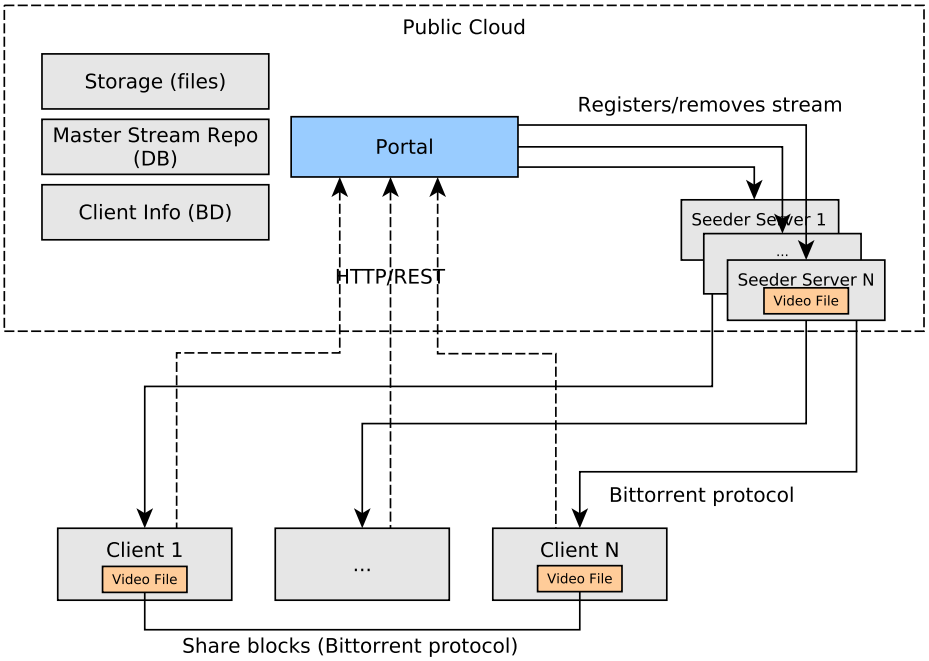
Assignment I: Edge Netflix

Distributed Systems 17/18

October 3, 2017

This assignment requires the implementation of a streaming facility, that makes use of edge computing. All the code must be written in Java or cxx11, with the recommendation being Java. The system **must have exactly** the components depicted below:



This streaming service is a mesh between Bittorrent and Netflix. The idea is to decentralized the streaming service, so instead of only using the cloud servers, the clients must communicate between them and share data blocks for a specific file.

Seeder

The seeder is responsible for seeding **one** file, e.g. a video file, and multiple seeders will be present in the system.

Mandatory features:

- Register in the portal as a provider of a video stream. The registration information must include: 1) stream name, 2) endpoint (transport, ip, port), 3) video size, 4) bitrate and 5) associated set of keywords, e.g., {a_video_stream_name, {tcp, 127.0.0.1, 10000}, 480x270, 400, {basketball, Cavs, indoor, sports}}

- When closing a stream, the portal should be notified (through a message)

System Requirements
Mandatory features:

- Implement a lightweight Bittorrent protocol, between seeders and clients.
  - Each file is divided in chunks, each chunk has a max of 10MB.
  - Each file, and every chunk, must have a hash (SHA256).

Assumptions:

- Assume one seeder per file.
- Must support multiple clients per file.
- For the DBs, use open-source ones, mySQL, Cassandra, etc...
- Use ICE as the middleware.
- Use Jetty as the servlet/REST server.
- All clients have direct connections (no firewalls between them), i.e., all clients are in a local common network
- **All other components must be design and build by the student!**.

Portal
Mandatory features:

- For seeders:
  - Registers seeders to allow a new video (Ice RPC)
  - Handle the closure of a streamer (Ice RPC)
- For clients:
  - Offer REST interface to:
    - Send seeder list
    - Create a new seed on behalf of the client (if no seeder is present)
- **Bonus**:
  - Offer ICE interface to:
    - Send seeder list (Ice RPC)
    - Create a new seed on behalf of the client (if no seeder is present)
    - Create a ICE publisher to announce new or delete seeders (IceStorm) to clients

Client

Mandatory features:

- Get seeder list from portal (either using REST or Ice RPC)

- Use `ffplay` to play video files, e.g., "ffplay tcp://127.0.0.1:10000", to visualize the downloaded file.
- In Java, you can use "ProcessBuilder" for launch a ffplay client.

