

Introduction to ElaSQL and ElaSQLBench

Database Systems

DataLab, CS, NTHU

Spring, 2021

Recap

- Our final project is to improve **Hermes**.
- Hermes is in ElaSQL.

Outline

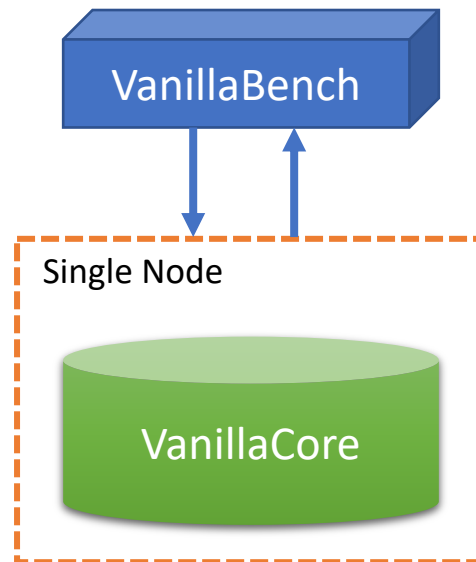
- What is ElaSQLBench & ElaSQL
- Let's Run a Benchmark on ElaSQL
 - Package code
 - Load Data
 - Workload configuration
 - Start Benchmarking
- Schedule: Next?

Outline

- What is ElaSQLBench & ElaSQL
- Let's Run a Benchmark on ElaSQL
 - Package code
 - Load Data
 - Workload configuration
 - Start Benchmarking
- Schedule: Next?

Recap

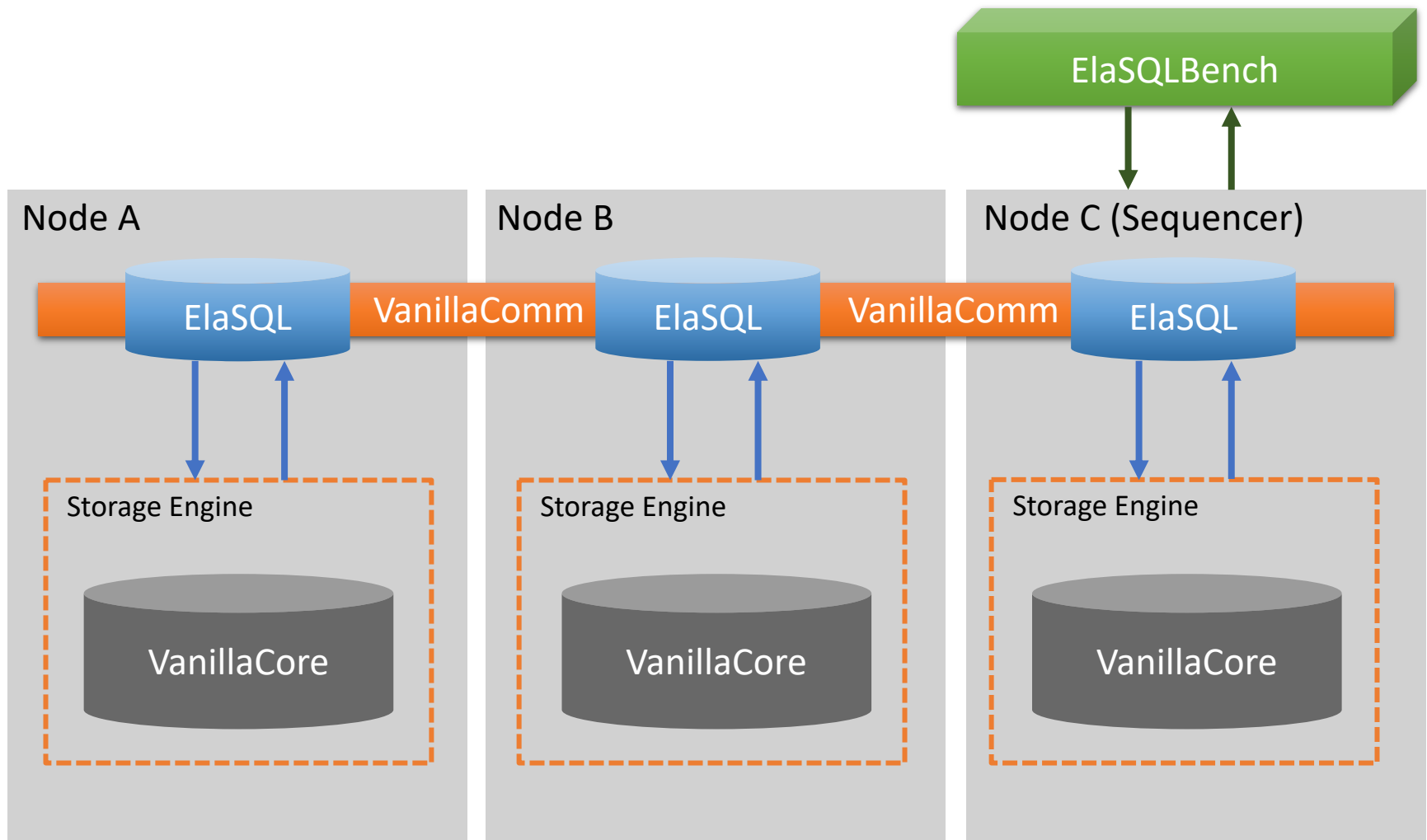
- VanillaCore is a **single-node** DBMS.
- VanillaBench is a benchmark framework aiming to test VanillaCore.



