

Dorm test de performance

Guide d'installation :

- Décompressez le fichier tar.gz.
- Mettez à jour la variable d'environnement GRINDERPATH dans le fichier `{ $GRINDER_HOME }/bin/setGrinderEnv.sh` avec le bon chemin.
- Exécutez le script `{ $GRINDER_HOME }/bin/startConsole.sh` :
 `~/ cd $GRINDER_HOME`
 `~/grinder-3.5 bin/startConsole.sh`

Normalement, s'il n'y a pas d'erreur vous devriez avoir la console de grinder qui s'affiche.

- Exécutez le script (dans une autre console) `{ GRINDER_HOME }/bin/startAgent.sh` :
 `~/ cd $GRINDER_HOME`
 `~/grinder-3.5 bin/startAgent.sh`

Si l'agent c'est bien lancé vous devriez avoir un message d'attente qui s'affiche :

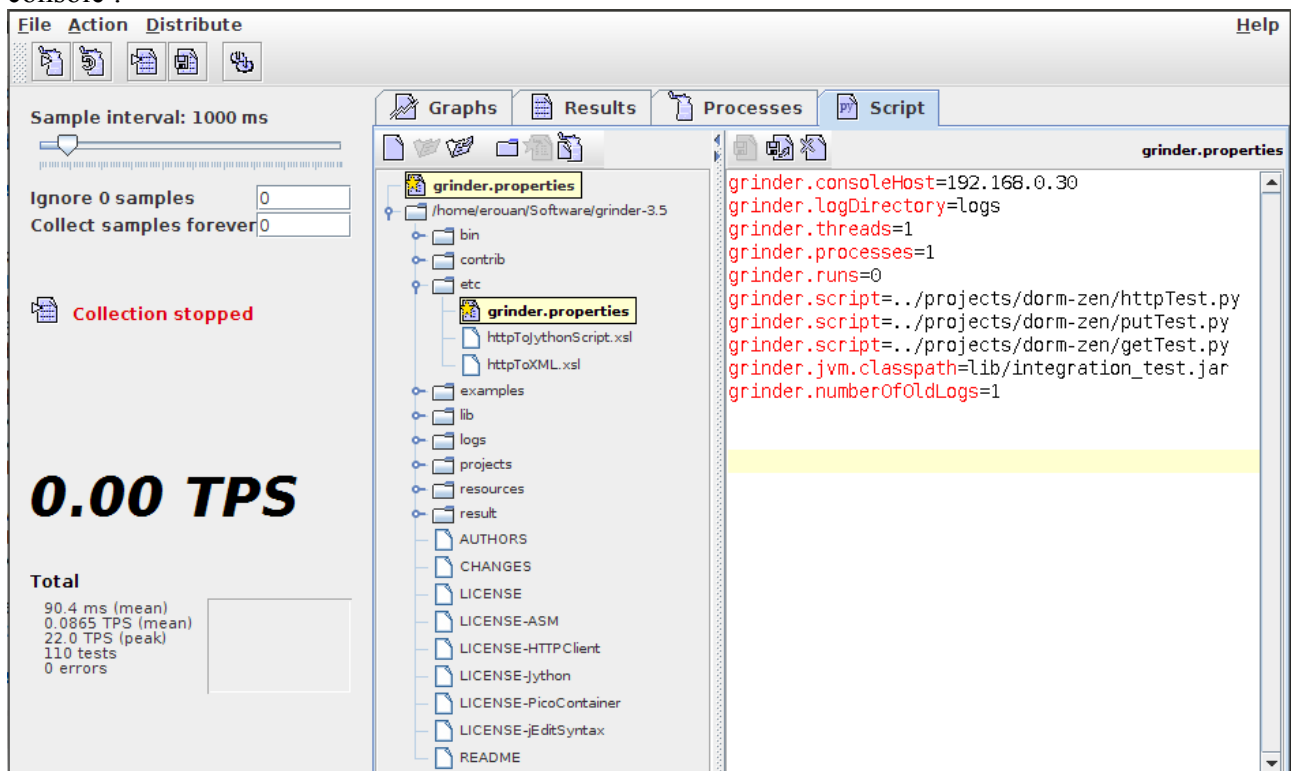
```
erouan@Erouan-pc:~/Software/grinder-3.5$ bin/startAgent.sh
9/23/11 2:43:01 PM (agent): The Grinder 3.5
9/23/11 2:43:01 PM (agent): connected to console at /192.168.0.30:6372
9/23/11 2:43:01 PM (agent): waiting for console signal
```

