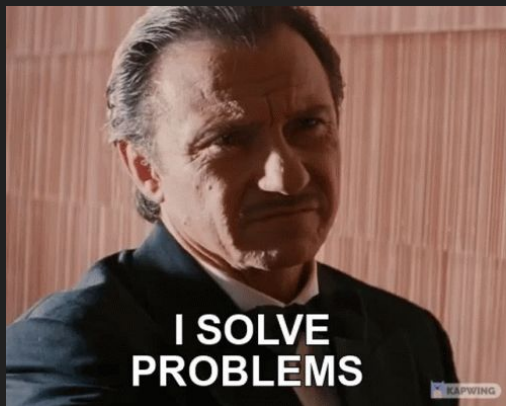
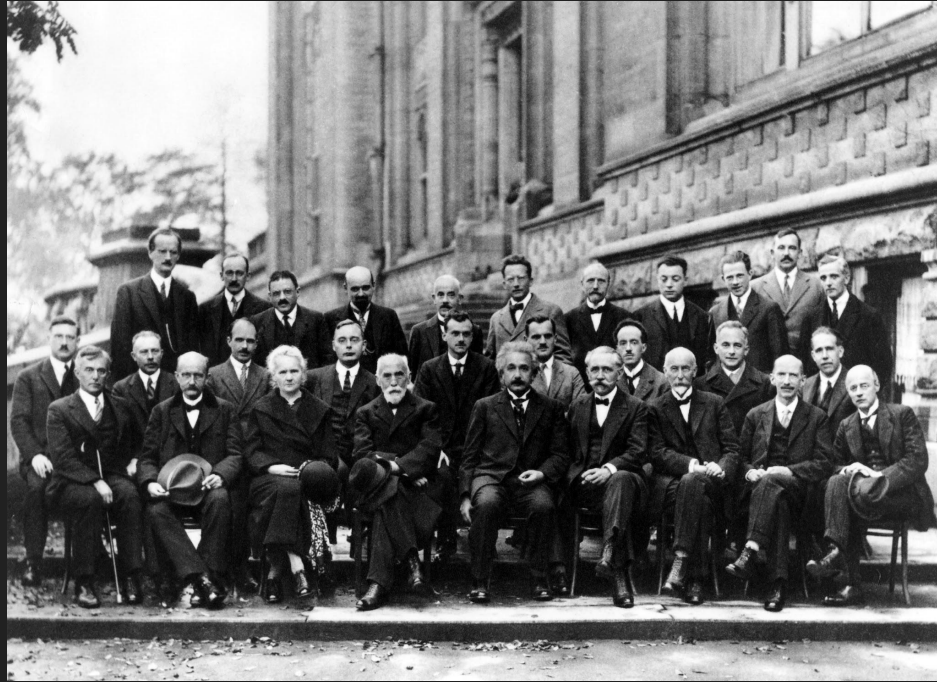
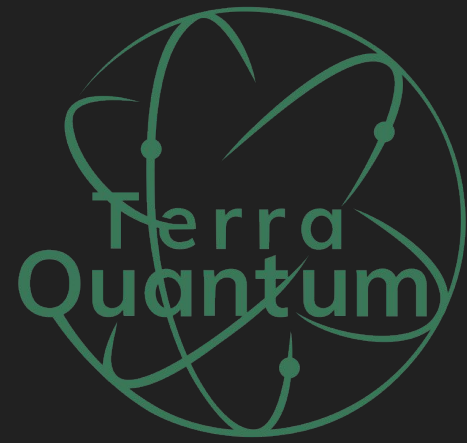


I am Pratik and...



... for the love and respect of Physics together with all of her saints, classic and modern.





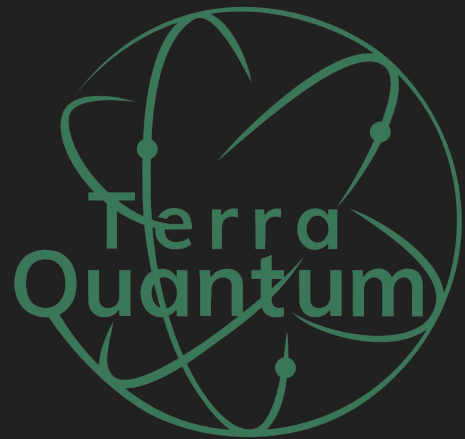
TQ Solution

for TTOpt solver



Table of contents:

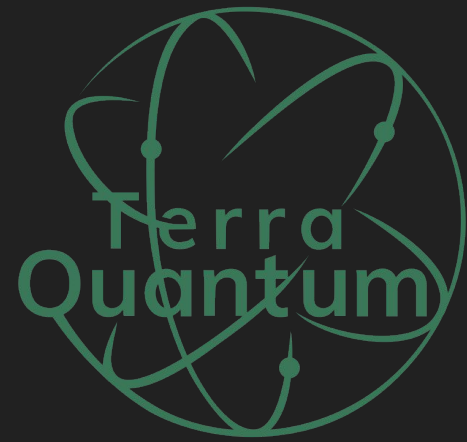
- Whats
 - Problem statement
 - Requirements
 - Solution
 - PoC & features
- Hows
 - Architecture
 - Design
 - Component details
 - Backend
 - Storage
 - Frontend
 - Proxy
 - Tech stack
 - Links & more
- Disclaimers



Whats

- Problem Statement

... task is to take TTOpt solver (use this [example](#)) and create simple software that the user can install locally to find minima of multi-dimensional function in Python and a simple web page that can do the same. The most important aspect is that the code for TTOpt must be hidden from the user, while the simple software is straightforward to use.



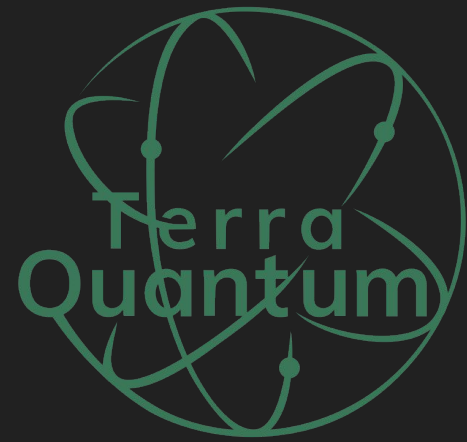
- Deliverables:

- **Source Code:** Python or C++ implementation of the algorithm.
- **Dockerfile:** For containerized application.
- **SDK Module:** Enable easy integration of the algorithm.
- **Qt:** Example of the solution for several functions.
- **CI/CD pipeline:** Perform build, deploy and unit tests of your solution.
- **Docker and Github.**
- **Detailed installation instructions:** Prepare the instructions for scientists not familiar with software development.



Whats

- Requirements

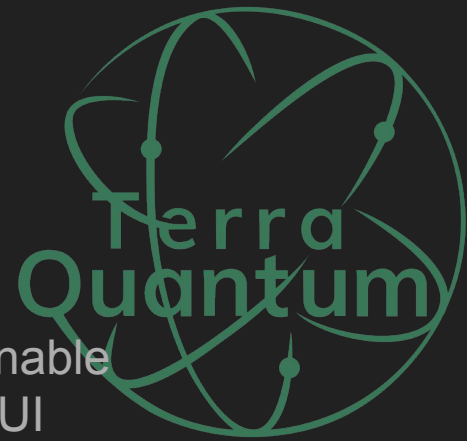


- Functional requirements
 - local installation - stand alone SW
 - utilize TTOpt solver locally
 - software **must display** the optimization results to the user: minima with func vector + logs
 - simple UI
 - Web-based version
 - utilize TTOpt solver on the server in cloud
 - similar simple UI.
 - **must display** optimization results
 - downloadable solution result with logs



Whats

- Requirements



- Non functional requirements
 - **Performance** - complete optimization within a reasonable time frame (e.g., ~ under a minute), responsive web UI
 - **Usability** - simple UI, accessible and easy to use on different devices and browsers
 - **Security** - TTOpt solver code **must be** isolated
 - **Compatibility** - local software must run on major operating systems (Windows, macOS, Linux)
 - **Reliability** - handle errors, provide clear msgs, uptime 99.9%
 - **Scalability** - handle multiple concurrent users
 - **Maintainability** - well-documented future updates
 - **Installation** - one-click installer for local sw



Whats

- Solution

Standalone,
local SW

Terra Quantum Optimization

Terra Quantum Optimization Tool

Online : Straight from the cloud

Function: Simple $f(x) = \text{sum}(\text{abs}(x * \sin(x) + 0.1 * x))$, x in [-10, 10]

Dimensions (d): 10

Grid Lower Bound (a): -1

Grid Upper Bound (b): 1

Grid Size Factor (p): 4

Grid Size Factor (q): 4

Number of Evals: 100,000

is_func: ☒ is_vect: ☒ with_cache: ☐

with_log: ☒ with_opt: ☐

Run Optimization

Solver Status: Done

Function Name: Simple-10d

Evals: 1.00e+05

t_all or t_cur: 2.88e+00

e_x: 1.24e-02

e_y: 1.40e-05

Recalculate Download

Download completed successfully

2024-10-17 10:39:40,003 - root - INFO - Optimization request received

2024-10-17 10:39:40,008 - root - INFO -

2024-10-17 10:39:40,008 - root - INFO - {'dimensions': 10, 'evals': 100000, 'forceRecal': False, 'funcName': 'Simple', 'gri

