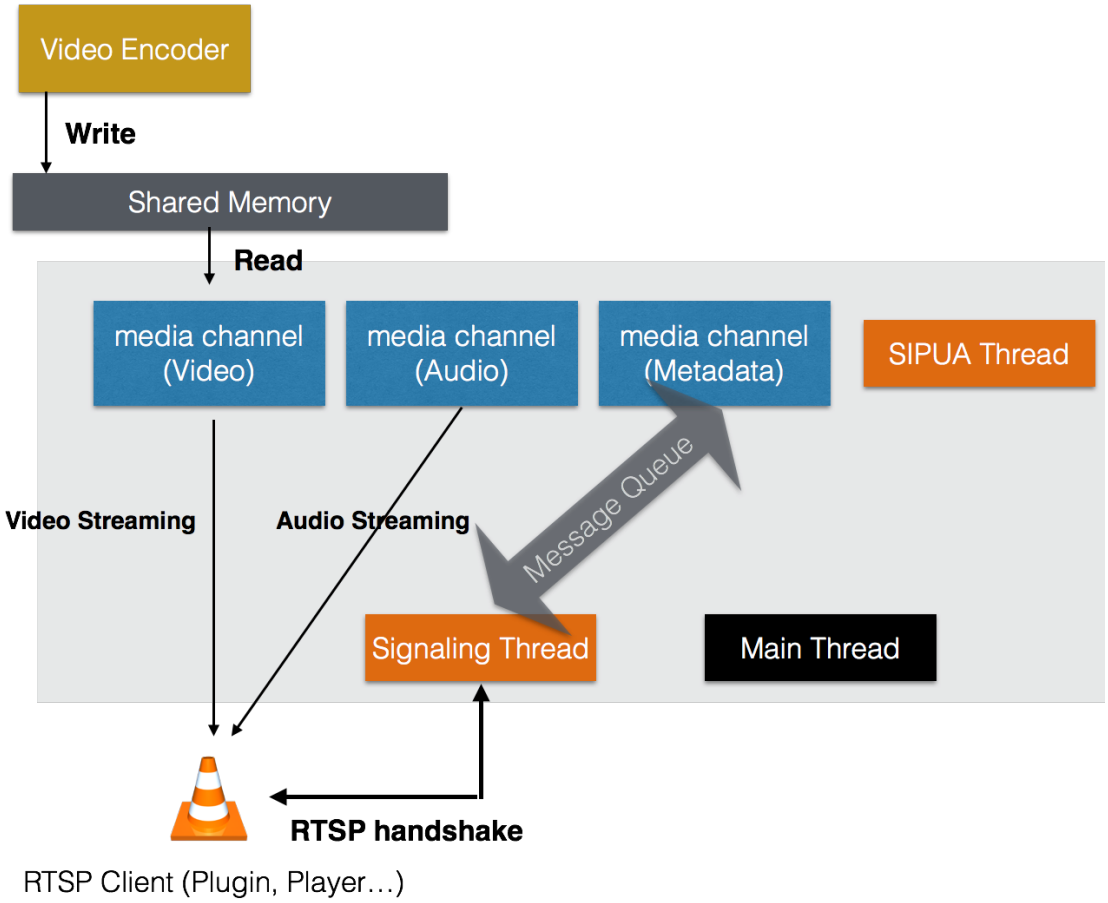
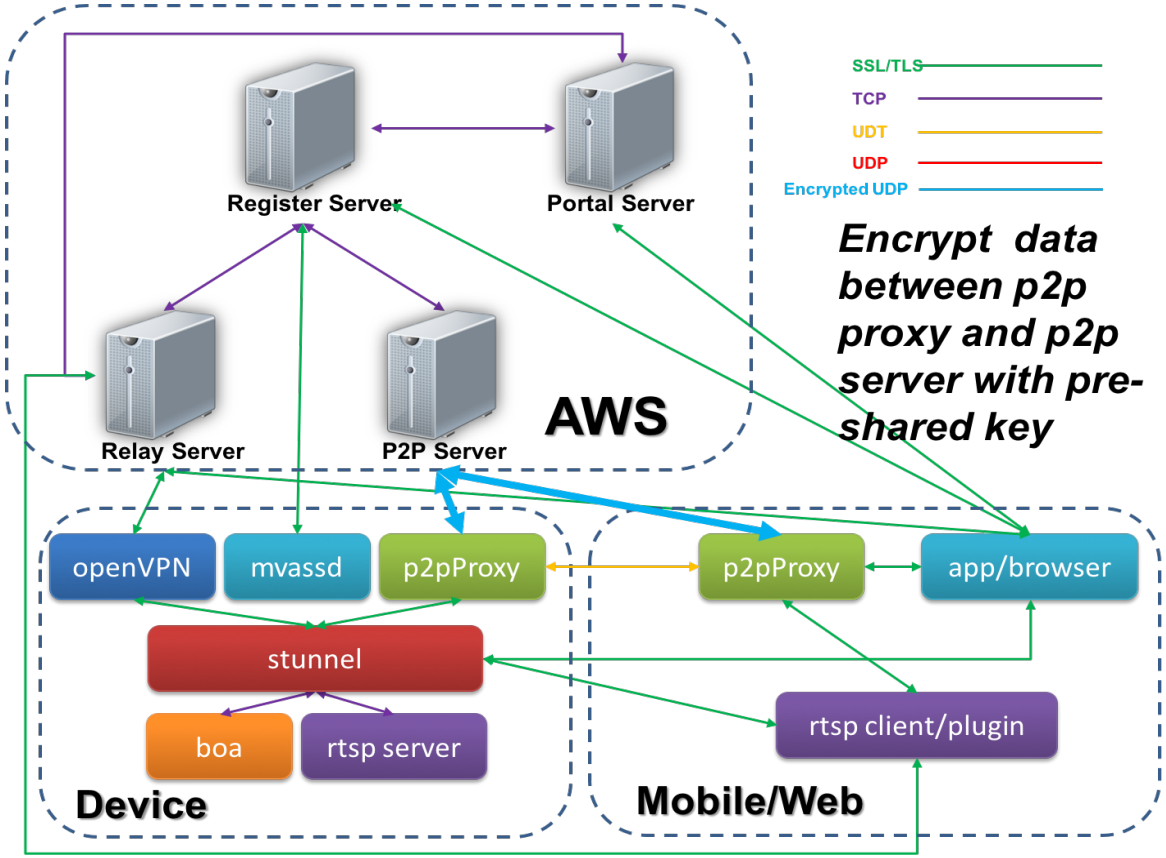
[Project] RTSP Streaming Server



