Démonstration du projet PDR: étape 2

Commandes à exécuter pour tester IRC

Objectif : L'objectif de ce scénario est d'illustrer le fait que l'on ne passe plus par le serveur pour read lorsqu'on est abonné. Nous voulons illustrer ceci par des affichages ("println") sur le Server en comparant le cas où l'on est abonné et le cas où on ne l'est pas.

Commandes à exécuter :

- Faire un make dans étape 2
- Dans out, lancer le serveur : java Server
- Dans 2 autres terminaux (ou plus), lancer des IRC : java Irc <name>
- Ici, on lance 2 Irc avec les commandes: java Irc Instance1, java Irc Instance2 après avoir lancer le serveur.
- On peut ensuite tester les différents cas où les clients, subscribe, unsubscribe, read ou write.



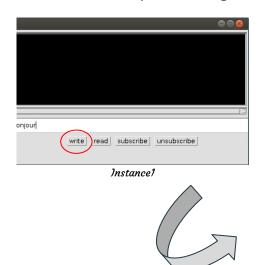
Instance1



Instance2

1er scénario : Cas où l'on n'est pas abonné

- On écrit "Bonjour" sur le premier -> write, après on lit sur le deuxième.
- On remarque l'affichage des phrases entourées ci-dessous en rouge lors du read sur le Server.





Server: reducing lock, activating read mode Server: lock_read executed successfully

Affichage sur le terminal du Server

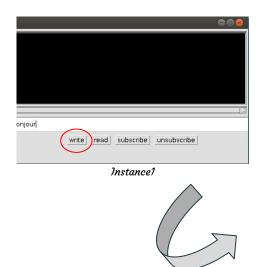
Server: lookup demand received Server: create demand received Server: register demand received Server: lookup demand received

Server : lock_write demand received Server : lock write executed successfully

Server: publish demand received
Server: lock read demand received

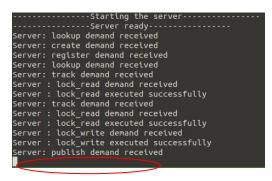
2ème scénario : Cas où l'on est abonné

- On écrit "Bonjour" sur le premier -> write, après on lit sur le deuxième.
- On remarque qu'il n'y a pas de print Server (non passage par le serveur).





Instance2



Affichage sur le terminal du Server

Conclusion

Affichage sur le terminal du Server (cas non abonné)

```
Server: lookup demand received
Server: create demand received
Server: create demand received
Server: lookup demand received
Server: lookup demand received
Server: track demand received
Server: track demand received
Server: lock_read demand received
Server: lock_read executed successfully
Server: track demand received
Server: lock_read executed successfully
Server: lock_read demand received
Server: lock_write demand received
Server: lock_write executed successfully
Server: lock_write executed successfully
Server: publish demand received
```

Affichage sur le terminal du Server (Cas abonné)

Quand on est abonné, on ne passe effectivement pas par le Server. On a aussi affiché dans le cas de l'abonnement, un message "update" sur le Client pour signaler la mise à jour de la copie locale de l'objet.