2024-10-17 10:39:40,009 - root - INFO -

2024-10-17 10:39:40,032 - root - INFO - Simple-10d | evals=8.00e+01 | t_cur=5.20e-04 | e_x=1.89e+00 e_y=2.70e-01

2024-10-17 10:39:40,034 - root - INFO - Simple-10d | evals=1.44e+02 | t_cur=6.70e-04 | e_x=1.88e+00 e_y=2.66e-01

2024-10-17 10:39:40,036 - root - INFO - Simple-10d | evals=2.08e+02 | t_cur=7.95e-04 | e_x=1.88e+00 e_y=2.66e-01

2024-10-17 10:39:40,037 - root - INFO - Simple-10d | evals=2.72e+02 | t_cur=9.32e-04 | e_x=1.87e+00 e_y=2.64e-01

2024-10-17 10:39:40,039 - root - INFO - Simple-10d | evals=3.36e+02 | t_cur=1.09e-03 | e_x=1.86e+00 e_y=2.59e-01

2024-10-17 10:39:40,040 - root - INFO - Simple-10d | evals=4.00e+02 | t_cur=1.22e-03 | e_x=1.63e+00 e_y=1.79e-01

2024-10-17 10:39:40,042 - root - INFO - Simple-10d | evals=4.64e+02 | t_cur=1.34e-03 | e_x=1.63e+00 e_y=1.79e-01

2024-10-17 10:39:40,043 - root - INFO - Simple-10d | evals=5.28e+02 | t_cur=1.48e-03 | e_x=1.63e+00 e_y=1.79e-01

2024-10-17 10:39:40,045 - root - INFO - Simple-10d | evals=5.92e+02 | t_cur=1.62e-03 | e_x=1.63e+00 e_y=1.79e-01

2024-10-17 10:39:40,047 - root - INFO - Simple-10d | evals=6.56e+02 | t_cur=1.74e-03 | e_x=1.61e+00 e_y=1.74e-01

2024-10-17 10:39:40,049 - root - INFO - Simple-10d | evals=7.20e+02 | t_cur=1.90e-03 | e_x=1.61e+00 e_y=1.74e-01

2024-10-17 10:39:40,050 - root - INFO - Simple-10d | evals=7.84e+02 | t_cur=2.03e-03 | e_x=1.61e+00 e_y=1.74e-01

2024-10-17 10:39:40,052 - root - INFO - Simple-10d | evals=8.48e+02 | t_cur=2.16e-03 | e_x=1.59e+00 e_y=1.67e-01

2024-10-17 10:39:40,053 - root - INFO - Simple-10d | evals=9.12e+02 | t_cur=2.27e-03 | e_x=1.59e+00 e_y=1.67e-01

Whats

- Solution

Web based
SW

apptq_frontend

Not secure 34.32.71.110:3000/apptq_frontend.html

Terra Quantum Optimization Tool

Online : Straight from the cloud

Function:

Simple

f(x) = sum(abs(x * sin(x) + 0.1 * x)), x in [-10, 10]

Dimensions (d):

10

Grid Lower Bound (a):

-1

Grid Upper Bound (b):

1

Grid Size Factor (p):

4

Grid Size Factor (q):

4

Number of Evals:

-

100,000

+

is_func: ☒

is_vect: ☒

with_cache: ☐

with_log: ☒

with_opt: ☐

Run Optimization

Solver Status: Done

Function Name: Simple-10d

Evals: 1.00e+05

t_all or t_cur: 2.92e+00

e_x: 1.24e-02

e_y: 1.40e-05

Recalculate

Download

Download completed successfully

Save

2024-10-17 10:47:16,642 - root - INFO - Optimization request received

2024-10-17 10:47:16,643 - root - INFO - -----

2024-10-17 10:47:16,643 - root - INFO - {'dimensions': 10, 'evals': 100000, 'forceRecal': True, 'funcName': 'Simple', 'gridSizeFactorP': 4, 'gridSizeFactorQ': 4, 'isFunc'

2024-10-17 10:47:16,643 - root - INFO - -----

2024-10-17 10:47:16,660 - root - INFO - Simple-10d | evals=8.00e+01 | t_cur=2.68e-04 | e_x=1.89e+00 e_y=2.70e-01

2024-10-17 10:47:16,662 - root - INFO - Simple-10d | evals=1.44e+02 | t_cur=4.17e-04 | e_x=1.88e+00 e_y=2.66e-01

2024-10-17 10:47:16,663 - root - INFO - Simple-10d | evals=2.08e+02 | t_cur=5.35e-04 | e_x=1.88e+00 e_y=2.66e-01

2024-10-17 10:47:16,665 - root - INFO - Simple-10d | evals=2.72e+02 | t_cur=6.56e-04 | e_x=1.87e+00 e_y=2.64e-01

2024-10-17 10:47:16,666 - root - INFO - Simple-10d | evals=3.36e+02 | t_cur=7.90e-04 | e_x=1.86e+00 e_y=2.59e-01

