Introduction to Benchmark

Database Systems
DataLab, CS, NTHU
Spring, 2018

Outline

- Benchmark Project
  - Introducing Benchmark Project
  - Starting Up Server for Benchmarking
  - Setting Benchmark Configurations
  - Running Benchmark Client

Outline

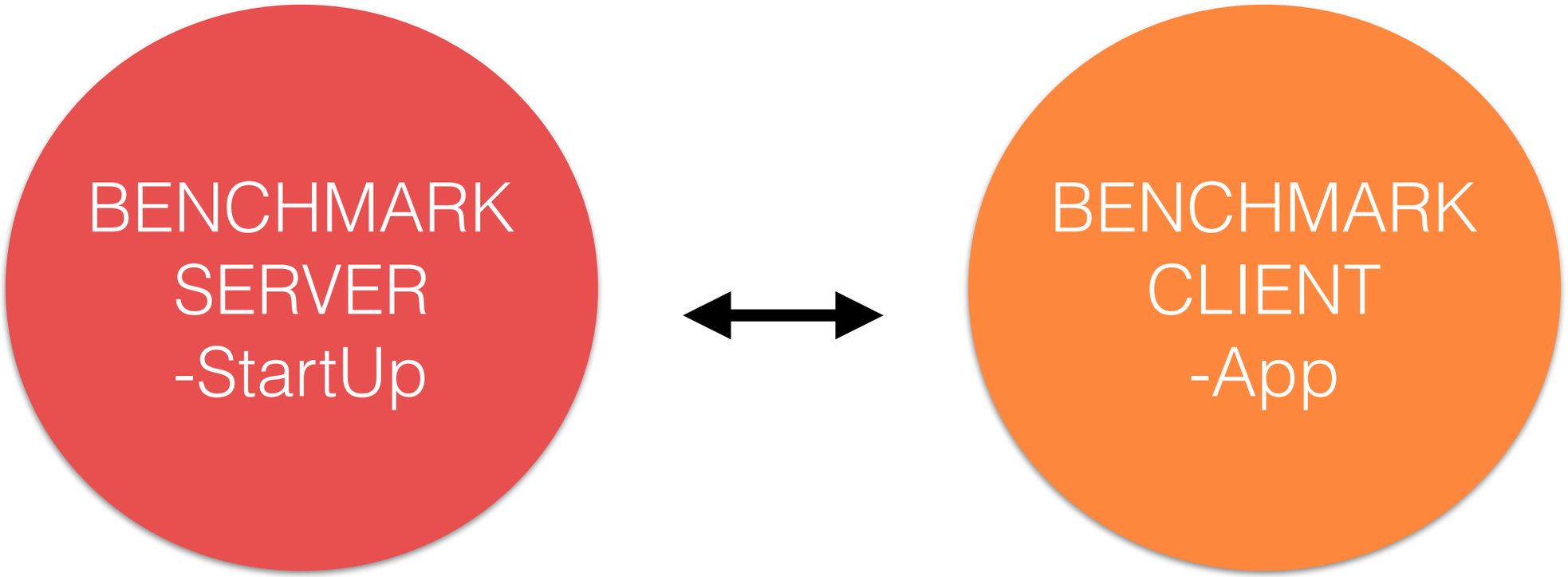
- Benchmark Project
  - Introducing Benchmark Project
  - Starting Up Server for Benchmarking
  - Setting Benchmark Configurations
  - Running Benchmark Client

Benchmark Project

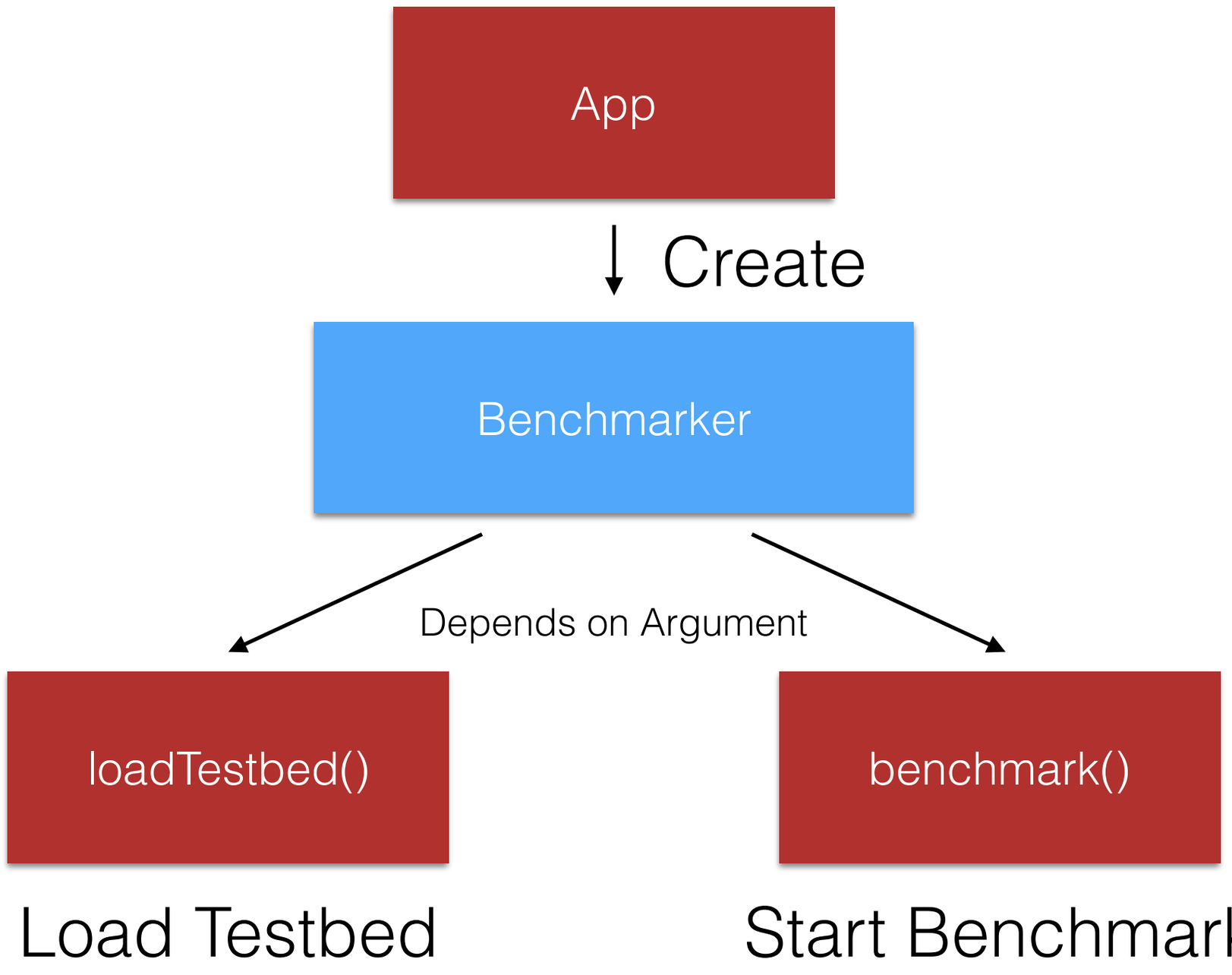
- Benchmark project is a special client project for automatically benchmarking
- It contains a testbed loader and a testing process
- It also has a lot of adjustable testing parameters

