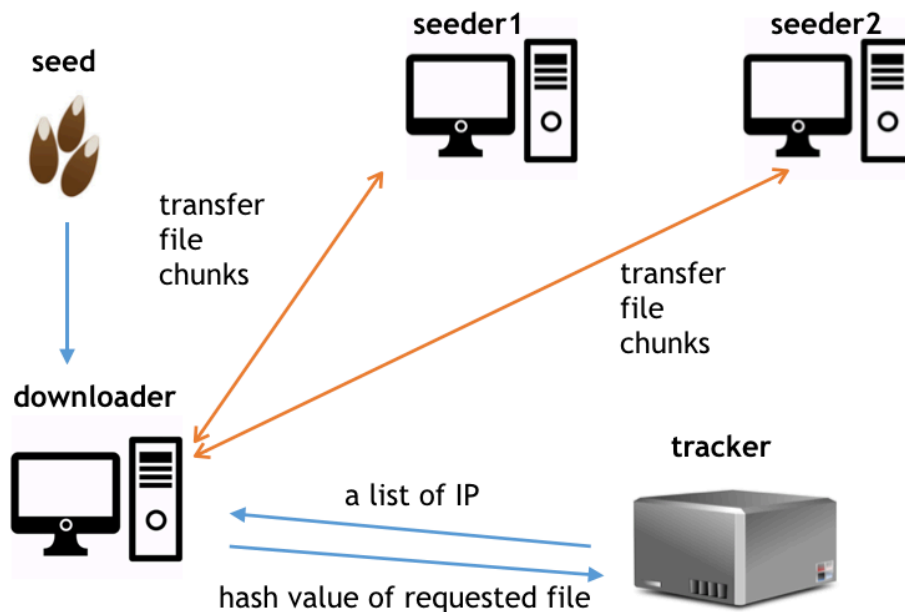


P2P File Share protocol

Haibin He - 11611601
Xinton Lu - 11612325
Siyu Zhang - 11612326

Introduction

- Each node serving as a seeder and downloader
- A tracker maintain status of nodes
- A downloader can download file from multiple seeders at the same time



Requirement:

You are going to design a BT-like protocol.

Following features are required:

- ✓ Find other nodes using tracker
- ✓ Download from multiple nodes if available

You can reuse the assignment of HTTP server with partial downloading.

A possible downloading procedure is:

1. Query the seeder from tracker
2. Download from seeder

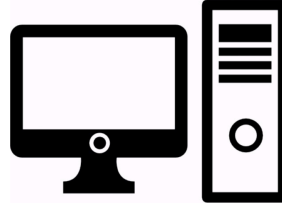
A possible seeding procedure is:

1. Report status to tracker
2. Serving the file, waiting for incoming request

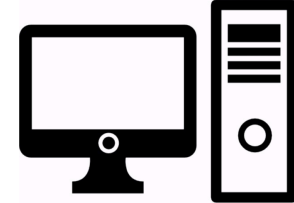
Architecture



seeder1



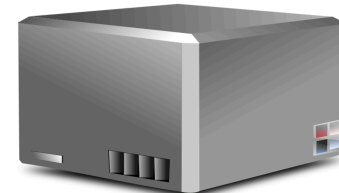
seeder2



downloader



tracker



Implementation

Seed

A seed is an identifier for a file. It includes:

1. file name
2. file length
3. hash value for the whole file
4. a list of hash values for each chunk of the file

```
README.md
4259
db6ecbab8af86ceeb475c93ec12e1f43
fe2a3dbc7deb6a348905c5cd2c86da3a
0a25e9885aa69dad487edd0d7624526f
1b394fa4bce72c8389fbd86dde7b89fe
5007f13101a134aa6beda96853a32de5
b77180407fb4ab1f01f8b48e687efdf4
caf4644c92c9466f4eff75dc5356f5a9
8695b0b6367d4918831f74f2dbfd27b0
7c75425eb31696295ad9a6351202e940
e6aa9fb9c6be71e6ec84519617c191c5
```