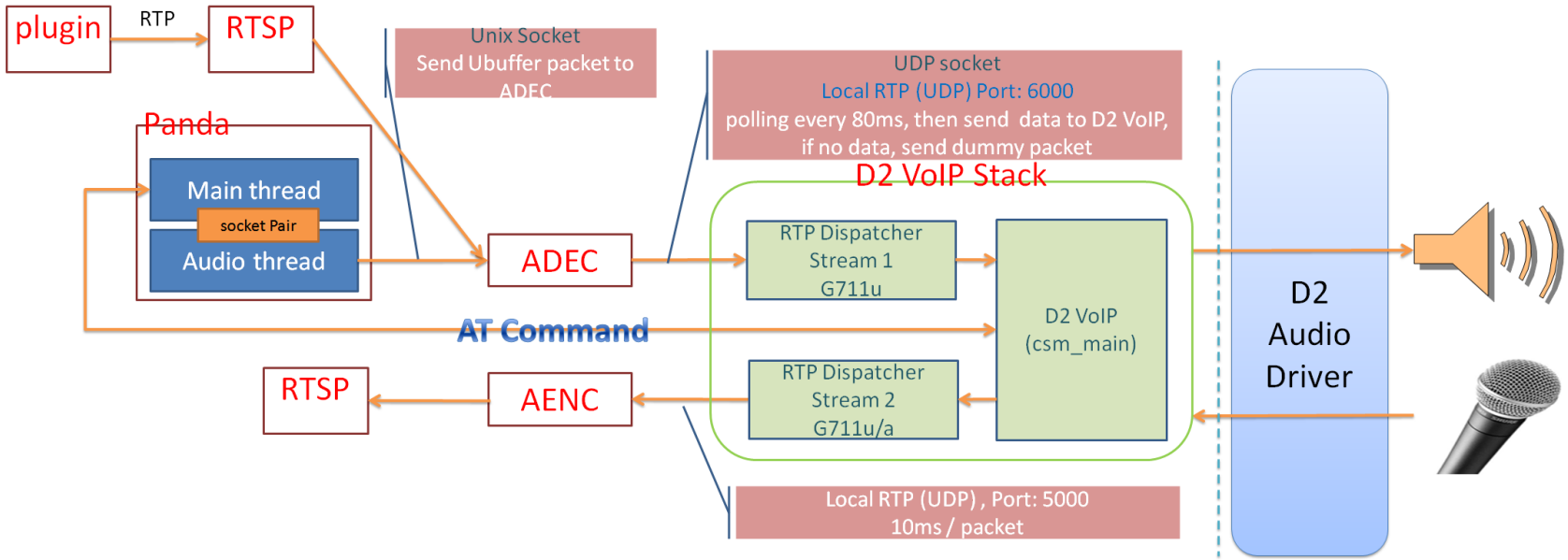
- I was responsible for developing this crucial module inside IP cameras, using C on embedded Linux.
- It is mainly constructed with 6 threads: Main thread, signaling thread, SIPUA thread, and three media channel threads (video, audio, metadata).
- The shared memories are ring buffers for encoders to write frame data & media threads to read frame data.
- It supports various protocols such as RTP over UDP, RTSP over HTTP and SIP. I dedicated to the RFC compliance while keeping flexibility with other third-party software.
- I Implemented H.265 streaming and knew how to encode raw data, composed the corresponding header, and packetized into RTP packets.
- I resolved various issues on networks, OS, and video/audio encoding.