[Icon Source]

Two Main Methods



Start a Client



Loading Testbed

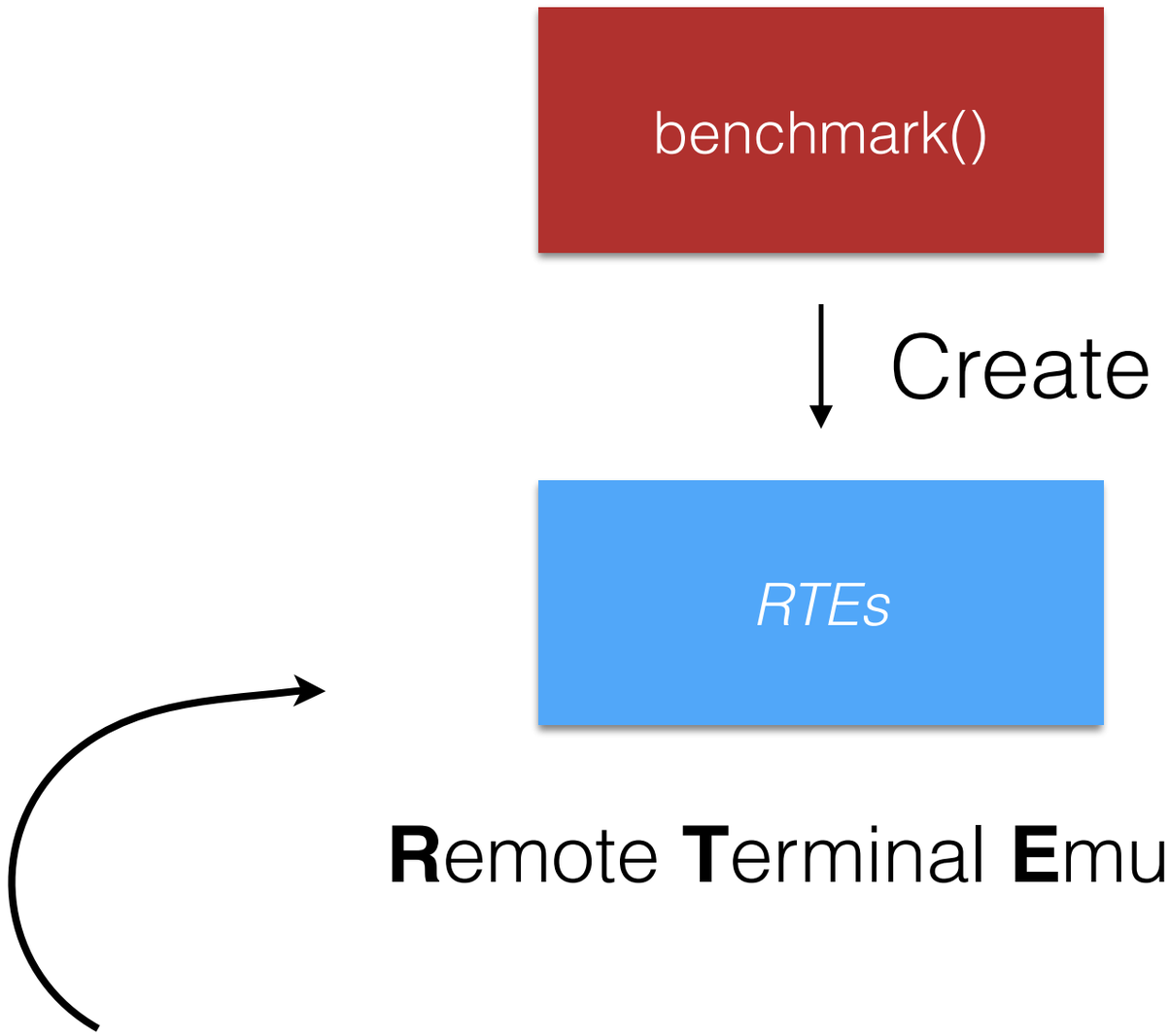
loadTestbed()

Connect to server and execute two stored procedures:

SchemaBuilder

TestbedLoader

Starting Benchmark



Server & Client

RTE Life Cycle

RTEs

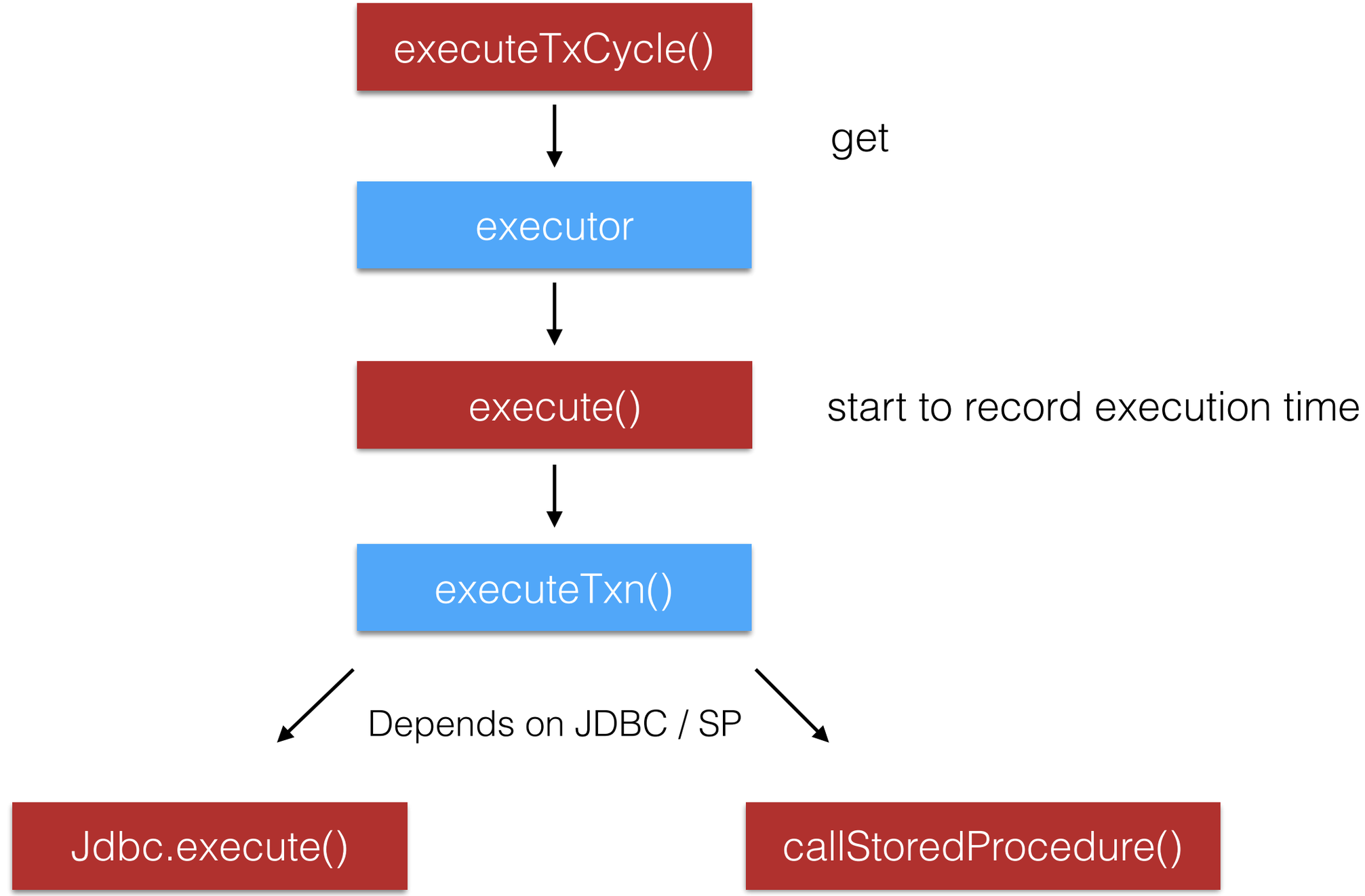
run()

executeTxCycle()

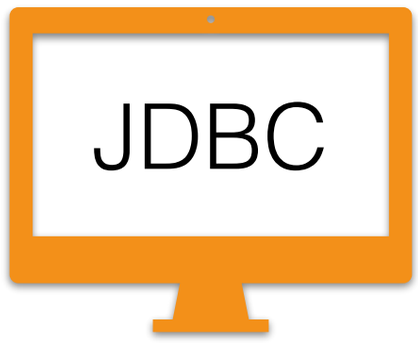
stopBenchmark()