Architecture

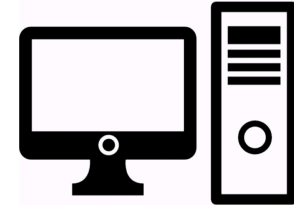
seed



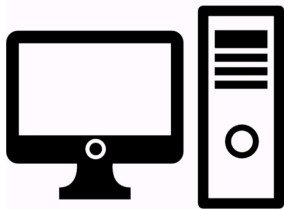
seeder1



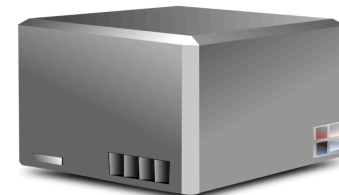
seeder2



downloader



tracker



Architecture

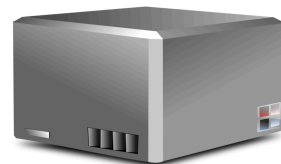
Tracker

- **Node management**

- **Handle periodic status update (including seed list update)** Each node will periodically send message to tracker to update the seeds it has. In our design, each seed has its time to live. If a seed hasn't been update by its seeder for a certain time it will be deleted from the seed list held by the tracker.

- **File sharing**

- **Maintain a seed list of files held by each node in the network.** When a node join the network, it will sends a list of all its nodes to the tracker. When a node send a and successfully downloaded the file from seeders, it will update its file list and send it to the tracker. The tracker need to handle these information and maintain a file list with files and corresponding seeders for query.
- **Handle request for files:** When receiving a request for a specific file, the tracker should check the seed list it maintains, and return the address of all available seeders for sharing the requested file.