[Project] MVaaS (Managed Video as a Service) Cloud System



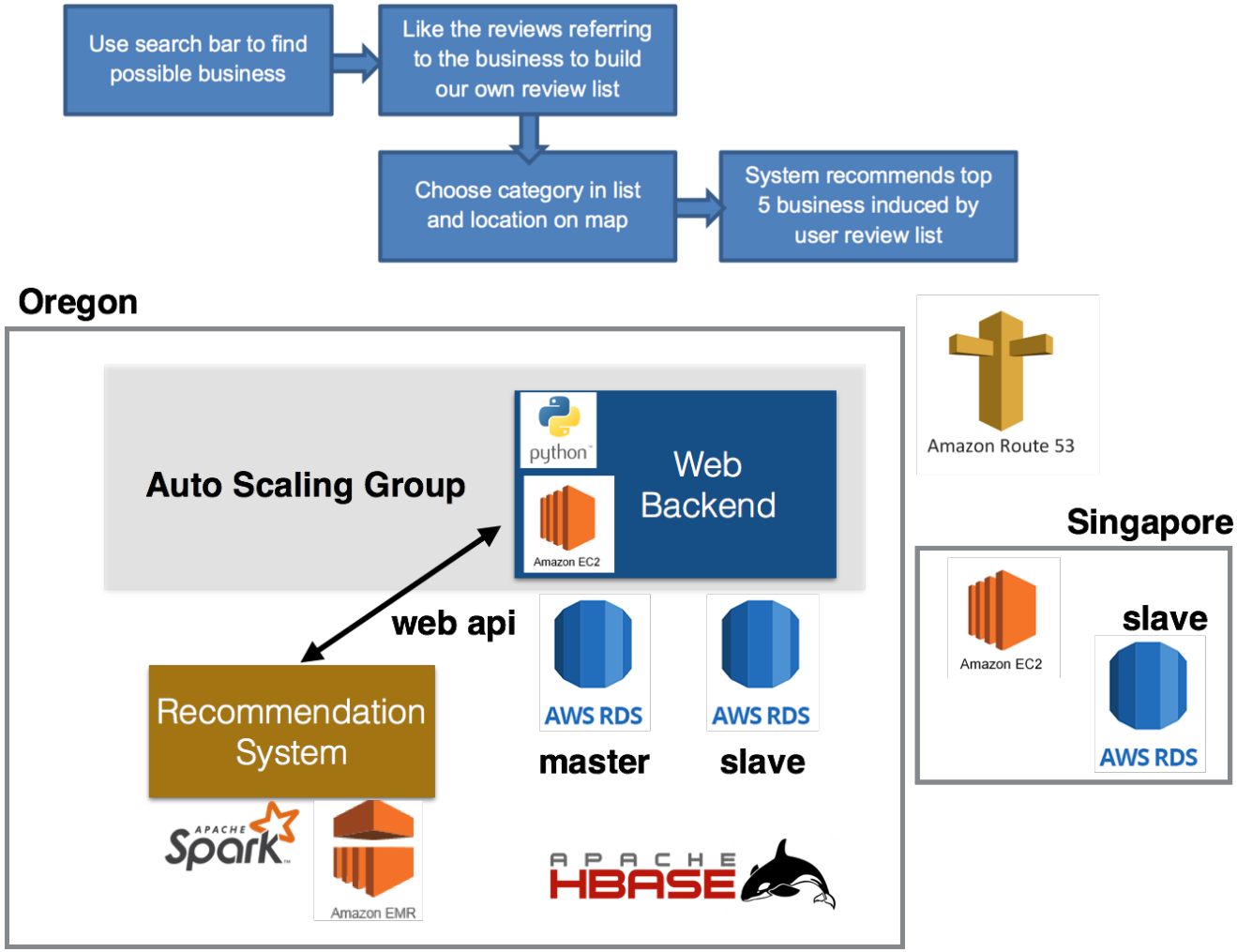
- Users can easily manage their devices (IP cameras) after logging in. Once the camera connects to the cloud platform, users can freely watch streaming by mobile apps without additional installation.
- I mainly developed cross-platform P2P library both for IP cameras and mobile apps, using C++. The P2P hole punching procedure is based on UDP and the data transmission is based on UDT.
- I designed a novel port predict algorithm to improve the success rate of P2P hole punching by 20%.
- The messages exposed to the public network are all well encrypted.