2024-10-17 10:47:16,668 - root - INFO - Simple-10d | evals=4.00e+02 | t_cur=9.09e-04 | e_x=1.63e+00 e_y=1.79e-01

2024-10-17 10:47:16,669 - root - INFO - Simple-10d | evals=4.64e+02 | t_cur=1.04e-03 | e_x=1.63e+00 e_y=1.79e-01

2024-10-17 10:47:16,686 - root - INFO - Simple-10d | evals=5.28e+02 | t_cur=2.38e-03 | e_x=1.63e+00 e_y=1.79e-01

2024-10-17 10:47:16,699 - root - INFO - Simple-10d | evals=5.92e+02 | t_cur=2.59e-03 | e_x=1.63e+00 e_y=1.79e-01

2024-10-17 10:47:16,718 - root - INFO - Simple-10d | evals=6.56e+02 | t_cur=2.76e-03 | e_x=1.61e+00 e_y=1.74e-01

2024-10-17 10:47:16,720 - root - INFO - Simple-10d | evals=7.20e+02 | t_cur=2.91e-03 | e_x=1.61e+00 e_y=1.74e-01

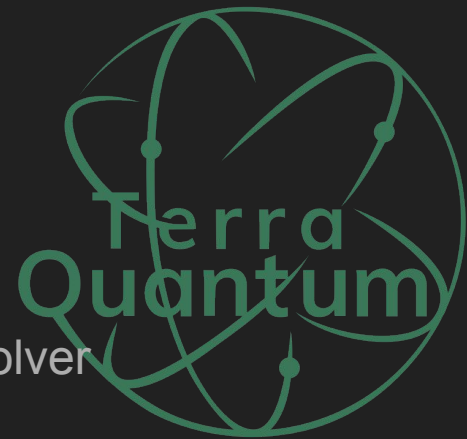
2024-10-17 10:47:16,721 - root - INFO - Simple-10d | evals=7.84e+02 | t_cur=3.04e-03 | e_x=1.61e+00 e_y=1.74e-01

2024-10-17 10:47:16,723 - root - INFO - Simple-10d | evals=8.48e+02 | t_cur=3.16e-03 | e_x=1.59e+00 e_y=1.67e-01



Whats

- Features



- Modes: Online & Offline

- Both versions (web based & local app) can use TTSolver available on cloud as well as local installation



Online : Straight from the cloud

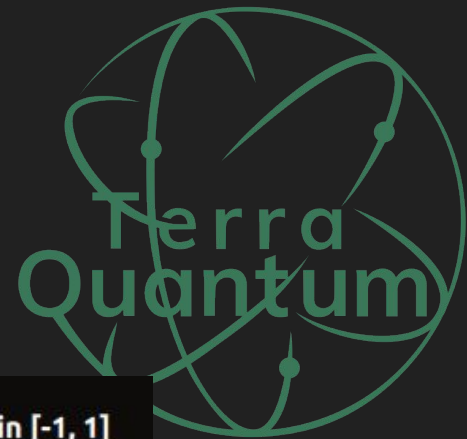


Offline: Make sure tq-backend is available at <http://localhost:5000>



Whats

- Features
 - Easy drop down for function selection



Function: $f(x) = \sin(0.1 * x[0])**2 + 0.1 * \text{sum}(x[1:]**2), x \text{ in } [-1, 1]$

Tensor

Alpine

Dimensions: 10

Grid Lower Bound (g): -1

Solver Status:

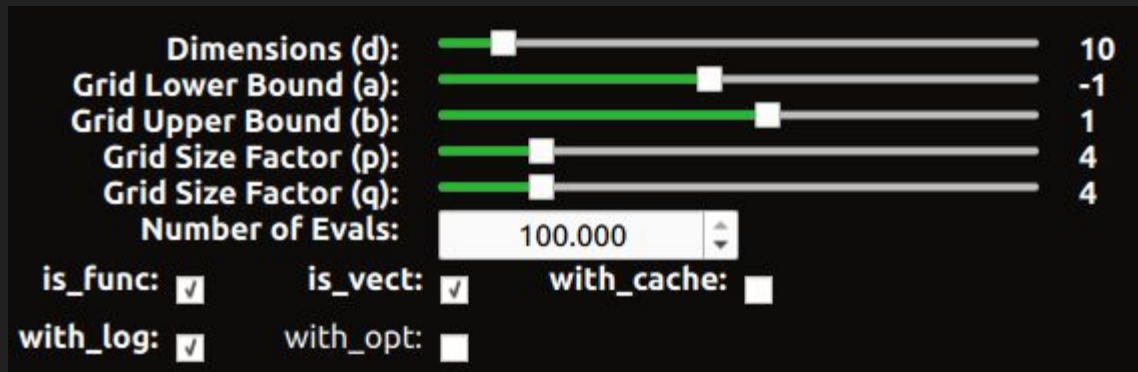
Function: $f(x) = \sin(0.1 * x[0])**2 + 0.1 * \text{sum}(x[1:]**2), x \text{ in } [-1, 1]$



