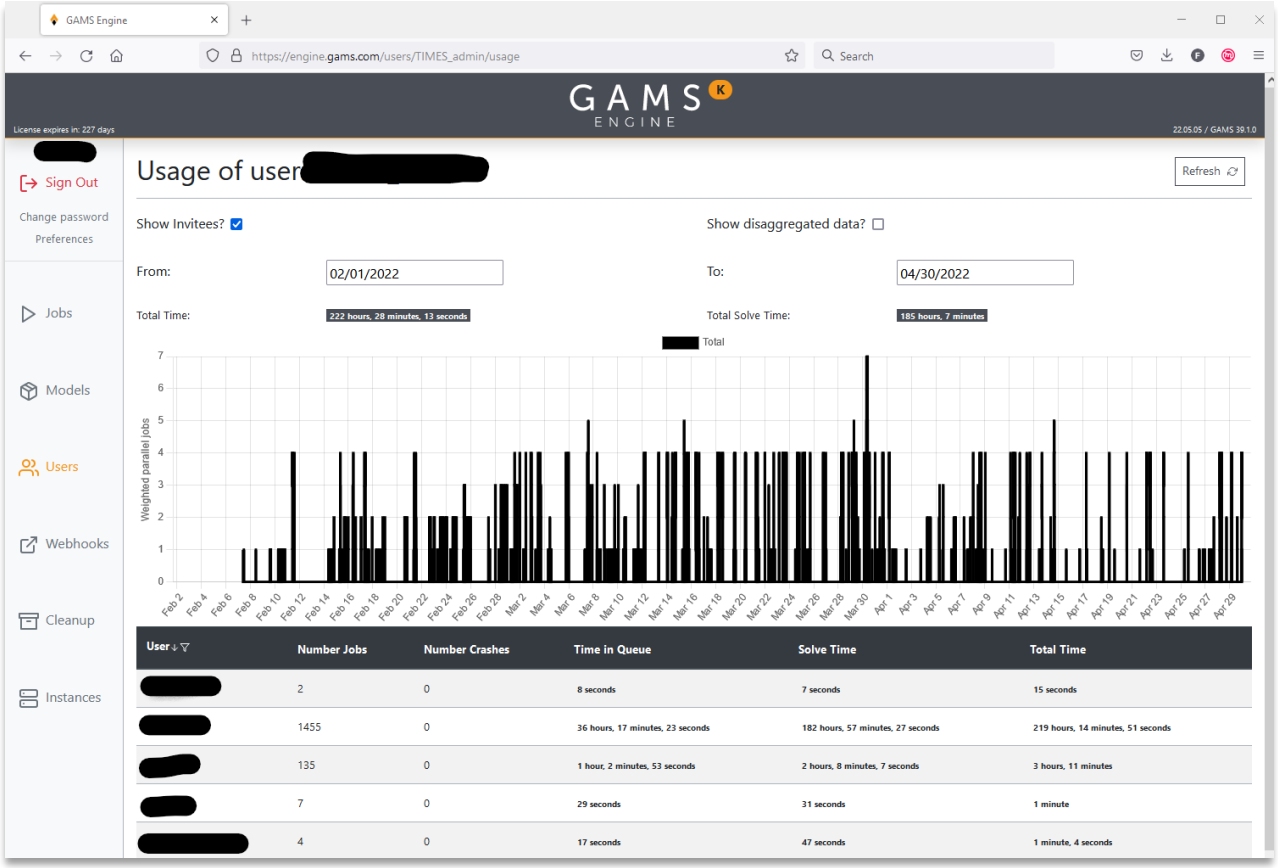
GAMS Engine Clients



GAMS Engine Clients - The TIMES MIRO App (GitHub)



TIMES Cloud Service Statistics



2022

- 14 users
- 1940 jobs
- 301+ hours of solve time
- 0 crashes

Key Benefits

The TIMES Cloud Service (and the TIMES MIRO app) enhance...

- the **diversity**
- the **transparency**
- the **affordability**

... of the TIMES toolbox.

→ Attract new user groups
→ Facilitate new ways to disseminate TIMES models
→ Increase the openness of the TIMES modelling framework

Thank You!

ffiand@gams.com
www.gams.com

@GamsSoftware

https://www.linkedin.com/company/gams-development