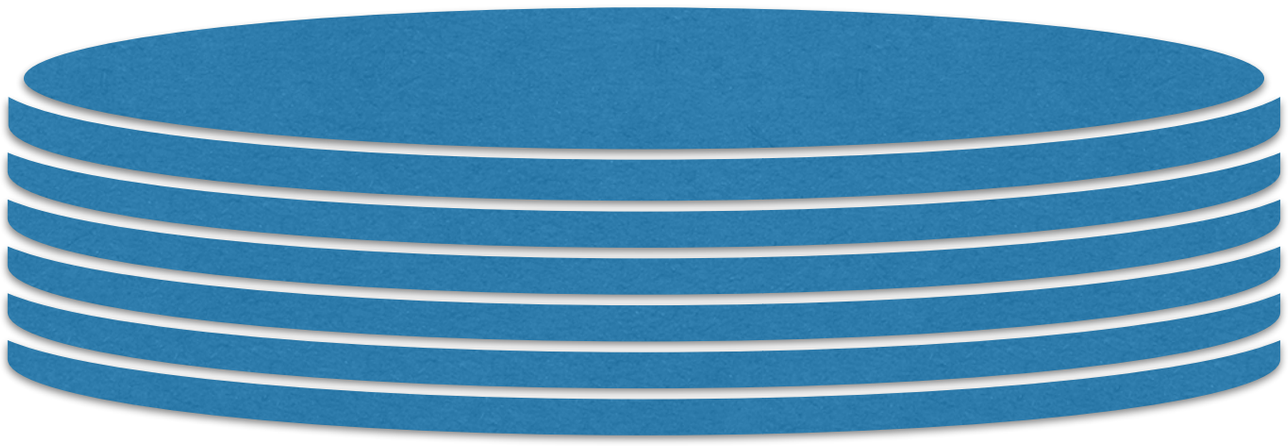
end

Execute Tx

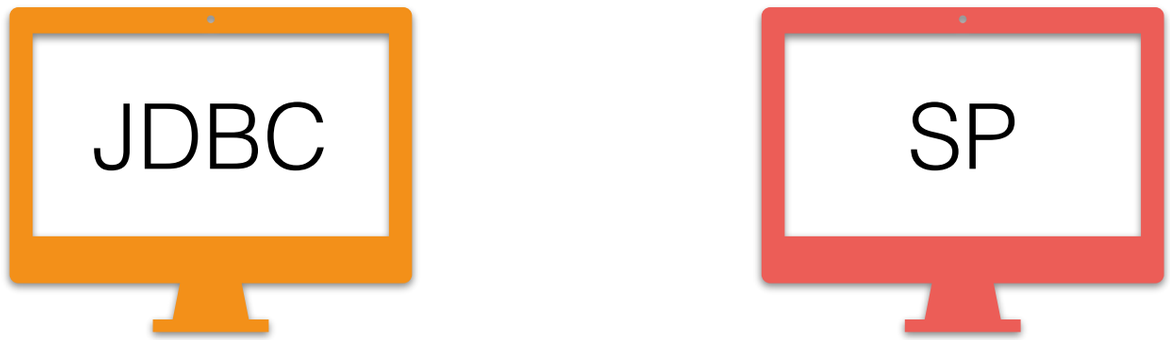

executeTxCycle()
get
executor
execute()
start to record execution time
executeTxn()
Depends on JDBC / SP
Jdbc.execute()
callStoredProcedure()

JDBC / SP ?