Whats

- Features

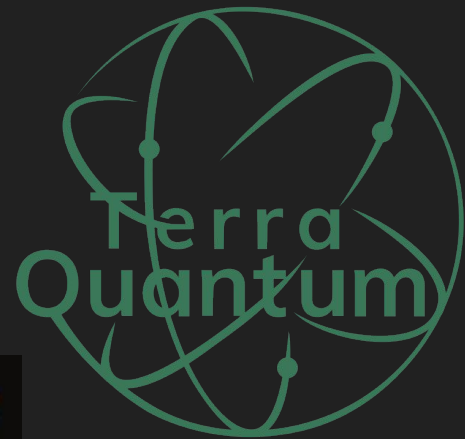
- Simple & smooth sliders to adjust input parameters



Dimensions (d): 10
Grid Lower Bound (a): -1
Grid Upper Bound (b): 1
Grid Size Factor (p): 4
Grid Size Factor (q): 4
Number of Evals: 100.000
is_func: ☒ is_vect: ☒ with_cache: ☐
with_log: ☒ with_opt: ☐

- Button to “**SEND IT!**”

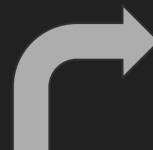
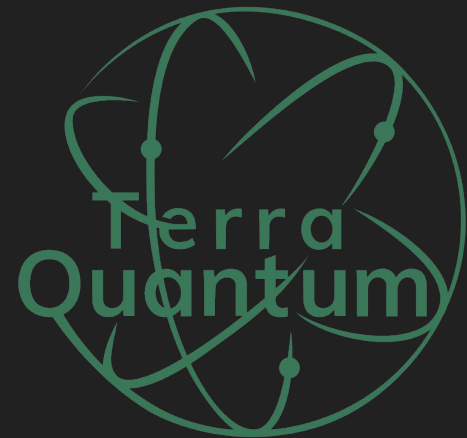
Run Optimization



Whats

- Features

- Results → with status
 - Processing
 - Done
 - Error : error msg
- Easy parsing of output



```
INFO - Tensor-10d | evals=1.00e+05 | t_all=2.93e+00 | e_x=2.24e+00 e_y=5.00e+00  
INFO - Optimization completed successfully  
INFO - -----calculated info-----  
INFO - Tensor-10d | evals=1.00e+05 | t_all=2.93e+00 | e_x=2.24e+00 e_y=5.00e+00
```

Solver Status:	Done
Function Name:	Tensor-10d
Evals:	1.00e+05
t_all or t_cur:	2.93e+00
e_x:	2.24e+00
e_y:	5.00e+00

