

Distributed Systems

Name : Seila Moro Herrero
Group : N/A

1. Tests

Test N1

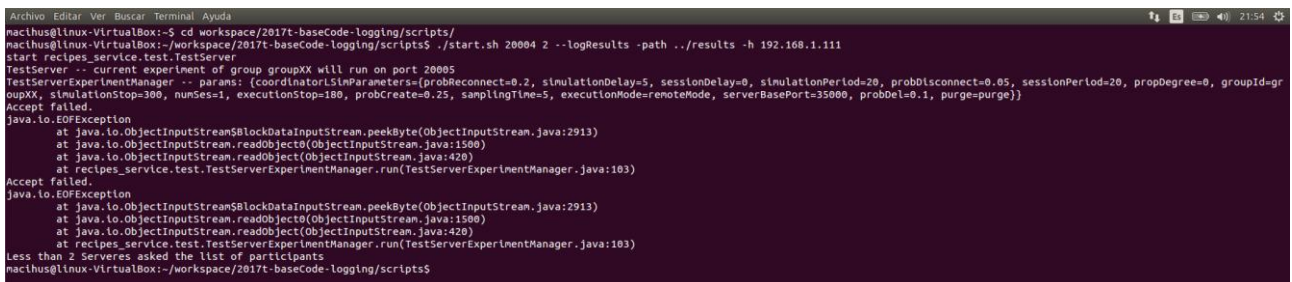
- Descripción: se prueba la “phase 4.1” de forma local, con los siguientes parámetros:
 - Scale: 15 servidores
 - dynamicity: 0.05 (desconexión) y 0.2 (conexión)
 - accelerate propagation: 20 segundos.
 - level of activity generation: 0.25 (crear), 0.1 (eliminar)

Test N2

- Descripción: se prueba la “phase 4.1” de forma local, con los siguientes parámetros:
 - Scale: 5 servidores
 - dynamicity: 0.10 (desconexión) y 0.4 (conexión)
 - accelerate propagation: 10 segundos.
 - level of activity generation: 0.50 (crear), 0.2 (eliminar)

Pruebas en entorno distribuido:

Para las pruebas de entorno distribuido se utilizan dos maquinas virtuales, con el programa VirtualBox, para emular el entorno, ya que no se dispone de toda la infraestructura física necesaria para estas pruebas.



```
Archivo Editor Ver Buscar Terminal Ayuda
maci@linux-VirtualBox:~$ cd workspace/2017t-baseCode-logging/scripts/
maci@linux-VirtualBox:~/workspace/2017t-baseCode-logging/scripts$ ./start.sh 20004 2 --logResults -path ../results -h 192.168.1.111
start_recipes_service_test.TestServer
TestServer -- current experiment of group groupXX will run on port 20005
TestServerExperimentManager -- params: (coordinatorSLNParameters={probReconnect=0.2, simulationDelay=5, sessionDelay=0, simulationPeriod=20, probDisconnect=0.05, sessionPeriod=20, propDegree=0, groupId=groupXX, simulationStop=300, numSes=1, executionStop=180, probCreate=0.25, samplingTime=5, executionMode=remoteMode, serverBasePort=35000, probDel=0.1, purge=purge})
Accept failed.
java.io.EOFException
    at java.io.ObjectInputStream$BlockDataInputStream.peekByte(ObjectInputStream.java:2913)
    at java.io.ObjectInputStream.readObject0(ObjectInputStream.java:1506)
    at java.io.ObjectInputStream.readObject(ObjectInputStream.java:420)
    at recipes_service_test.TestServerExperimentManager.run(TestServerExperimentManager.java:103)
Accept failed.
java.io.EOFException
    at java.io.ObjectInputStream$BlockDataInputStream.peekByte(ObjectInputStream.java:2913)
    at java.io.ObjectInputStream.readObject0(ObjectInputStream.java:1506)
    at java.io.ObjectInputStream.readObject(ObjectInputStream.java:420)
    at recipes_service_test.TestServerExperimentManager.run(TestServerExperimentManager.java:103)
Less than 2 Servers asked the list of participants
maci@linux-VirtualBox:~/workspace/2017t-baseCode-logging/scripts$
```

No ha sido posible realizar estas pruebas, debido al número de servidores. Ya que no es posible emular más equipos por falta de recursos.

4. Conclusions

A mayor número de servidores son necesarias más sesiones de anti-entropía para sincronizar todos los servidores.

El número de desconexiones y conexiones también afecta, ya que si un servidor está desconectado mucho tiempo puede perder sesiones de anti-entropía, haciendo que ya no esté totalmente dessincronizado con el resto de servidores.

La periodicidad de las sesiones de anti-entropía es un factor determinante. Ya que, si este parámetro es muy elevado habrá pocas sesiones y en cada una de ellas se intercambiarán muchas operaciones, además los servidores estarán dessincronizados mucho tiempo. Pero si este parámetro es muy bajo, se realizarán muchas sesiones y se puede saturar los recursos del sistema, e incluso haber muchas sesiones sin intercambio de operaciones.

El nivel de generación de actividad afecta al número de operaciones que se intercambian los servidores en cada sesión. Si es muy bajo podría haber muchas sesiones sin intercambio de operaciones. Y si es muy alto, en algunas sesiones hay que intercambiar muchas operaciones para sincronizar los servidores.

Lo óptimo para este tipo de pruebas sería utilizar un entorno distribuido, aunque no ha sido posible disponer del mismo. Entiendo que en este tipo de entornos también afectaría la infraestructura a las sesiones de anti-entropía. Ya que, por ejemplo, la velocidad de intercambio de mensaje dependerá de la red, del uso de la misma, etc.