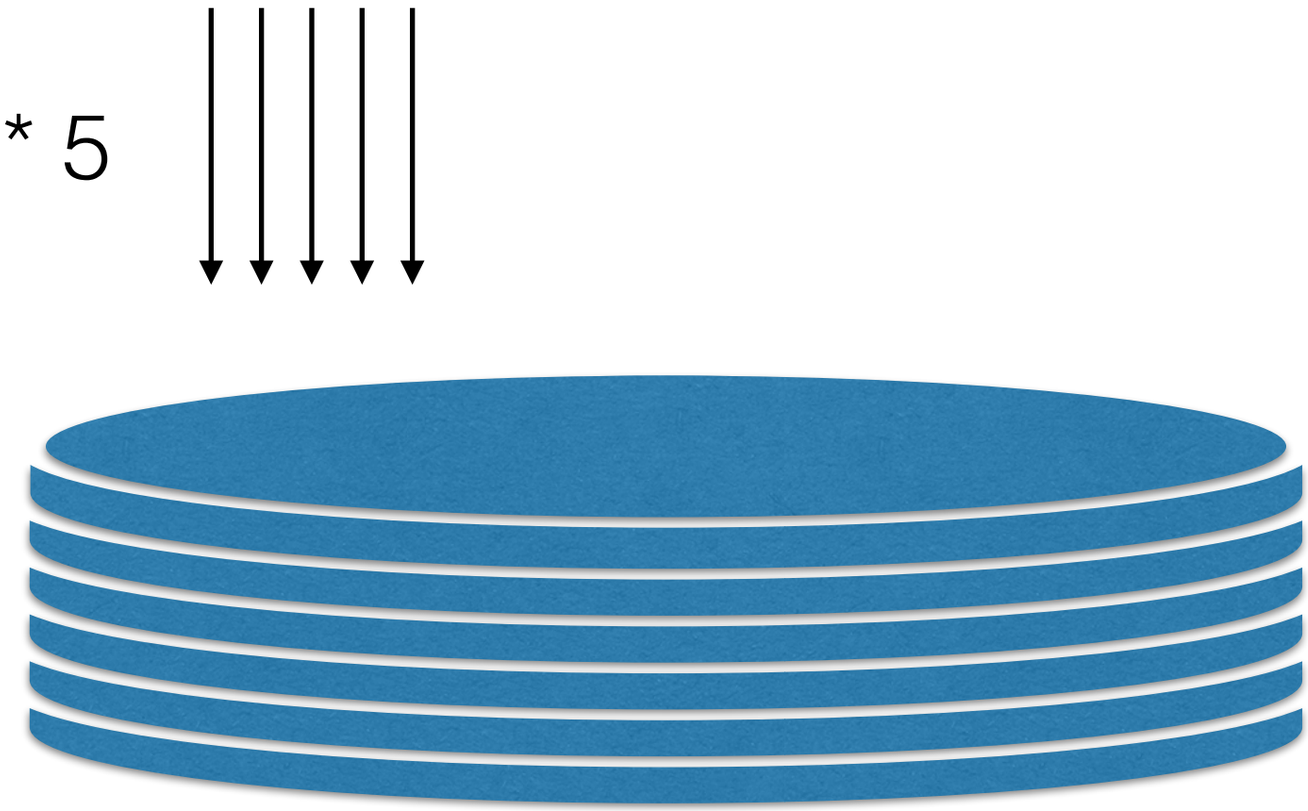




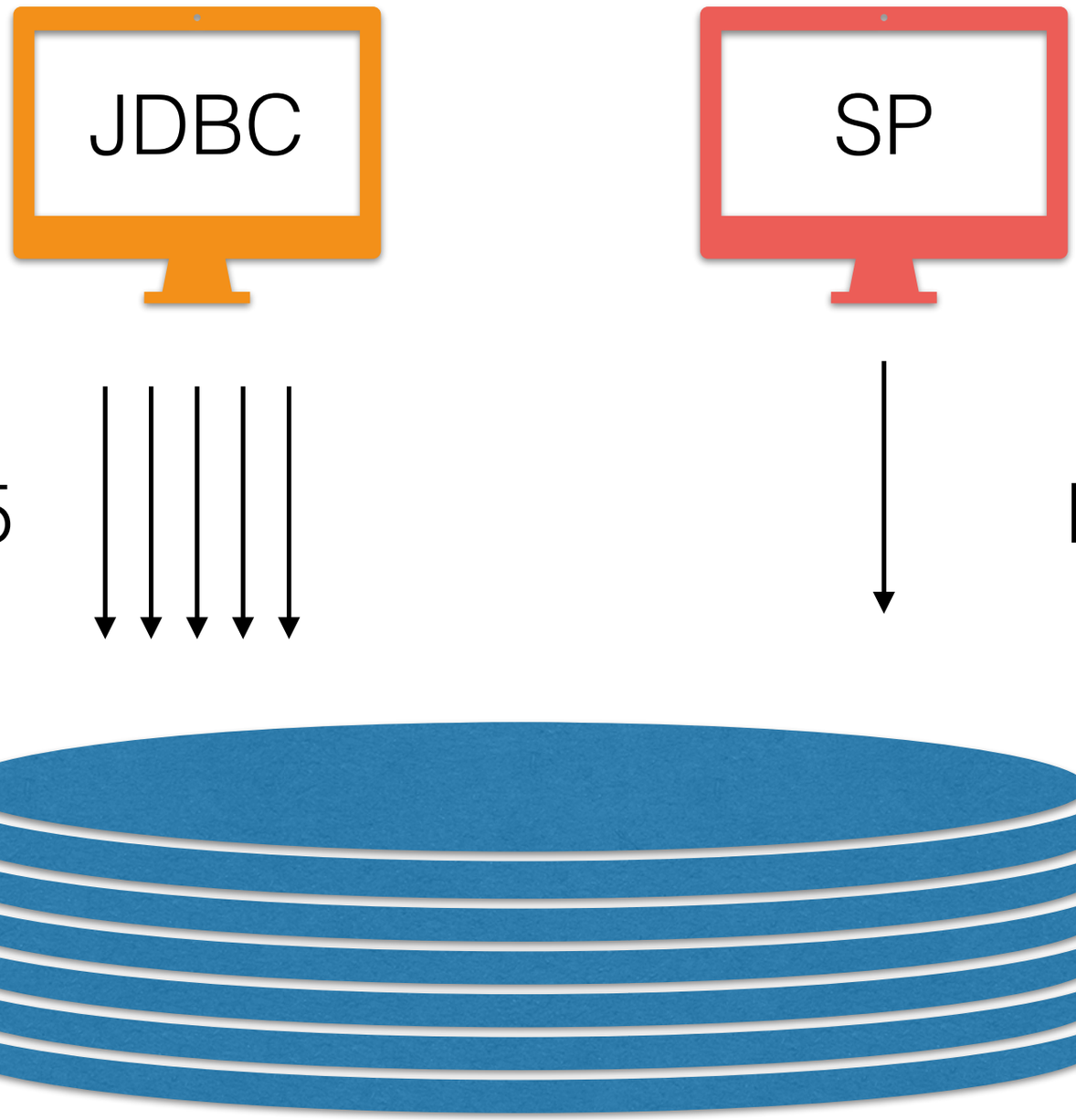
JDBC / SP ?



Query * 5



JDBC / SP ?



Executing a Transaction

- General steps
    - Generate parameters from TxParamGenerator
- Store Procedure
    - CallStoredProc() to start a stored procedure on the remote server, with the generated parameters
    - Remote server will return a SutResultSet when the procedure is finished
- JDBC
    - call Jdbc.execute() to start a Job on the local, with the generated parameters
    - Job will executing each sql via JDBC connector

How Server Process a StoredProc call ?

- When the server receive a remote procedure call, it will ask StoredProcFactory to generate the appropriate StoredProcedure
- The server will then call the StoredProcedure methods:
  - prepare(Obj…)
    - preparing the parameters
  - execute()
    - executing the transaction
    - this method will return the final result to the client

Outline

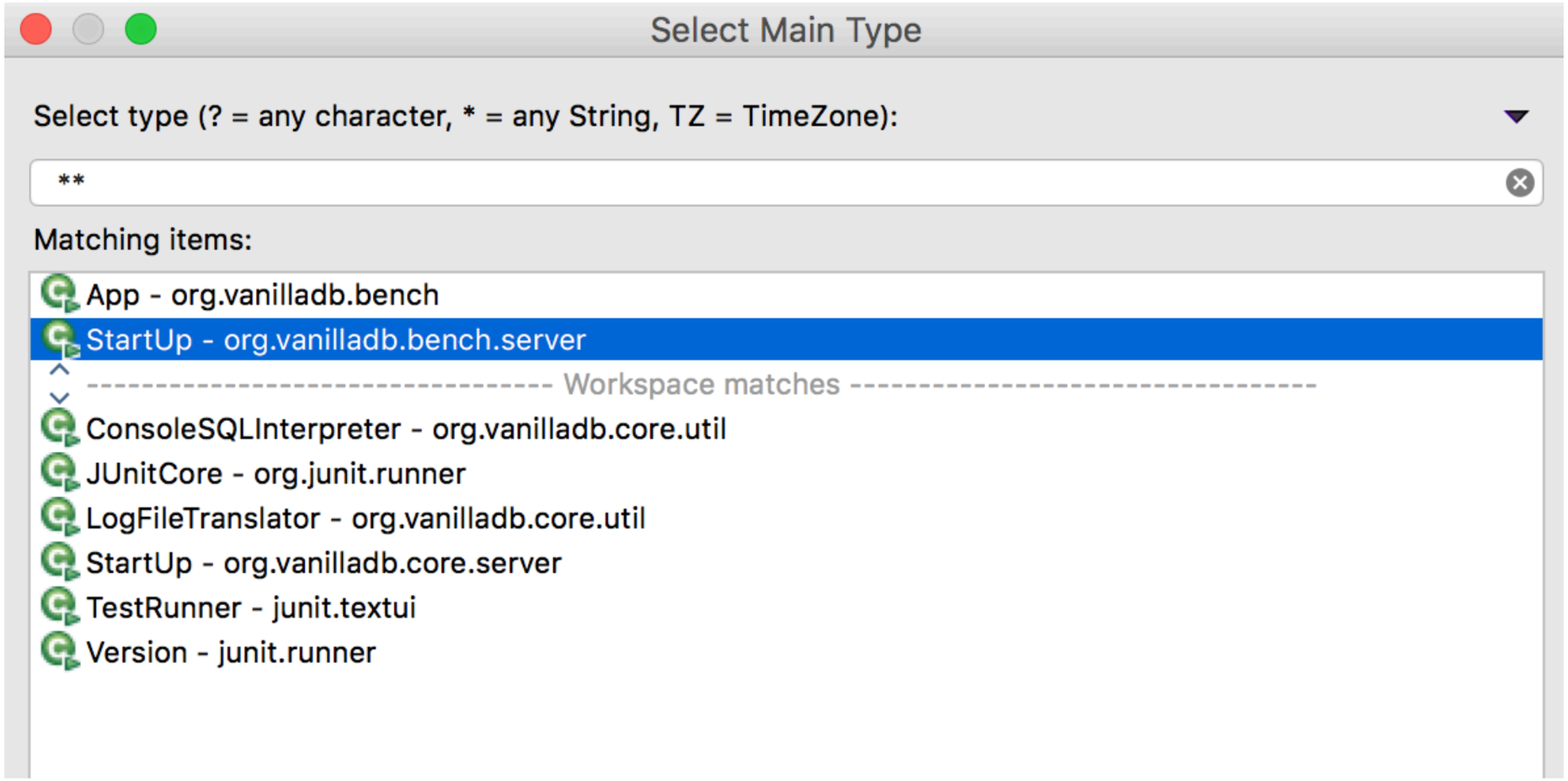
- Benchmark Project
  - Introducing Benchmark Project
  - Starting Up Server for Benchmarking
  - Setting Benchmark Configurations
  - Running Benchmark Client