Architecture

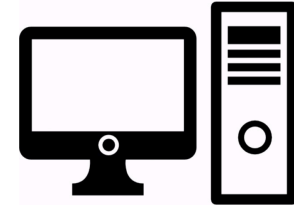
seed



seeder1



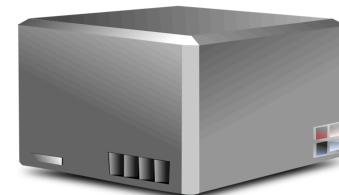
seeder2



downloader



tracker



Architecture

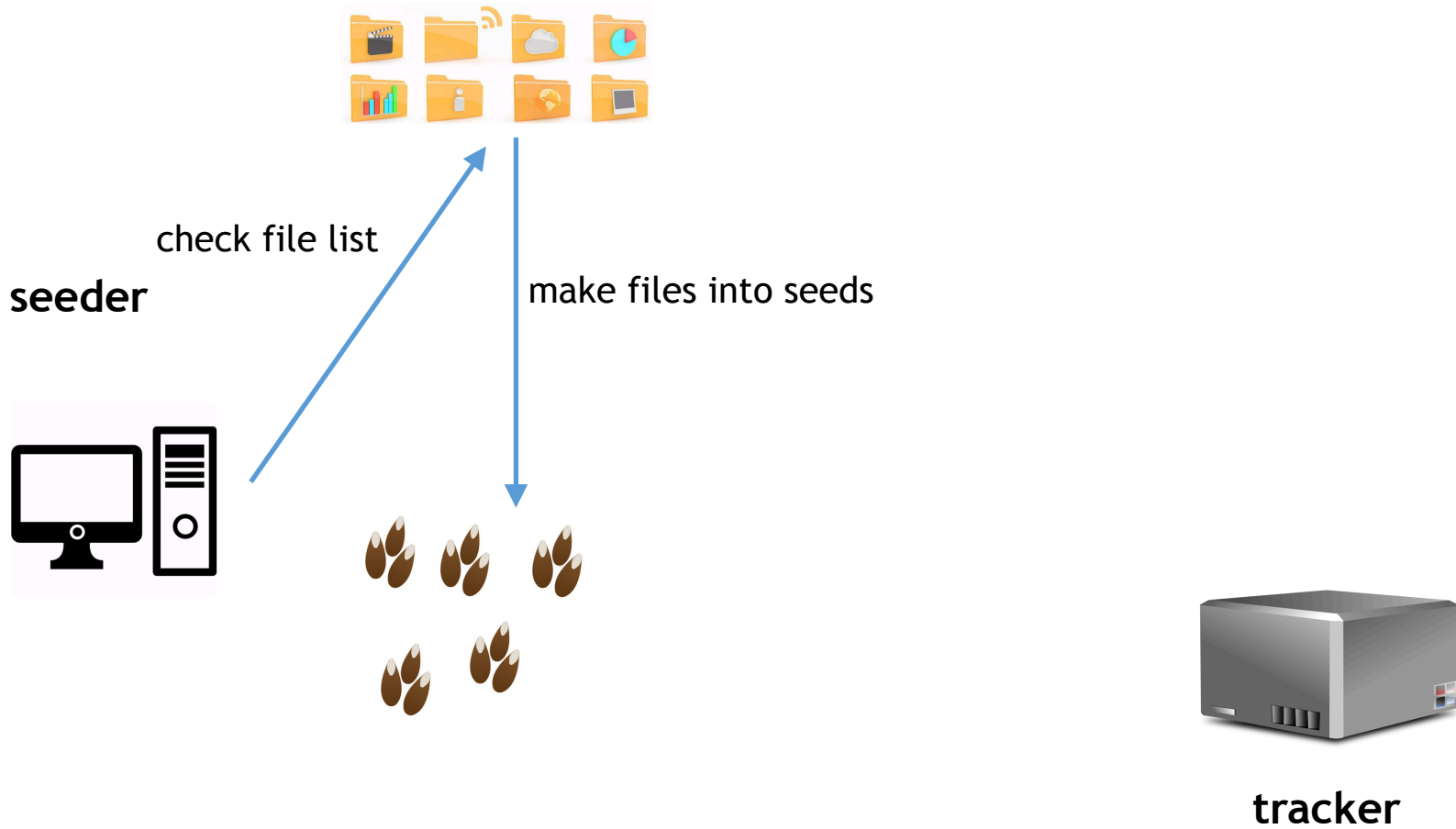
Node

- **status maintenance**
 - **update status** In correspondence to the handle message of tracker, each node acting as a seeder should periodically update all its seeds to the tracker.
- **file requesting**
 - **send download request** A node will send a request for certain file to the tracker and expect a set of available seeds.
- **file sharing**
 - **handle a download request** A node might receive file request form other nodes, and will need to send requested partial to the requesting nodes.