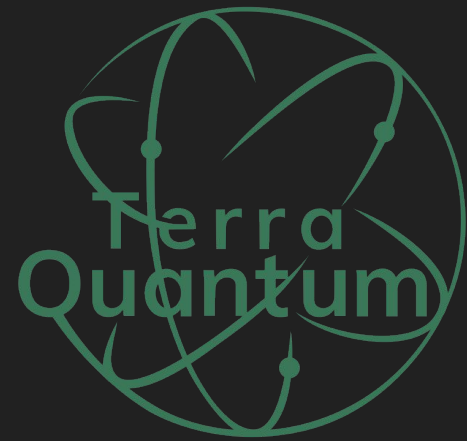
Recalculate

Download

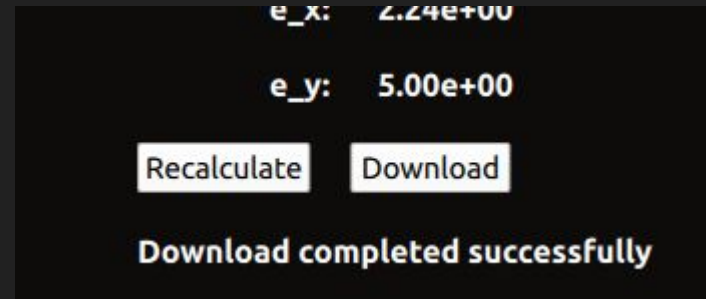


Whats

- Features



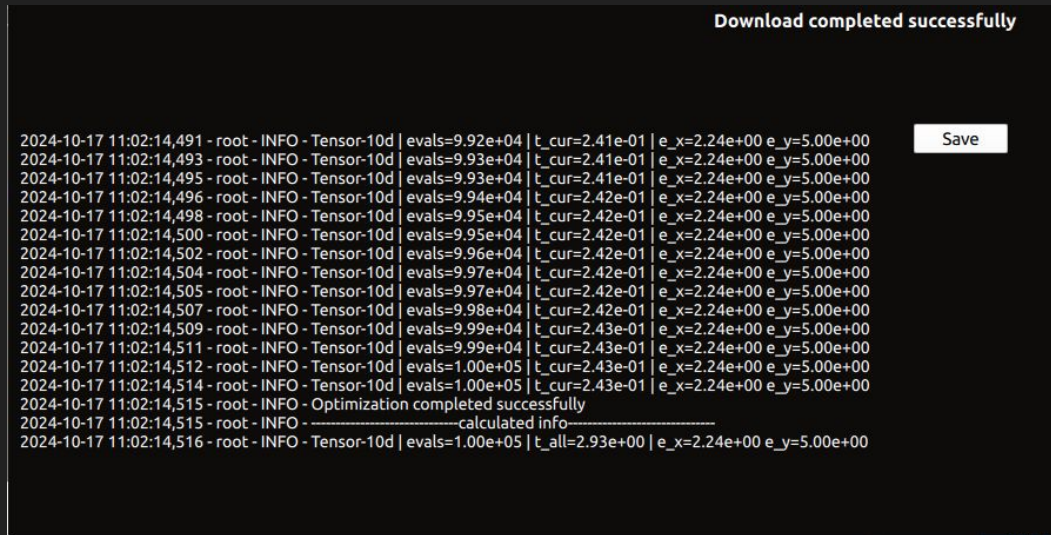
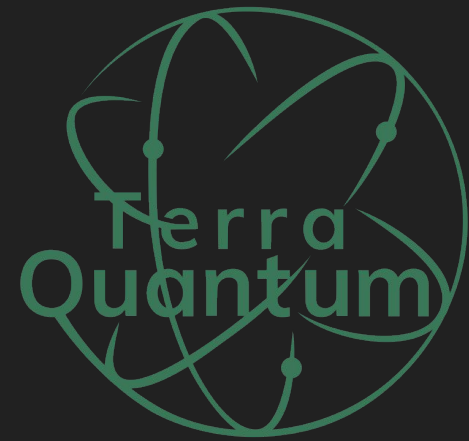
- To improve performance, TTSolver saves the previous calculations done by you and fetches the results faster when requested again. But you never know! You may want to recalculate the minima!
- Download button to get complete info logs from TTSolver
- Of course with download status msg



Whats

- Features

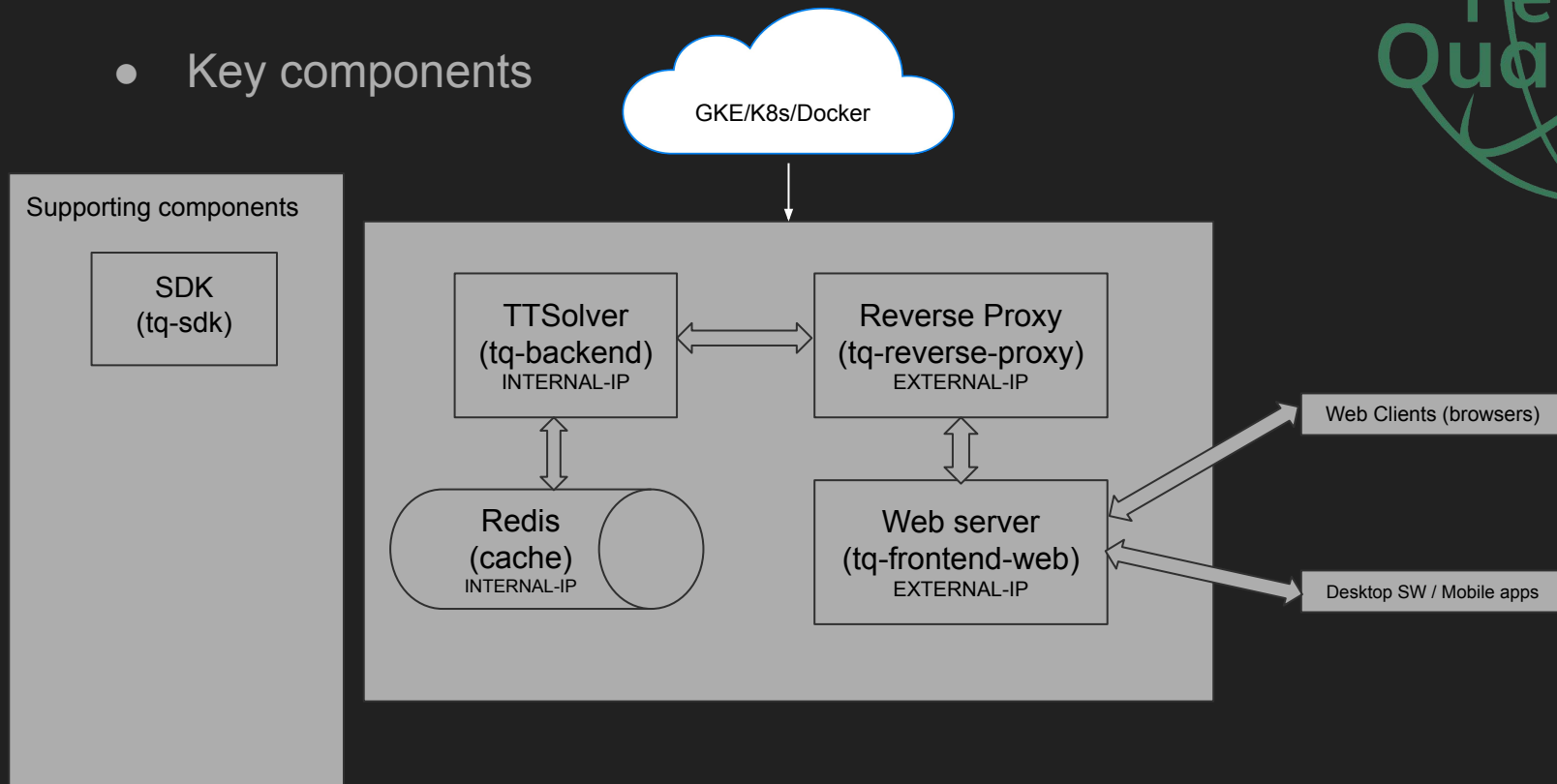
- See the logs directly in Text panel (can do live stream but not in PoC. You have to press download button to see logs). And finally the Save button to save the logs in .txt file format.