The client must implement a CLI (PC Desktop) that supports the following commands:

1. `seeder list`: List all seeders
2. `seeder search "keywords"`: List all seeders that match "keywords"
3. `download file`: Download file
4. `list files`: List all files (completed and being downloaded)"
5. `info file`: List all information relating a file, if completely downloaded, full path, size; if being downloaded: file size and neighbor list
6. `play "name"`: Play stream "name" (it should open a ffplay and return to the CLI, so non-blocking)

Bonus:

- Subscribe to portal publisher to receive notifications of new or deleted streams (IceStorm)

Bonus evaluation
Extra credits can be achieved if the student implements the following items:

1. Implement bandwidth control for the bittorrent clients.
2. Implement an Android version of the client (tip: use portable code in the mandatory features, then just provide a simple Android UI).
3. Replace REST with ICE (must use Glacier2, ICEStorm and regular ICE RPC).

Extra Penalization
Penalization will be given if the student not fulfill the following items:

1. Final presentation does not use Google's public cloud.

Video Files:
Big Buck Bunny (Comedy, Animation)
`https://archive.org/details/PopeyeAliBaba`

The Letter, Lego movie [2003] (Comedy, Animation, LEGO), 6m30s
`https://archive.org/details/tl`

Charlie Chaplin's "The Vagabond" [1916] (Comedy), 24m43s
`https://archive.org/details/CC_1916_07_10_TheVagabond`

Night of the Living Dead [1968] (Sci-Fi / Horror), 95m17s
`https://archive.org/details/night_of_the_living_dead`

Popeye the Sailor Meets Ali Baba's Forty Thieves [1937] (Comedy, Animation), 16m58s
`https://archive.org/details/PopeyeAliBaba`

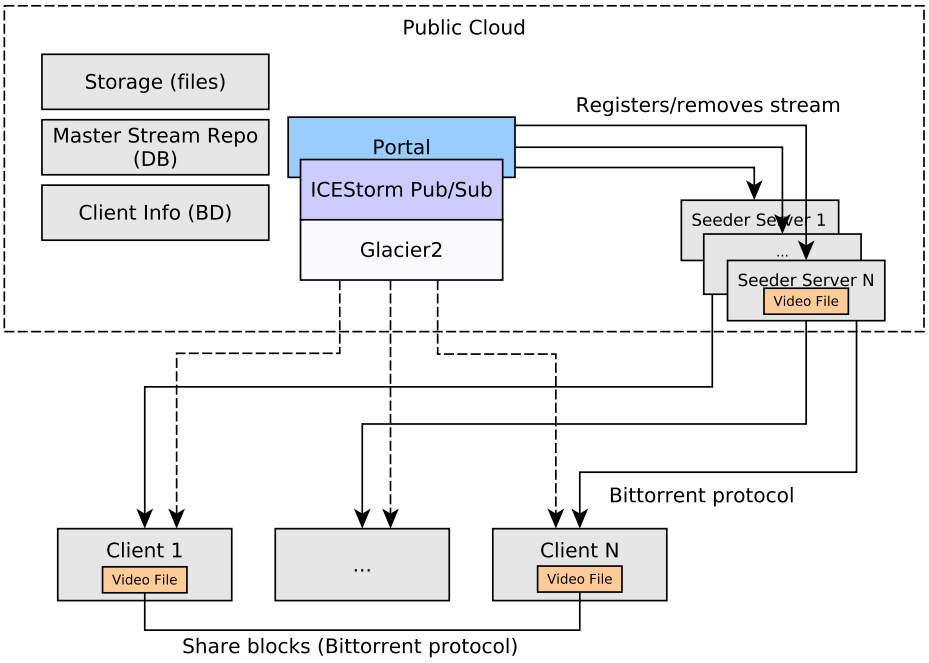


Figure 1: Bonus architecture, replace REST with ICE