

client /server chat

Gaël Léauté • 14.07.2023

Overview

Content

- Demo
- Solution strategy
- The solutions
- Tools and technology
- Architecture and design
- Advantages and disadvantages

Demo

Solution Strategy

Use a RPC library

- Limits risks, but adds constraints.
- Stakeholders' explicit time constraints.
- Not reinventing existing solutions

Find RPC libraries

- With existing client/server interfaces.
- Object oriented
- Check feasibility by reading documentation

Which solutions and technology

rpclib

- C++ library.
 - Uses cmake.
 - Can be configured for many platforms, including embedded solutions.
-
- Not a symmetric protocol, no slave service
 - Source code quality and maintainability issues.
 - Not widely used: limited documentation and support.

rpcy

- Python library.
 - Source code portability across many platforms.
 - Limited support for embedded solution.
-
- Works out of the box, no configuration needed.
 - Documented as a symmetric protocol.

Tools used for the development

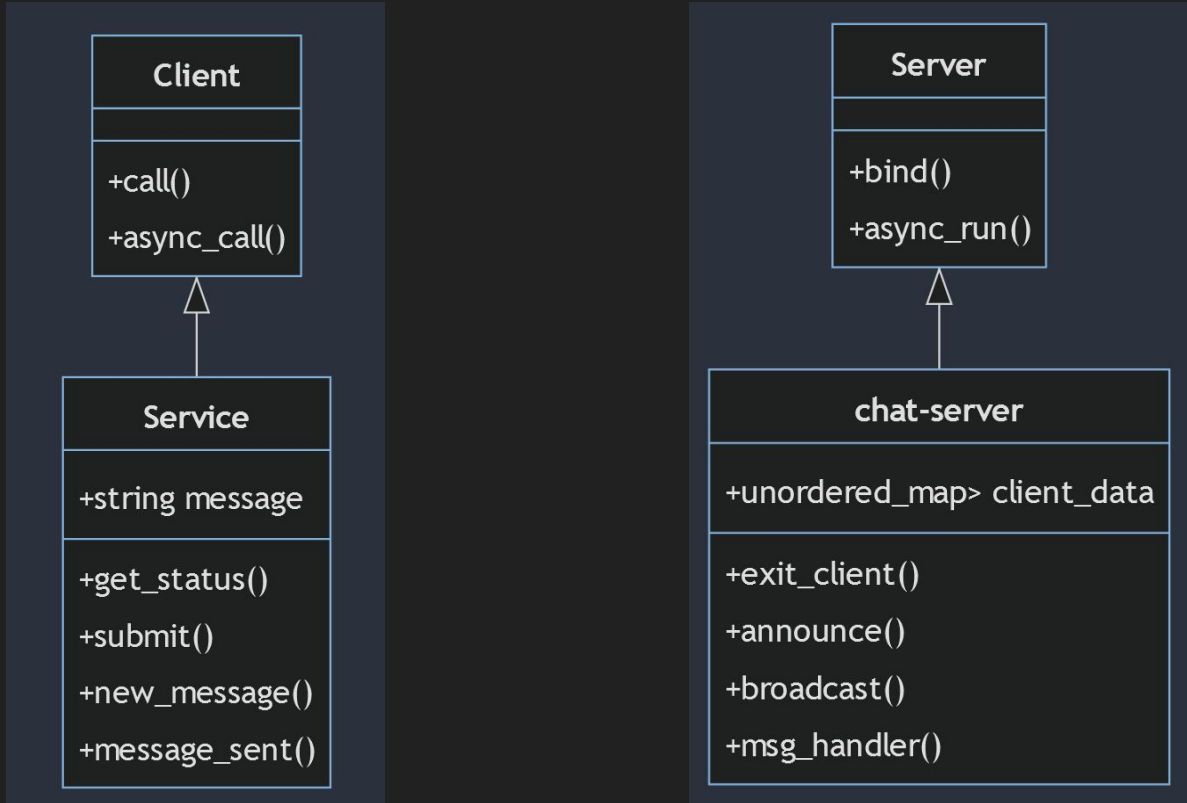
rpclib setup

- Visual Studio Code (for rpcLib)
- Visual Studio (for project solutions)
- CMake
- Visual Studio C++ Tool chains and the necessary plugins