Hows

- Architecture

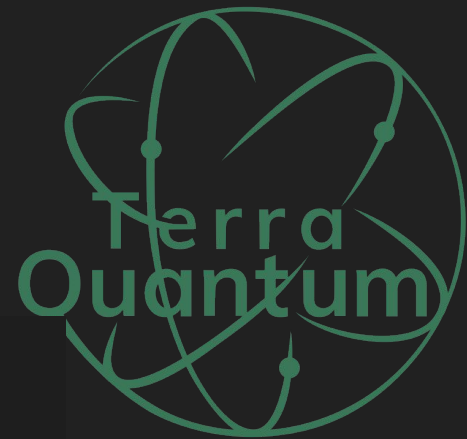
- Key components



Hows

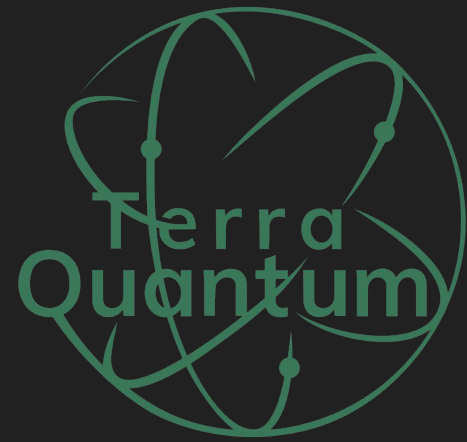
- Architecture

- Sequence flow



Hows

- Architecture



- Components

- Backend (tq-backend)

- Internal component with only an Internal IP
 - Performs the actual calculations for finding minima
 - Communicates with Redis Cache for data storage and retrieval
 - Not directly accessible from outside the cluster

- Redis Cache

- Internal component with only an Internal IP
 - Provides fast, in-memory data storage for the TTSolver Backend
 - Not directly accessible from outside the cluster



Hows

- Architecture

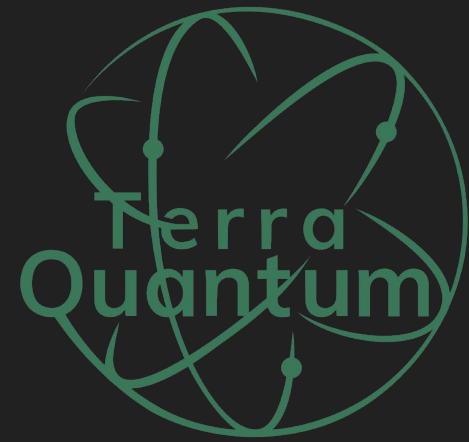
- Components

- Frontend Web UI

- Exposed with an External IP
 - Serves the user interface for calculation requests
 - Communicates with the Reverse Proxy to forward requests to the backend

- Reverse Proxy

- Exposed with an External IP
 - Acts as an intermediary between the Frontend and Backend
 - Handles known endpoints only (e.g., public frontend)
 - Provides an additional layer of security



