

# Tracker bittorrent

Sistemas distribuidos avanzados

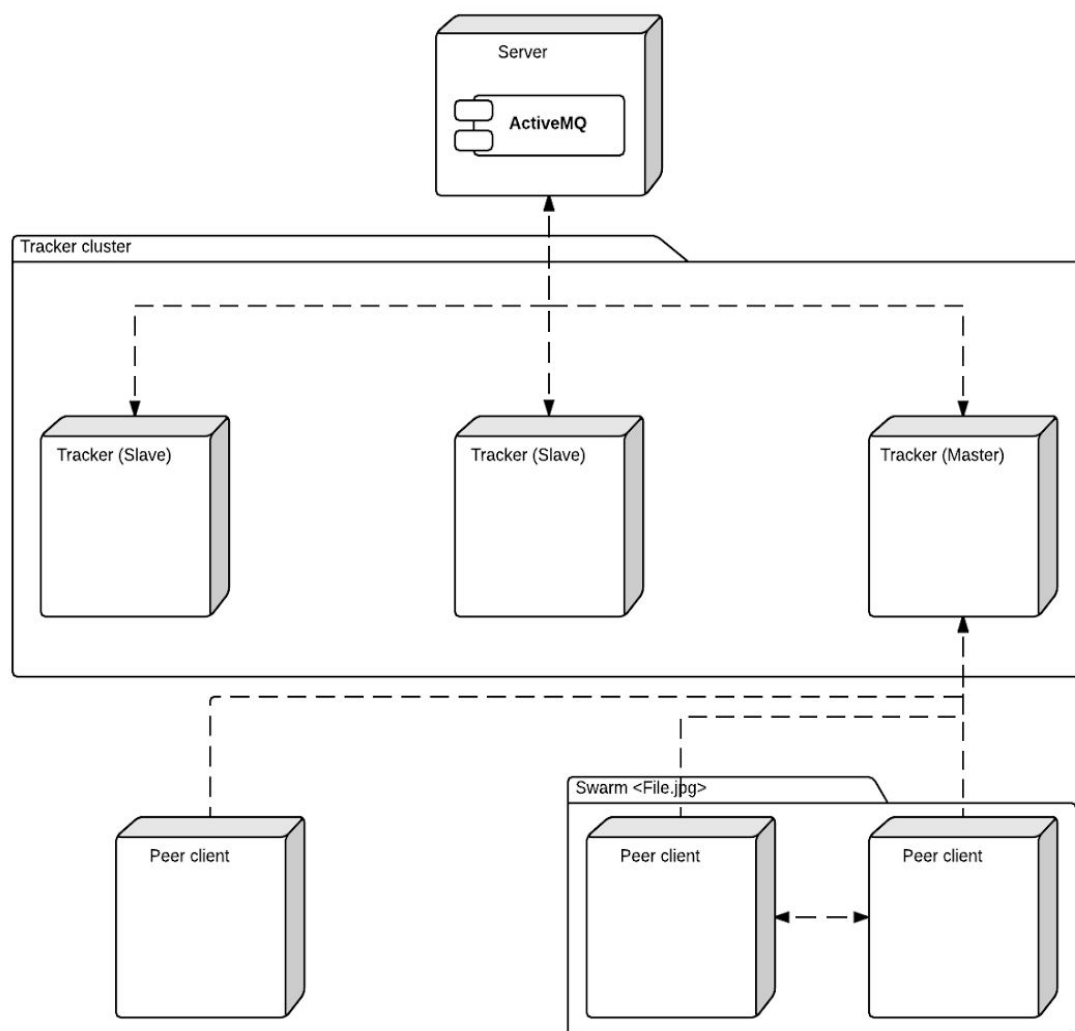
Group 01

Sergio Anguita  
Aitor Brazaola

## 1. Deployment diagram

This project will be focused in the tracker-tracker interaction, to synchronize the different trackers, there will be a central server *Apache ActiveMQ* with *JMS*. Although the following diagram tries to explain how the tracker cluster is arranged, it also contains a few peer clients to show the big picture of the system.

The trackers, can play two different roles, master or slave, at the same time only can exist one master responsible of attending all peer requests and synchronize the persistence operations of the other trackers. Slave trackers should pay attention if the master tracker fails to change their role to master.