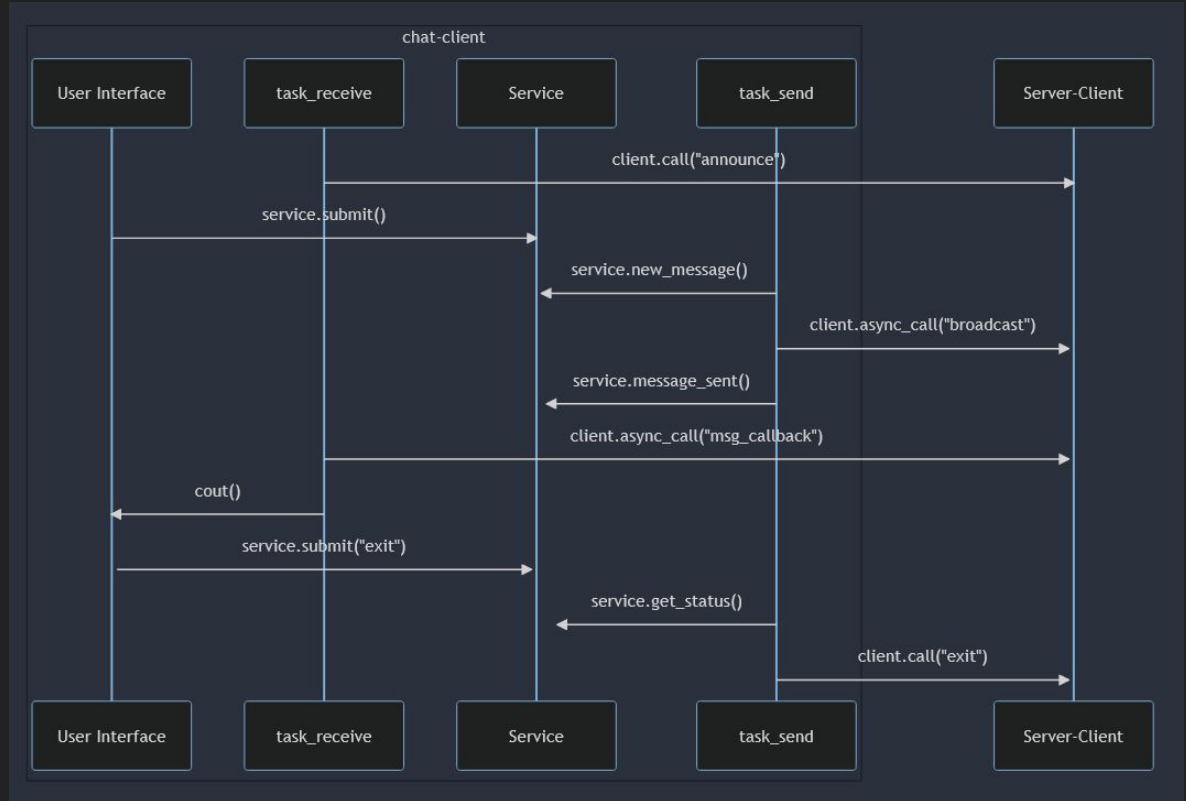
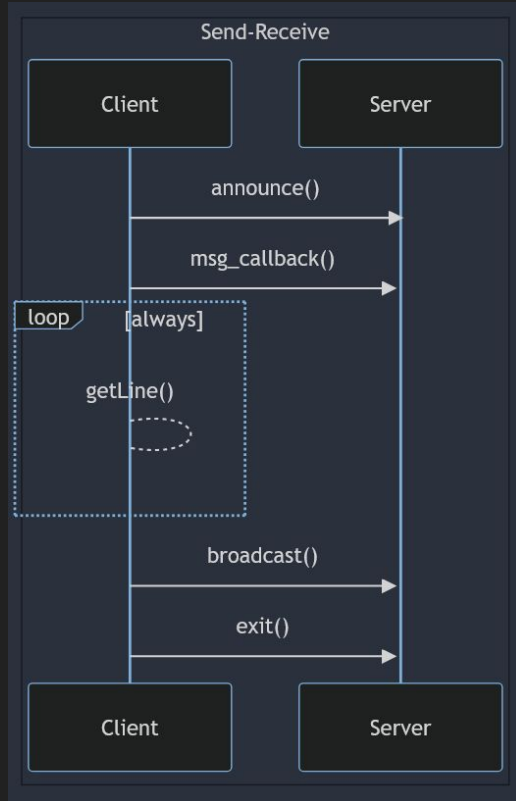
rpypc setup

- Spyder
- Python 3.7.9
- pyflakes

Architecture and design (class diagram)



Architecture and design (sequence diagram)



Advantages and disadvantages

Advantages

- TCP based
- `Unordered_map` for client data brings elegance to the implementation.
- Fast coding, small solution.
- New request commands can be implemented in a short time.
- On server each client sink has its own thread: clients are served in parallel.
- Server does not need to wait for the client.

Disadvantages

- Client does not wait if server is not online or disconnects.
- Not event based. The usage of events would optimize resource usage.
- No error handling
- Client callback is blocking a thread
- The connection is not secure, but implementation of SSL or ECC encryption possible.
- Messages in the queue are overwritten if client not ready.

Questions?

Summary:

1. RPC based client server chat
2. Simplicity of the solution
3. Advantages and disadvantages
4. Further improvements possible