Grinder est maintenant installé et prêt à être utilisé.

Configuration :

Grinder.properties :

Le fichier `grinder.properties` sert à configurer les agents qui vont exécuter nos tests. Il est situé à ce chemin : `{ $GRINDER_HOME }/etc/grinder.properties`. Vous pouvez l'éditer directement dans la console :



Vérifiez bien que la librairie « integration_test.jar » est intégrée dans le classpath, car c'est dans ce .jar que sont les classes qui permet d'écrire le fichier de résultat des tests. Elles contiennent aussi les URL à tester.

Pour sélectionner le test à effectuer vous devez renseigner le fichier du script Jython à exécuter. Comme ci-dessus vous pouvez en indiquer plusieurs, mais un seul (le dernier) sera exécuté.

Script Jython de Dorm :

Pour le projet Dorm deux scripts Jython ont été réalisés pour tester les web services du plugin maven. Ils se trouvent à ce chemin : { \$GRINDER_HOME }/projects/dorm-zen/. Le script « getTest.py » effectue 6 requêtes GET HTTP et le script « putTest.py » effectue 6 requêtes PUT HTTP. Ces deux scripts émulent un client maven.

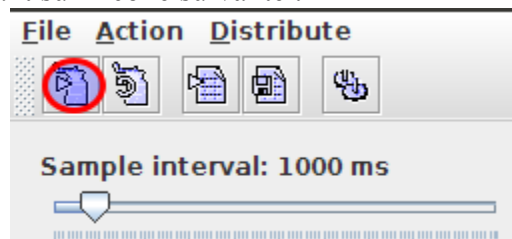
maven_get_requetes et maven_put_requetes :

Ces deux fichiers se trouvent à la racine de l'archive { \$GRINDER_HOME }/lib/integration_test.jar. Ils fournissent les informations nécessaires à l'assemblage des requêtes HTTP. Vous pouvez donc définir l'URL des 6 requêtes à effectuer ainsi que le nom de l'hôte, son port et l'URL de base. Voici un exemple :