Clone the Project First

- https://shwu10.cs.nthu.edu.tw/courses-databases-2018-spring/db18-assignment-2

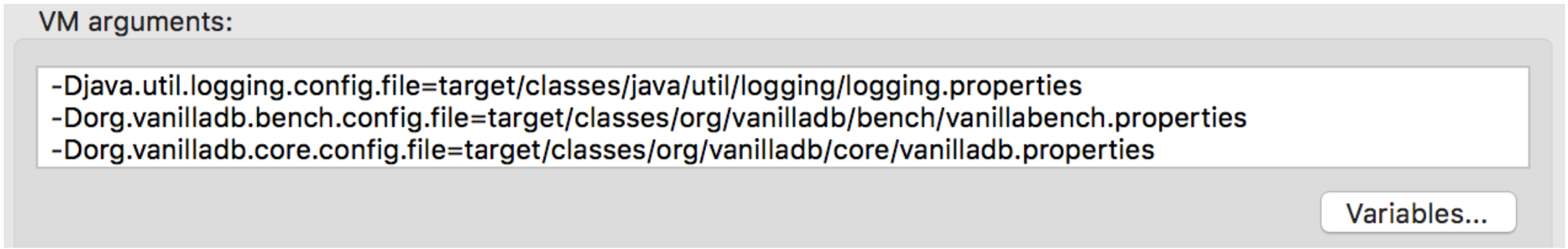
Starting Up Server (1/2)

- To benchmark a VanillaDB server, you need to start up the server in another entry point



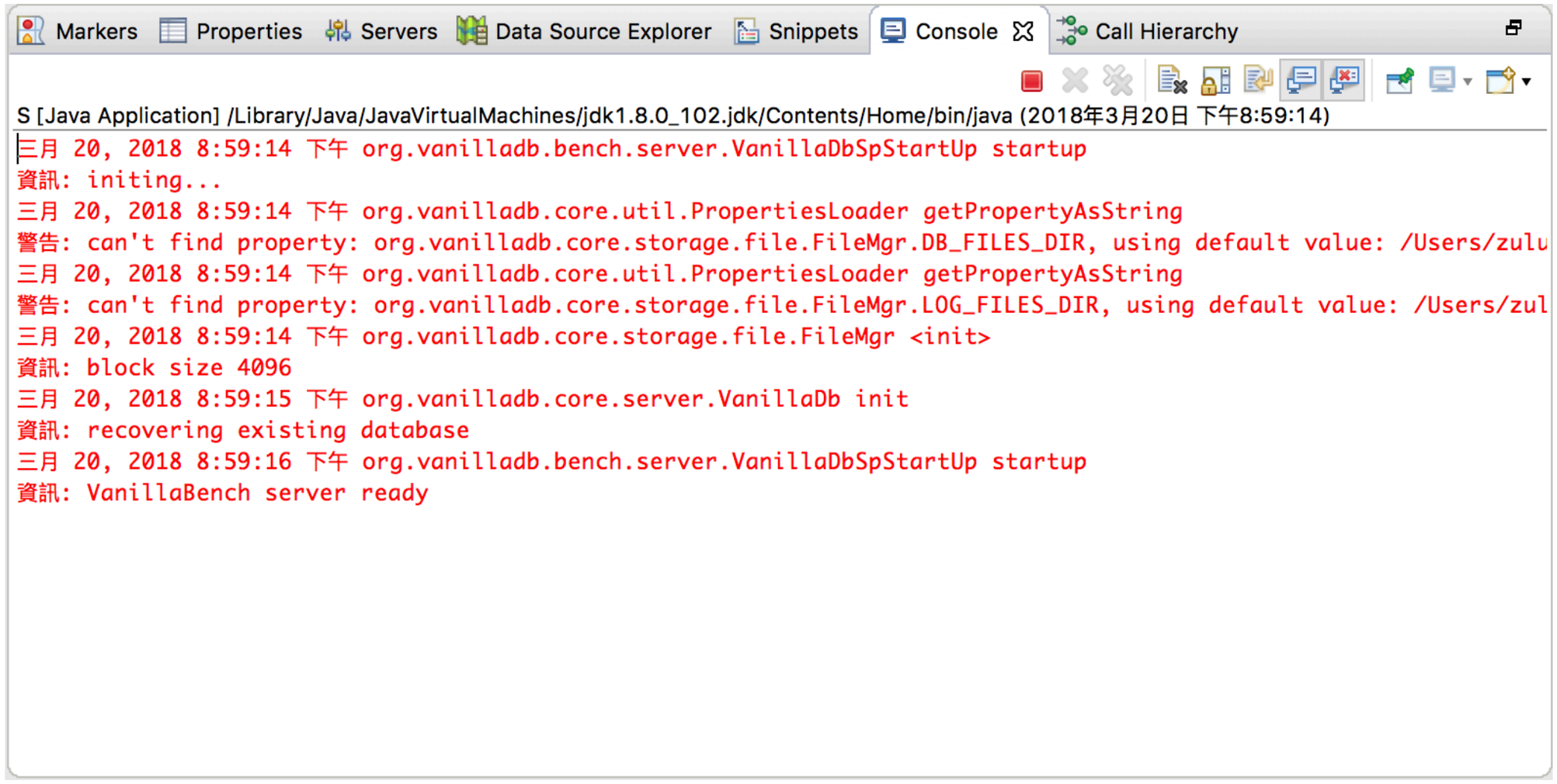
Starting Up Server (2/2)