[Project] SIP Based Video Doorbell (ODM Project)



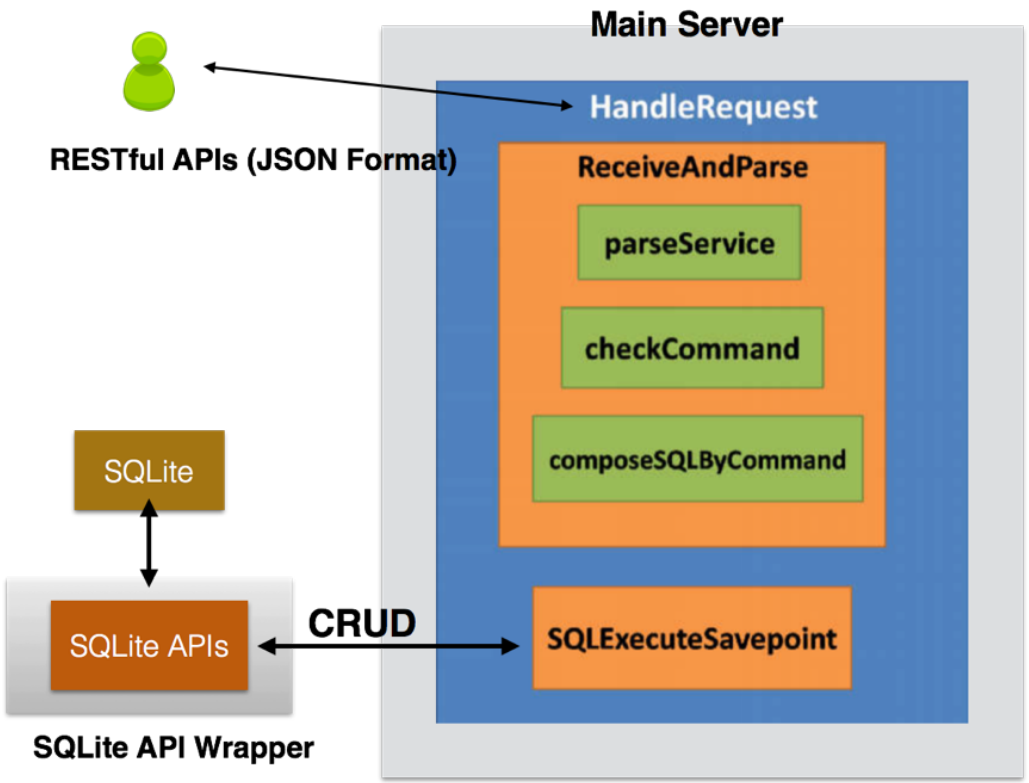
- This project is to design a time-critical VoIP system. The SIP protocol stacks is a kernel driver, and ensure our audio delay is no more than 10ms.
- The video doorbell can be compatible with general SIP phones and RTSP Clients.

[Project] Yelp Recommendation System



- Designed recommendation system by Implementing collaborative filtering algorithm, using Apache Spark and AWS EMR. It can handle requests from backend servers and trained for new recommendations to the client.
- Built the search engine by using HBase to store inverted index of Yelp's open dataset.
- Deployed applications and databases on different AWS region.

[Project] Social Networking Service



- Users can register, login, post articles, response others, add friends, instant chat, and transfer files.
- The main server is a single-threaded(asynchronous non-blocking) servers implemented by C.
- I Encapsulated SQLite APIs into a series of higher level APIs. Through the new APIs, I could register callback functions, which would be called before or after CRUD.
- After handling client's requests, the corresponding SQL Commands would be inserted into a queue as a transaction.