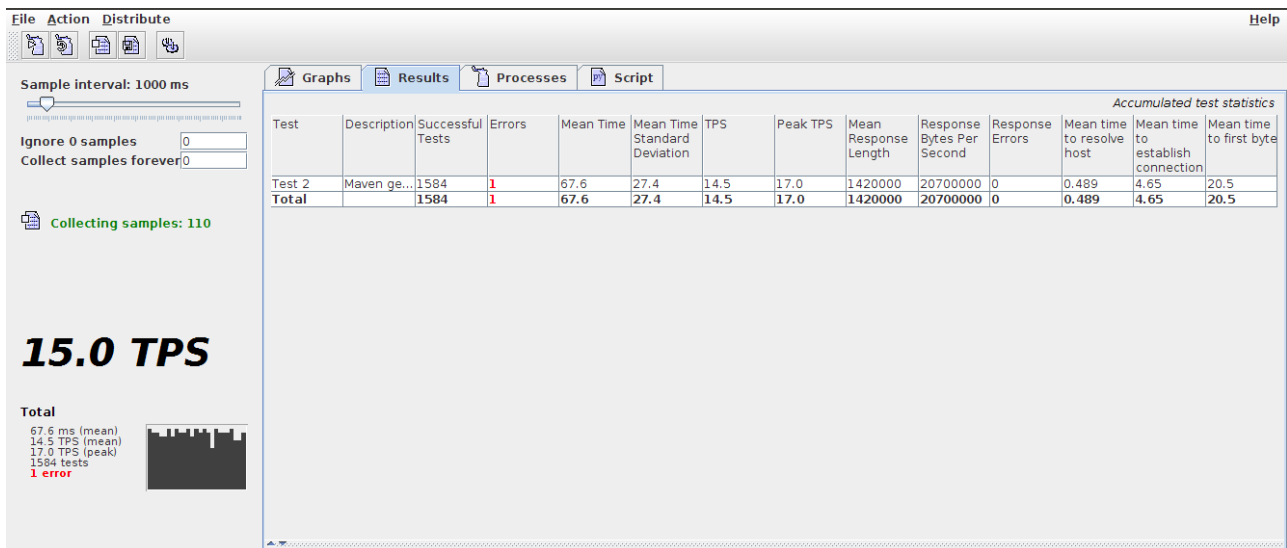
```
{
  "configuration":{
    "target.hostname":"192.168.182.129",
    "target.port":"8081",
    "target.baseUrl":"artifactory/rep01",
    "proxy.hostname":"192.168.182.1",
    "proxy.port":"8008",
    "proxy.active":true
  },
  "resources":[
    {
      "type": "getResource",
      "pomUrl": "org.apache.commons/commons-io/1.0.0/commons-io-1.0.0.pom",
      "pomSha1Url": "org.apache.commons/commons-io/1.0.0/commons-io-1.0.0.pom.sha1",
      "pomMd5Url": "org.apache.commons/commons-io/1.0.0/commons-io-1.0.0.pom.md5",
      "jarUrl": "org.apache.commons/commons-io/1.0.0/commons-io-1.0.0.jar",
      "jarSha1Url": "org.apache.commons/commons-io/1.0.0/commons-io-1.0.0.jar.sha1",
      "jarMd5Url": "org.apache.commons/commons-io/1.0.0/commons-io-1.0.0.jar.md5",
      "expectedResult": {
        "httpCode": 404
      }
    }
  ]
}
```

Exécution des tests :

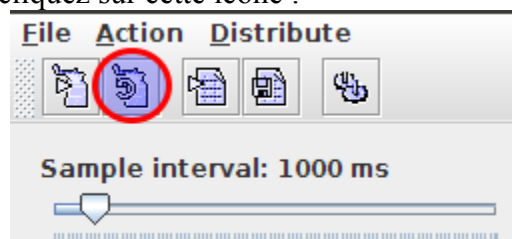
- Démarrez la console et au moins un agent.
- Distribuez les fichiers aux agents en cliquant sur Distribute/Distribute files :
- Démarrez le test en cliquant sur l'icone suivante :



- Lorsque le test est commencé les résultats commencent à arriver dans l'onglet Result :



- Pour arrêter les tests, cliquez sur cette icône :



Les résultats des tests sont le dossier “result” du répertoire de votre agent. Si vous avez lancé l’agent sur la même machine que la console son répertoire est à ce chemin : `{ $GRINDER_HOME } / { nom_du_pc } -file-store/current/result/test.csv`.

N’oubliez pas de supprimer le fichier de résultat avant de lancer chaque test !

Changer de Dao dans dorm :

Pour changer de Dao dans dorm il faut modifier la classe `DormGuiceServletContextListener` qui est dans le module webapp. Dé-commenter la ligne qui correspond au Dao choisi. Pour le moment il y a trois choix :

- Le Dao JDBC (ligne : `modules.add(new DormCoreJdbcModule());`)
- Le Dao NEO4J (ligne : `modules.add(new DormCoreNeo4jModule());`)
- Le Dao NUXEO (linge : `modules.add(new DormCoreNuxeoModule());`)

Attention vous ne pouvez choisir qu’un seul dao à la fois.

Compilez et packagez le projet avec Gradle. Ensuite déployez le sur serveur tomcat.