- You also need to add one more VM argument for benchmarking.
- Don’t forget to add the Database Directory Name



You can copy those arguments from here

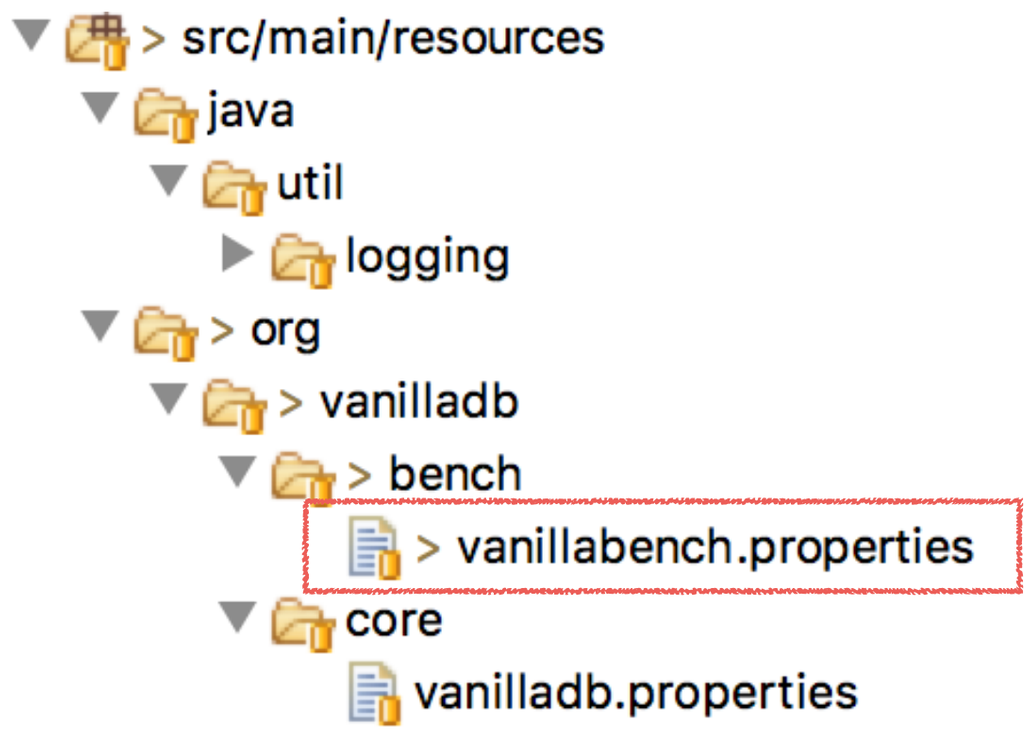
Server Messages



You should see similar messages if there is nothing wrong.

Setting Benchmark

- Benchmark project also has a properties file for configurations
  - vanillabench.properties



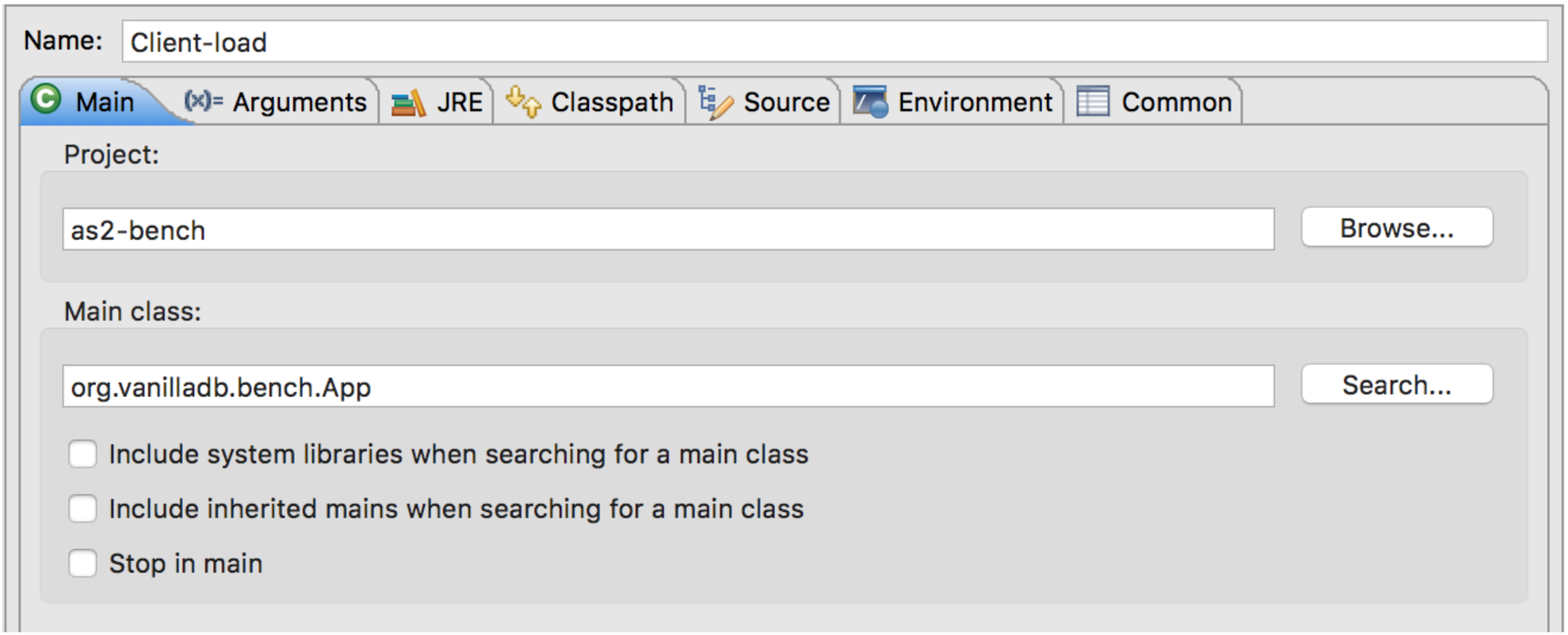
```
1
2 #
3 # Basic Parameters
4 #
5
6 # The running time for warming up before benchmarking
7 org.vanilladb.bench.BenchmarkerParameters.WARM_UP_INTERVAL=3000
8 # The running time for benchmarking
9 org.vanilladb.bench.BenchmarkerParameters.BENCHMARK_INTERVAL=6000
10 # The number of remote terminal executors for benchmarking
11 org.vanilladb.bench.BenchmarkerParameters.NUM_RTES=10
12 # The IP of the target database server
13 org.vanilladb.bench.BenchmarkerParameters.SERVER_IP=127.0.0.1
14 # 1 = JDBC, 2 = Stored Procedures
15 org.vanilladb.bench.BenchmarkerParameters.CONNECTION_MODE=2
16 # The path to the generated reports
17 org.vanilladb.bench.StatisticMgr.OUTPUT_DIR=
18 # Whether the RTEs display the results of each transaction
19 org.vanilladb.bench.rte.TransactionExecutor.DISPLAY_RESULT=false
20 # The number of items in the testing data set
21 org.vanilladb.bench.as2.As2Constants.NUM_ITEMS=100000
```