Hows

- Architecture

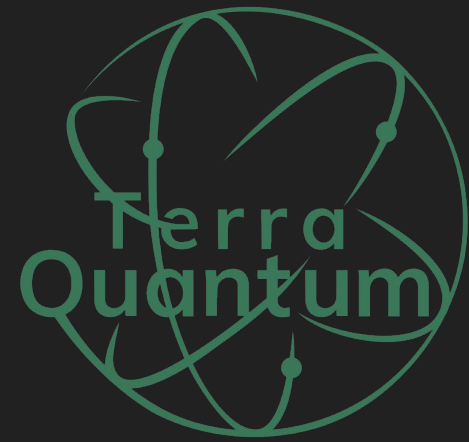
- Components

- Frontend Web UI

- Exposed with an External IP
 - Serves the user interface for calculation requests
 - Communicates with the Reverse Proxy to forward requests to the backend

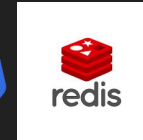
- Reverse Proxy

- Exposed with an External IP
 - Acts as an intermediary between the Frontend and Backend
 - Handles known endpoints only (e.g., public frontend)
 - Provides an additional layer of security

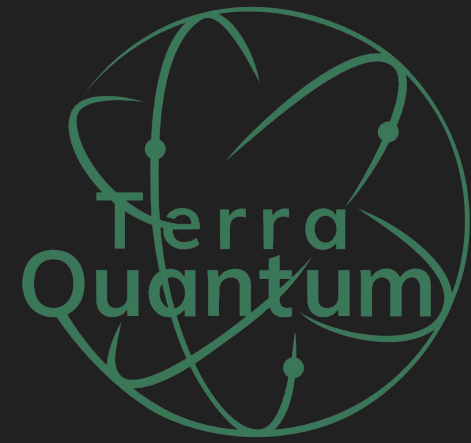


Hows

- Architecture
 - Tech stack



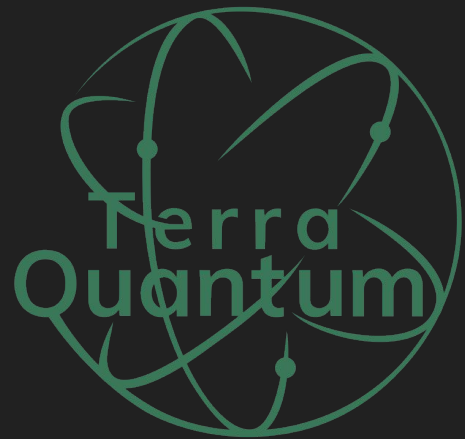
Google Cloud



Hows

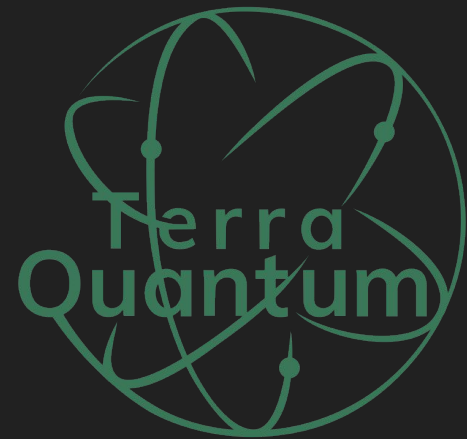
- Links & more...

- Check out: http://34.32.71.110:3000/apptq_frontend.html
- Github: (README.md must be read!)
 - https://github.com/ppraiap/tq_solution.git
 - https://github.com/ppraiap/tq_backend.git
 - https://github.com/ppraiap/tq_frontend.git
 - WIP: https://github.com/ppraiap/tq_sdk.git
- Docker Images:
 - <https://hub.docker.com/r/ppraijapa/tq-backend>
 - <https://hub.docker.com/r/ppraijapa/tq-frontend-web-light>
 - <https://hub.docker.com/r/ppraijapa/tq-reverse-proxy>
 - <https://hub.docker.com/r/ppraijapa/tq-frontend-desktop>
- Fun facts
 - Total commits: [109+](#)
 - Google cloud credits:
 - ~200 EUR remaining
 - Expires in Jan. 2025
 - CI/CD pipelines run: [79+](#)
 - [Link to this very presentation!](#)



Disclaimer

- **Google Cloud:** One can use any cloud service provider to host the solution. I choose Google cloud service because I already exhausted AWS & Azure's free period / initial free credits. And this is my first time with GCR + GKE. Good that I get to explore!
- **Redis:** Redis cache is not included in the solution. It doesn't allow to deploy services using Redis with its open source version. Sad but fair enough. I instead use `unordered_map` directory directly inside backend service app. Plus GCR takes care of scaling in case of heavy memory requests.
- **Github commits:** Some commit messages on solution repo may show little frustration! But I'm sure it will bring some chuckles :)
- **Code visibility:** I have kept the code open / public, anyone can see the solution. If you have concerns, feel free to write me:
pratik.prajapati1989@gmail.com



... thank you!