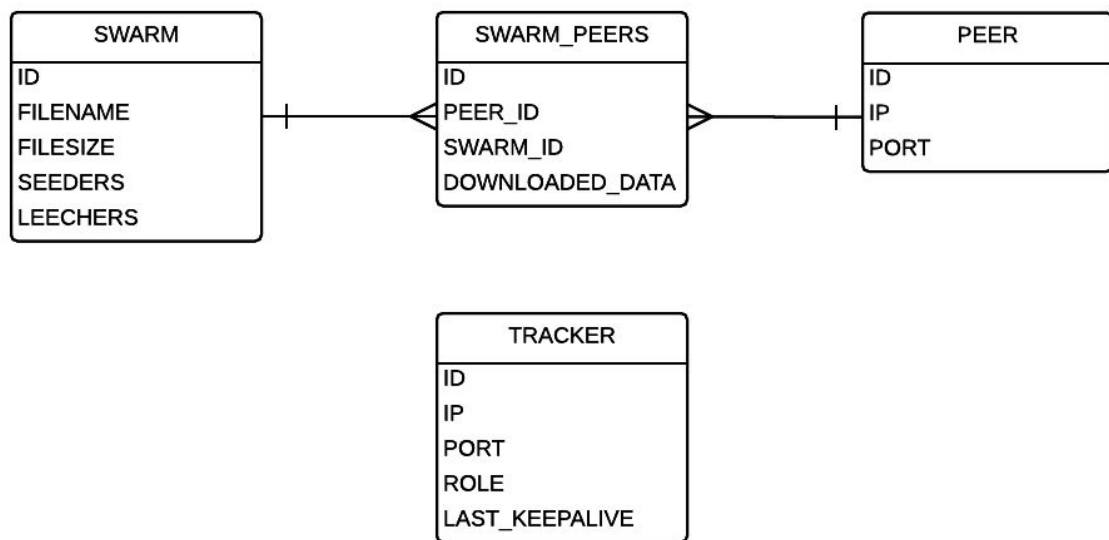


## 2. Entity functional definition

Entity	Functionality
Tracker (Master and slaves)	Send keep alive request to other trackers
	Receive keep alive request to other trackers
	Manage data redundancy of the cluster
Tracker (Master)	Receive 'ready for save' message from other trackers
	Send 'store' command to slave trackers
	Send swarm information to newcomer peer requests
	Check who the slave trackers are
Tracker (Slave)	Send 'store' command from master tracker to save data
	Receive 'store' command from master tracker to save data
	Receive cluster structure information from the master when is recently added
	Check who the master is
Peer	Request new tracker connection
	Receive tracker connection response tracker message
	Send file request information to the tracker
	Receive peer information about the swarm of a file requested from the tracker

### 3. Entity relationship model

In the following diagram will be explained the data architecture to store the relevant information for each tracker, the data persistence technology used will be SQLite because of the ease of use and versatility of only having to manage one .sqlite file for each tracker entity.



Entity	Attribute	Type
SWARM	ID	INT (autoincremental)
	FILENAME	VARCHAR (255)
	FILESIZE	FLOAT
	SEEDERS	INT
	LEECHERS	INT
SWARM_PEERS	ID	INT (autoincremental)
	PEER_ID	INT
	SWARM_ID	INT
	DOWNLOADED_DATA	FLOAT
PEER	ID	INT (autoincremental)
	IP	VARCHAR (15)
	PORT	INT

TRACKER	ID	INT (autoincremental)
	IP	VARCHAR (15)
	PORT	INT
	ROLE	INT
	LAST_KEEPAIVE	LONG

## 4. Interaction model among entities

To ensure an adequate redundancy of the information, all the trackers will send keep alive messages in a constant time interval among them to communicate their uptime status to the others. All trackers must keep track of which one is playing a master role and in case of master tracker is down decide will be the next master.

The time limit for consider a tracker disconnected is 1 seconds, if after 1 seconds the master does not send a keep alive message, the selection of new master tracker will start. The new master selection process will be based on the next higher value of the id integer of the last master, if there are no higher value id, the count will start from 0 until find the first tracker id.

To synchronize the data storage, every slave trackers have to send a small message with the id and the message 'storage ready' to master, and when every tracker that master have in the list of active trackers have sent the message, the master must send the 'store' command to everyone.

## 5. Failure models

Related entity	Failure	Solution
All	JMS Server not responds	No solution available, the entire cluster is down cause trackers can not communicate each others.
Tracker (Master)	Not every slave tracker has sent ready for write message	Wait 5 seconds and check again the list of confirmations received.If there is no confirmation after 5 seconds the tracker is deleted of the list of active trackers.
Tracker (Master)	File not found on tracker nor available swarms	No solution available if selected torrent does not exists
	A peer has no sent information about the status of the download after 1 minute	Probably the peer has been disconnected from the network and should be removed from the list of available peers of a swarm.
Tracker (Slave)	Not received keep alive message from the master	Start the new master selection process.
Tracker (Slave)	Database not received	Repeat database clone request to master
Tracker (Slave)	Database unreadable.	Repeat database clone request to master
Tracker (Both)	No consensus on new Master	Repeat master tracker selectio process until a master is selected
Peer	Unreachable host	Search another host that can provide the same information
Peer	No disk space available	Store in memory until some space is released and warn to the user
Peer	Bad hashing exception	Resend the affected block or file.

## 6. Improvements

Current approach of the tracker system can have some improvements that can enlighten a complete distributed system architecture. Some of those improvements are:

- Real time download folder control for space management.
- Master selection algorithm improvement based on uptime and bandwidth speed parameters.