Important Parameters

- CONNECTION_MODE
  - It means whether it should use JDBC or stored procedures

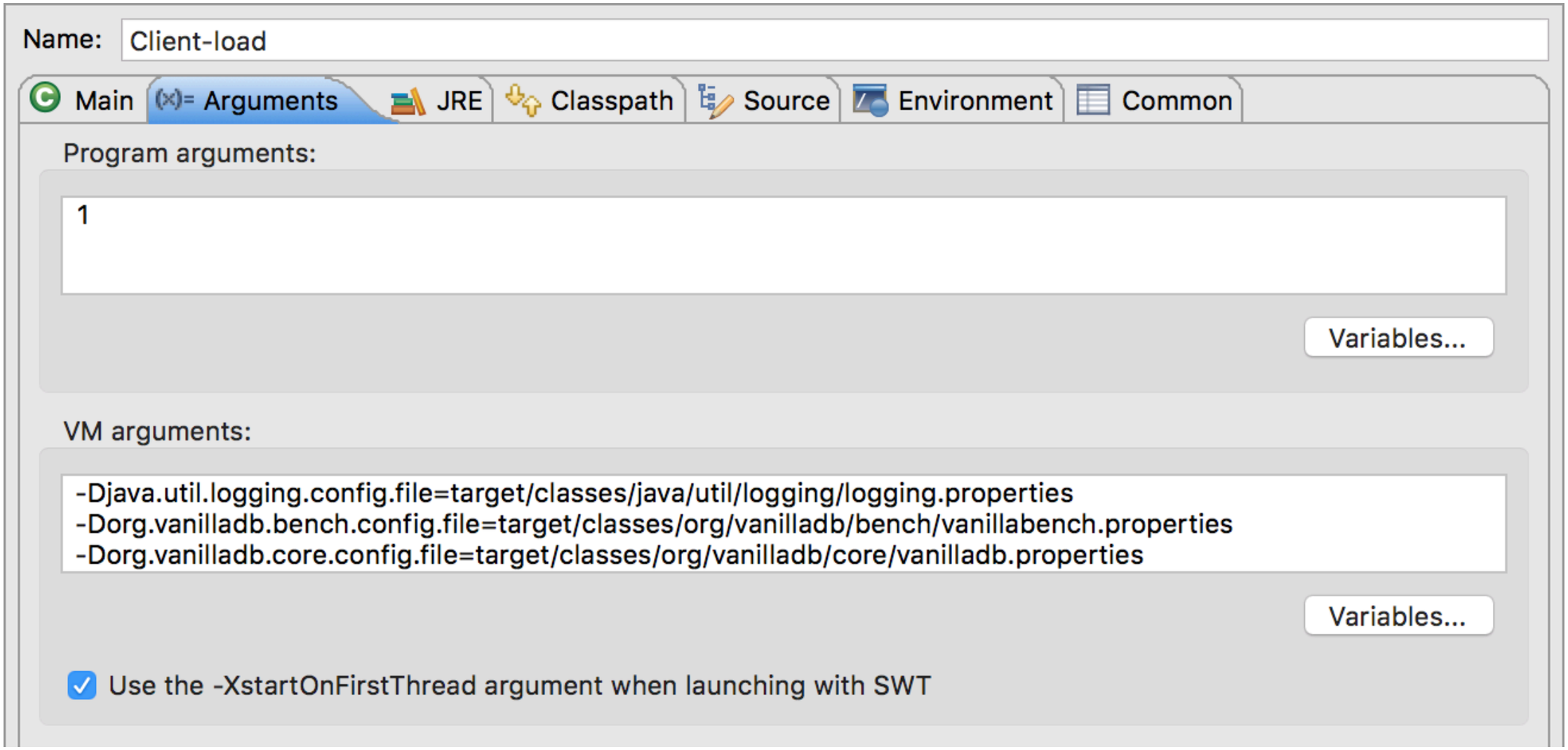
Running Client

- To run clients, create a run configuration for it



Arguments

- We also need to set some arguments
- Program Arguments
    - [Action]
    - 1 : Load Test-Bed
    - 2 : Lunch Benchmark
- VM Arguments
    - Use the same as the server



You can copy those arguments from here, then click ‘Apply’ and ‘Run’

Notes

- Before running testing, please **load testbed** first
- For now, loading testbed only works when you use stored procedures as interface
- Check if the output directory path that show in properties file exists
  - If not, create one or change the path

Run VanillaDB in Command Line

- Export client and server in .jar format
- Copy three property files into the folder where you export .jar file
- Type in the command as follow:

```
java { VM Arguments } -jar { server/client }.jar  { Program Arguments }
```

Q&A

- If you got any problem, you can check here first
- https://shwu10.cs.nthu.edu.tw/courses-databases-2018-spring/FAQ

- If your problem was very unique, just send a email let us know