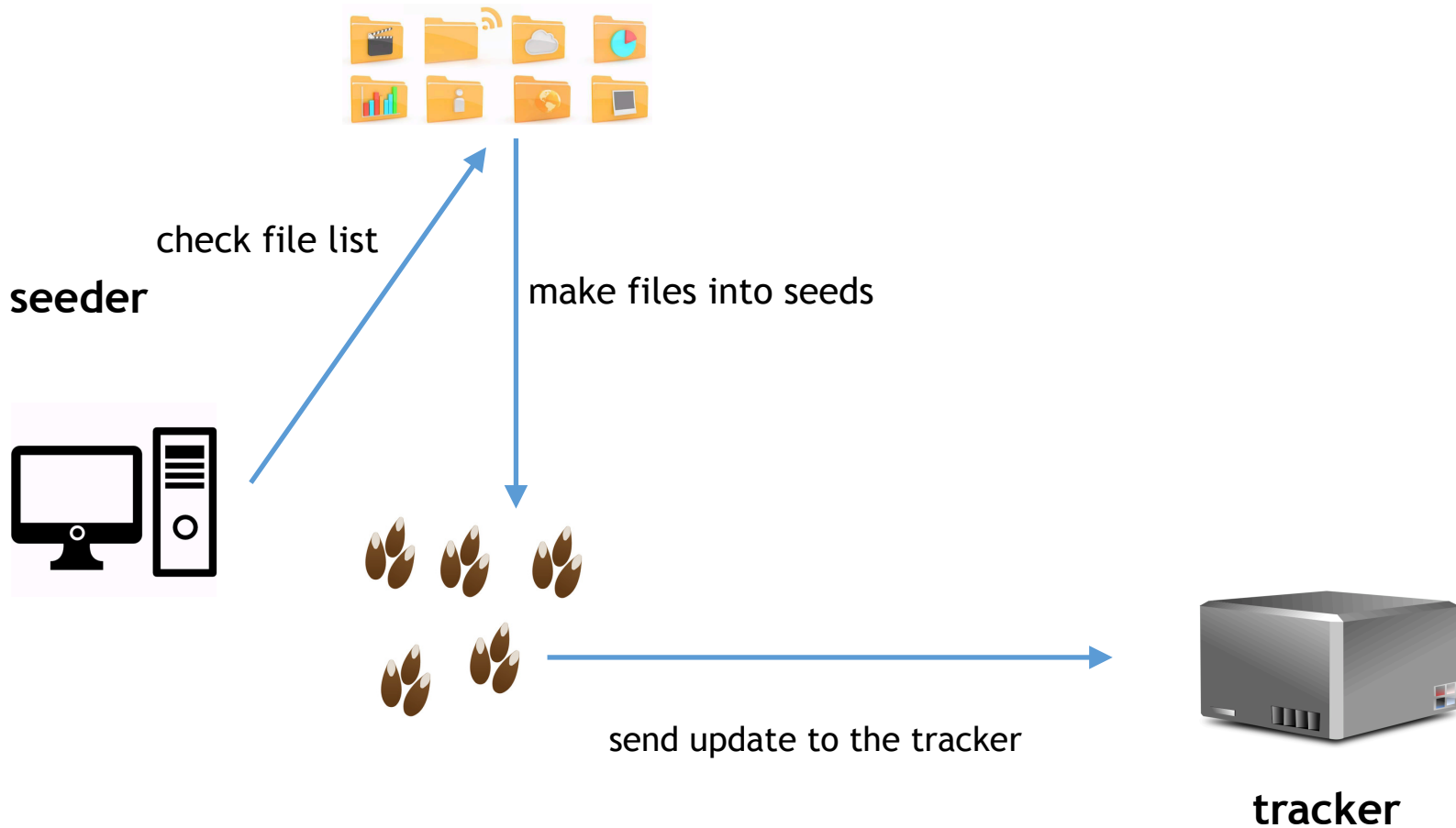
Architecture

Update



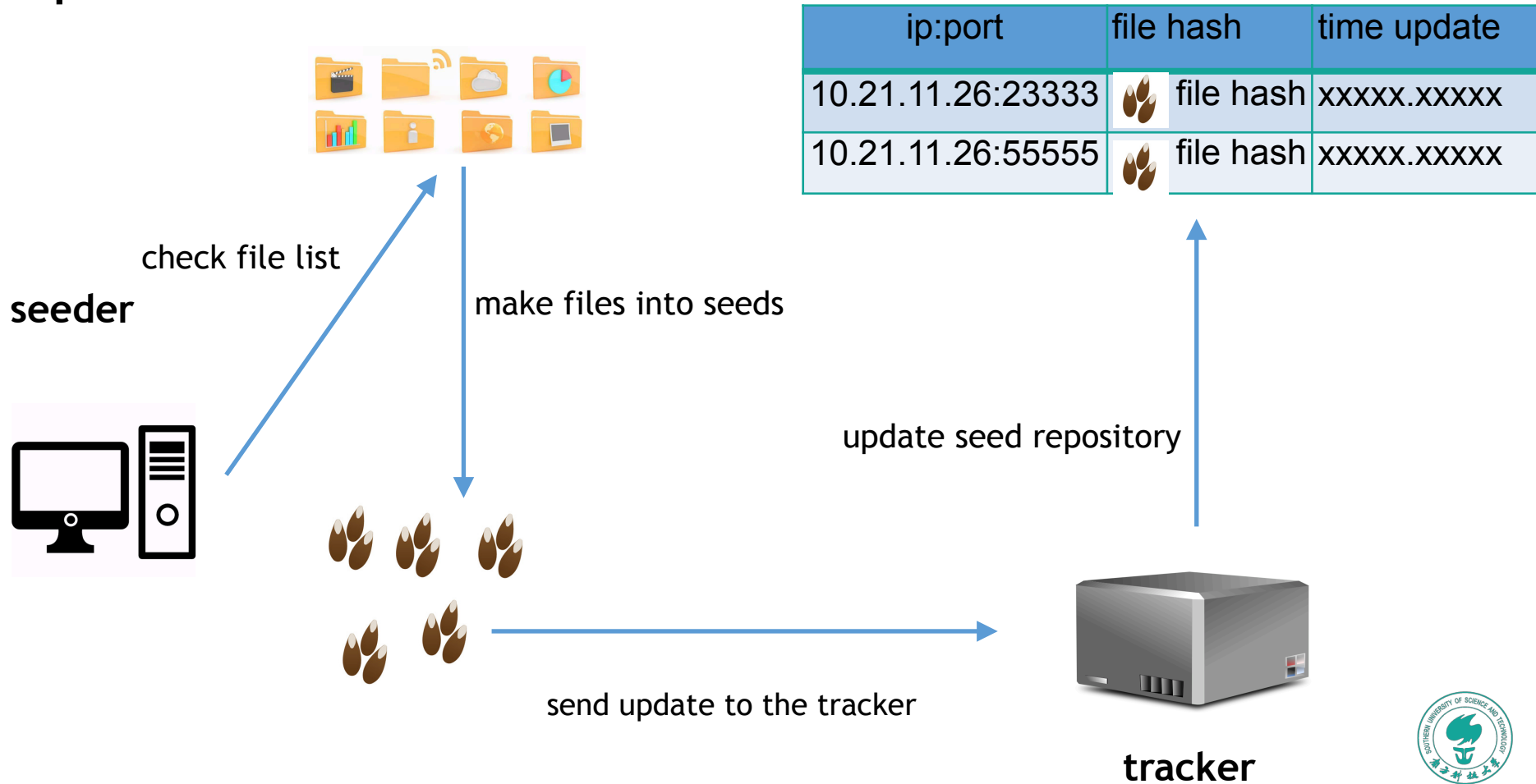
Architecture

Update



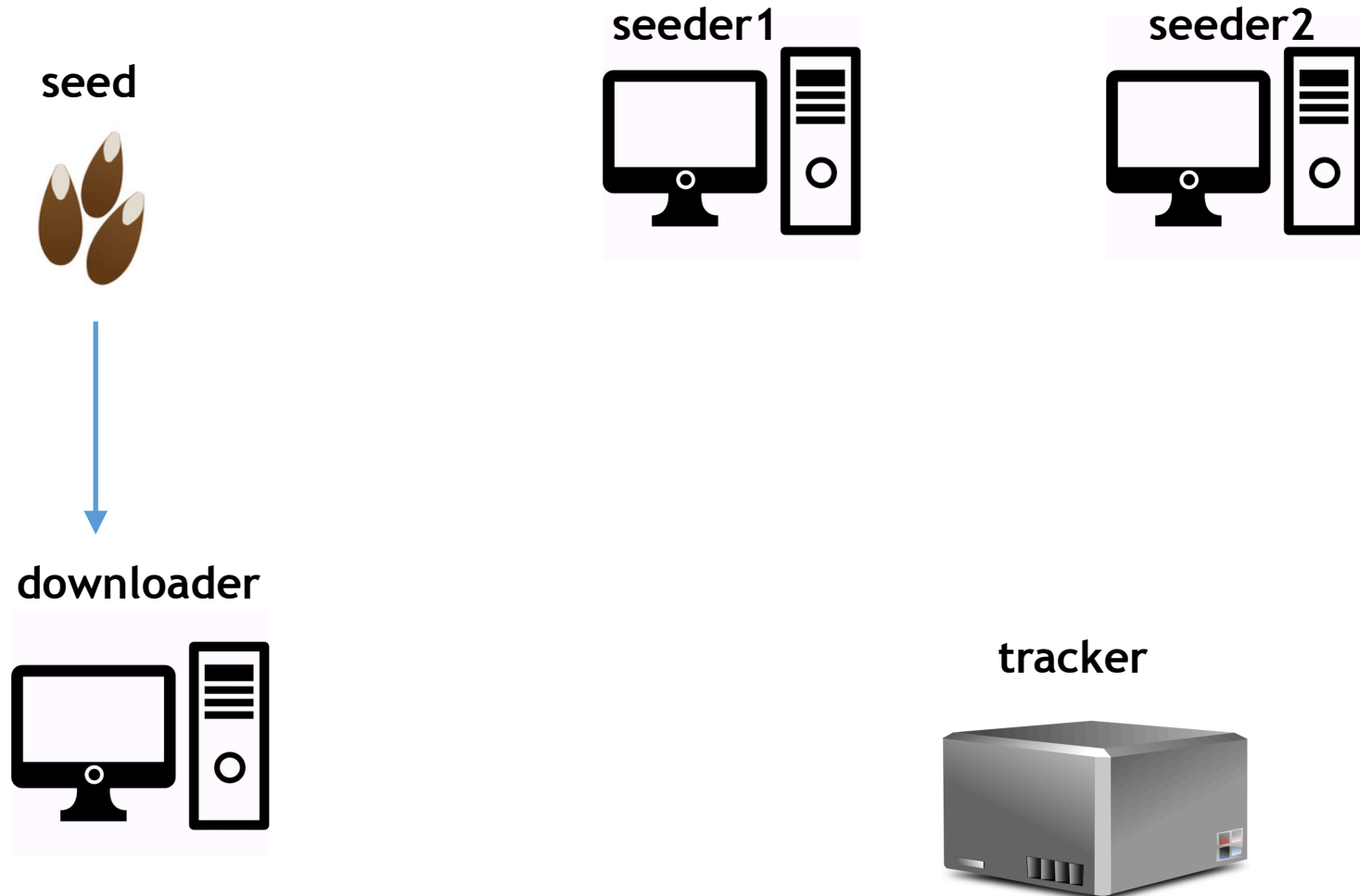
Architecture

Update



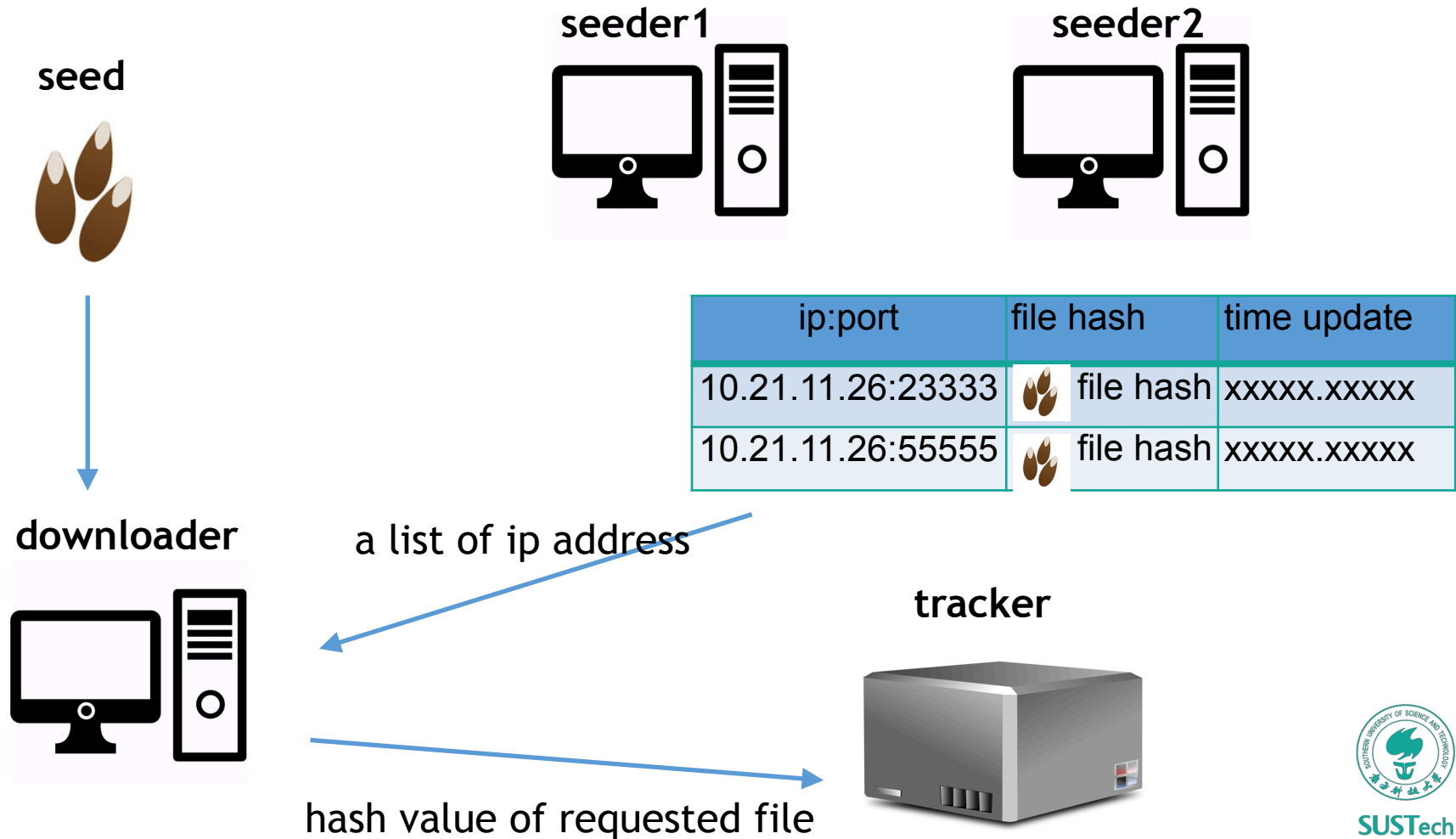
Architecture

Download



Architecture

