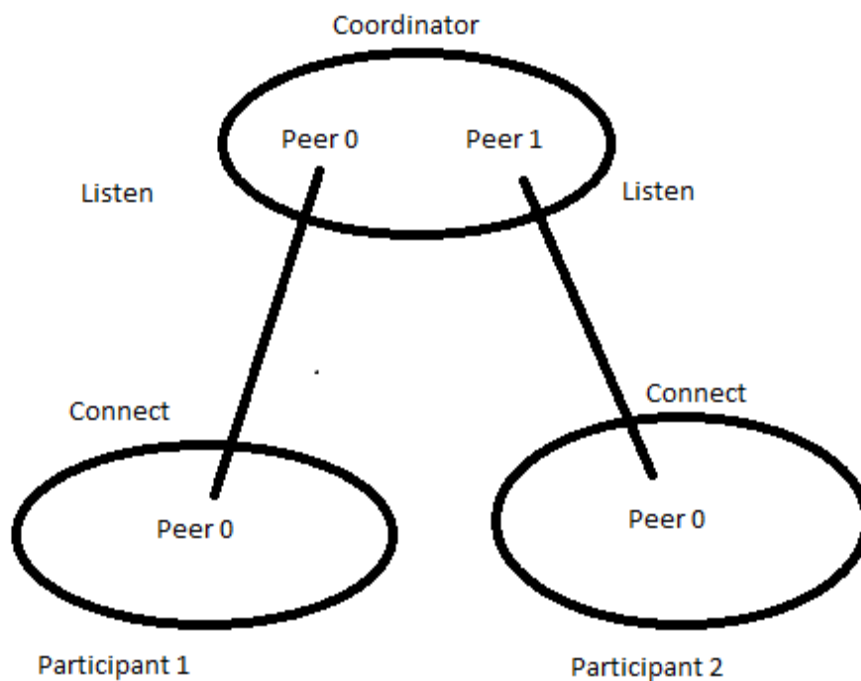


Δημήτρης Πάττας AM:2331

email:dimitris.pattas@gmail.com

Σε συνεργασία με τον Γιώργο Κυμπαρίδη AM:2279

Αρχικά δημιουργείται η σύνδεση με βάση το παρακάτω σχήμα



Οι command-line εντολές:

`python twoPhaseCommit.py (node id) (node ip to connect) (:) (port)`

όπου node id = 0 or 1 or 2

example

`python twoPhaseCommit.py 0 2331`

(ip node 1 : 10.7.4.46)

`python twoPhaseCommit.py 1 10.7.4.46:2331`

(ip node 2 : 10.7.4.49)

`python twoPhaseCommit.py 2 10.7.4.46:2331`

(ip node 3 : 10.7.4.48)

Αρχικά ο Coordinator πολυεκπέμπει 'VOTE\_REQUEST' και ενεργοποιεί ένα timeout ( χρησιμοποιώντας την `reactor.callLater(2, self.timeOut)` ), ταυτόχρονα ψηφίζει VOTE\_COMMIT βάζοντας την ψήφο του στο πίνακα votes []. Κάθε φορά που λαμβάνει ένα μήνυμα το επεξεργάζεται το τοποθετεί στην κατάλληλη θέση του votes []. Αν έχει λάβει όλα τα μηνύματα στέλνει την τελική απόφαση ( GLOBAL\_ABORT ή GLOBAL\_COMMIT ) στους συμμετέχοντες. Αν λήξει το timeout τότε ελέγχει το πίνακα votes [], αν ο πίνακας δεν έχει πληροφορίες για όλους τους κόμβους τότε ο Coordinator στέλνει GLOBAL.

Από την άλλη πλευρά ο κάθε συμμετέχων λαμβάνει ένα μήνυμα και το επεξεργάζεται. Αν είναι 'VOTE\_REQUEST' ψηφίζει με πιθανότητα 60% 'VOTE\_COMMIT', 40% 'VOTE\_ABORT' και ενεργοποιεί ένα timeout. Αν ψηφίσει 'VOTE\_ABORT' γράφει 'GLOBAL\_ABORT' στον δίσκο και μεταβαίνει στην κατάσταση abort. Αν λάβει 'GLOBAL\_COMMIT' το γράφει στο δίσκο. Επιπλέον αν λήξει το ρολόι και ο συμμετέχων δεν έχει λάβει την απόφαση από τον Coordinator πηγαίνει σε κατάσταση abort και γράφει στο δίσκο 'GLOBAL\_ABORT'

The image displays three terminal windows running a two-phase commit protocol. The windows are titled 'scylla.cs.uoi.gr - PuTTY' and show the following logs:

- Participant 1:** Connects to host 10.7.4.46 port 8888. Receives VOTE\_REQUEST from coordinator. Sends VOTE\_COMMIT. Receives GLOBAL\_ABORT from coordinator. Sends VOTE\_ABORT. Receives GLOBAL\_ABORT from coordinator. Sends VOTE\_COMMIT. Receives GLOBAL\_COMMIT from coordinator. Sends VOTE\_COMMIT. Receives GLOBAL\_ABORT from coordinator.
- Coordinator:** Starts factory and builds protocol. Sends VOTE\_REQUEST. Receives VOTE\_ABORT from participant 2. Receives VOTE\_COMMIT from participant 1. Sends GLOBAL\_ABORT. Receives VOTE\_COMMIT from participant 1. Receives VOTE\_ABORT from participant 2. Sends VOTE\_REQUEST. Receives VOTE\_COMMIT from participant 2. Sends GLOBAL\_ABORT. Receives VOTE\_REQUEST. Receives VOTE\_COMMIT from participant 1. Receives VOTE\_ABORT from participant 2. Sends GLOBAL\_ABORT.
- Participant 2:** Connects to host 10.7.4.46 port 8888. Receives VOTE\_REQUEST from coordinator. Sends VOTE\_ABORT. Receives GLOBAL\_ABORT from coordinator. Receives VOTE\_REQUEST from coordinator. Sends VOTE\_ABORT. Receives GLOBAL\_ABORT from coordinator. Receives VOTE\_REQUEST from coordinator. Sends VOTE\_COMMIT. Receives GLOBAL\_ABORT from coordinator. Receives VOTE\_REQUEST from coordinator. Sends VOTE\_COMMIT. Receives GLOBAL\_COMMIT from coordinator. Receives VOTE\_REQUEST from coordinator. Sends VOTE\_ABORT. Receives GLOBAL\_ABORT from coordinator.

Each terminal window has a red box at the bottom labeled 'Participant 1', 'Coordinator', and 'Participant 2' respectively.