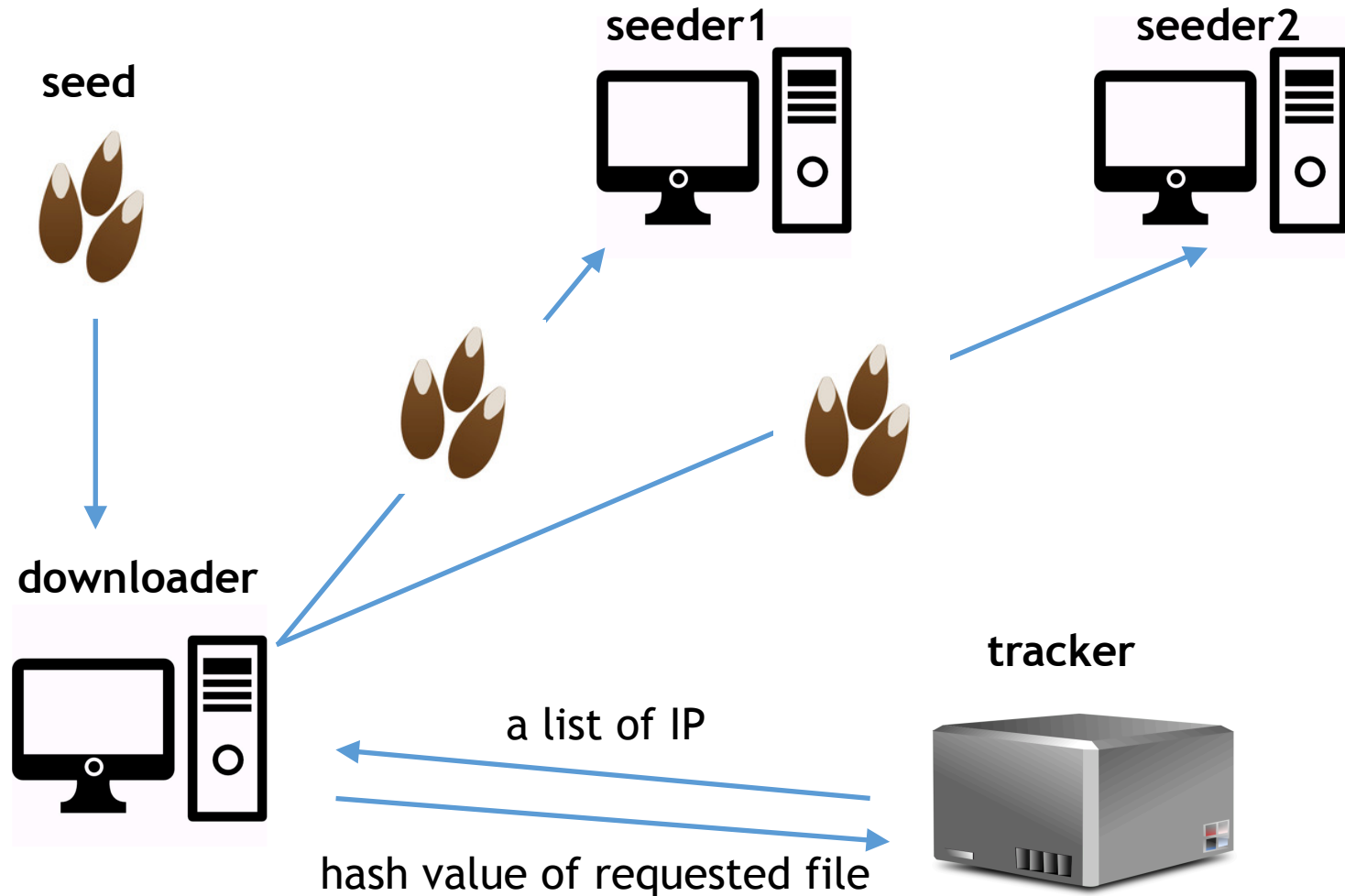
Download



Architecture

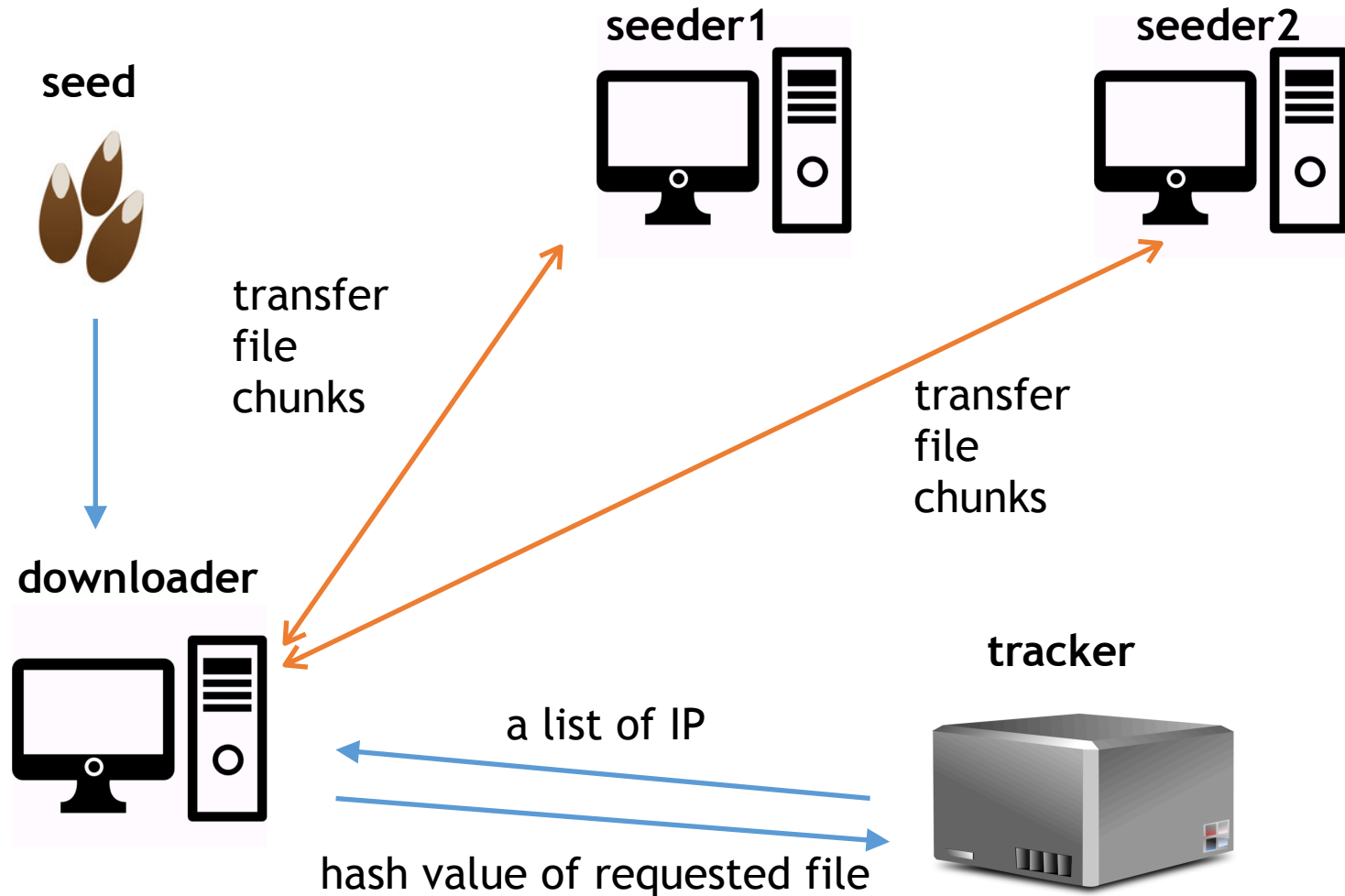
Download

query if the file is still sharing



Architecture

Download



Implementation

Seed

String in the format :

1. file_name
2. file_len
3. big_seed (hash of the whole file)
4. chunk1_hash
5. chunk2_hash
6. ...
7. chunkn_hash

Implementation

Tracker

For **tracker**, two tasks are executed in the event loop:

1. Update seed list: Check timestamp in each seed list, and then deleted seeds that are timeout.
2. Start server and listen for connection: The detail operation depends on the operation code. Specifically
 1. **get_seed**: Means to return a list with all the seeds it currently holds
 2. **Update**: Update the seed list according to update messages
 3. **Query**: Return an address list including requested file.



Implementation

Node

Node as seeder: put a dispatch function to listen to file request of a user and provide corresponding response

Node as client:

1. get the address list with given *seed* from tracker
2. check the availability of the addresses in the address list
3. download the file and assembling.

Video Demo



Q&A



End

Thank you!