What is ElaSQL & ElaSQLBench

- **ElaSQL** is a **cross-node** DBMS that build on top of VanillaCore and VanillaComm.
- **ElaSQLBench** is a benchmark framework aiming to test ElaSQL.
- In the cross-node architecture
 - VanillaCore: as a storage engine (handle create, insert, delete, update from ElaSQL)
 - **VanillaComm**: for cross-node communication
 - VanillaBench: as a basis of ElaSQLBench (code reuse)

Cross-Node Architecture



Roles

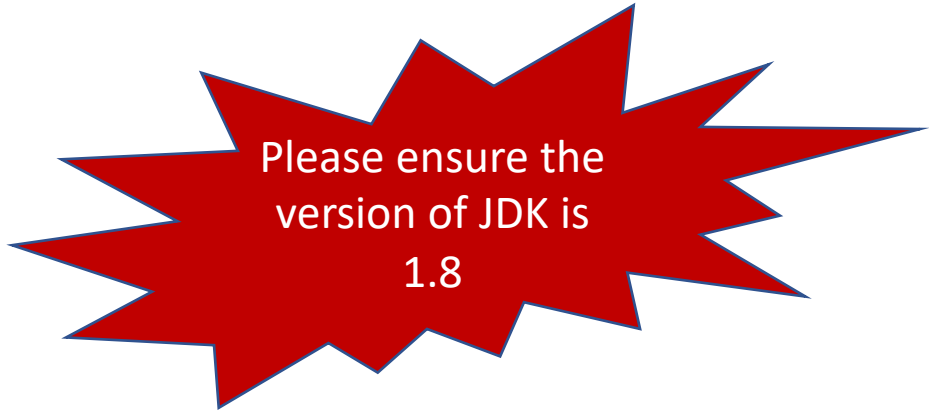
- There are three roles in a cluster
 - Client
 - Server
 - Sequencer
- In a cluster, we have several clients and servers.
- Among the servers, there must be a **SPECIAL** server called sequencer (kind of leader), which coordinates the total-order messages.

Outline

- What is ElaSQLBench & ElaSQL
- Recap: Target Workloads
- Let's Run a Benchmark on ElaSQL
 - Package code
 - Load Data
 - Workload configuration
 - Start Benchmarking
- Schedule: Next?

Package ElaSQLBench as JAR files

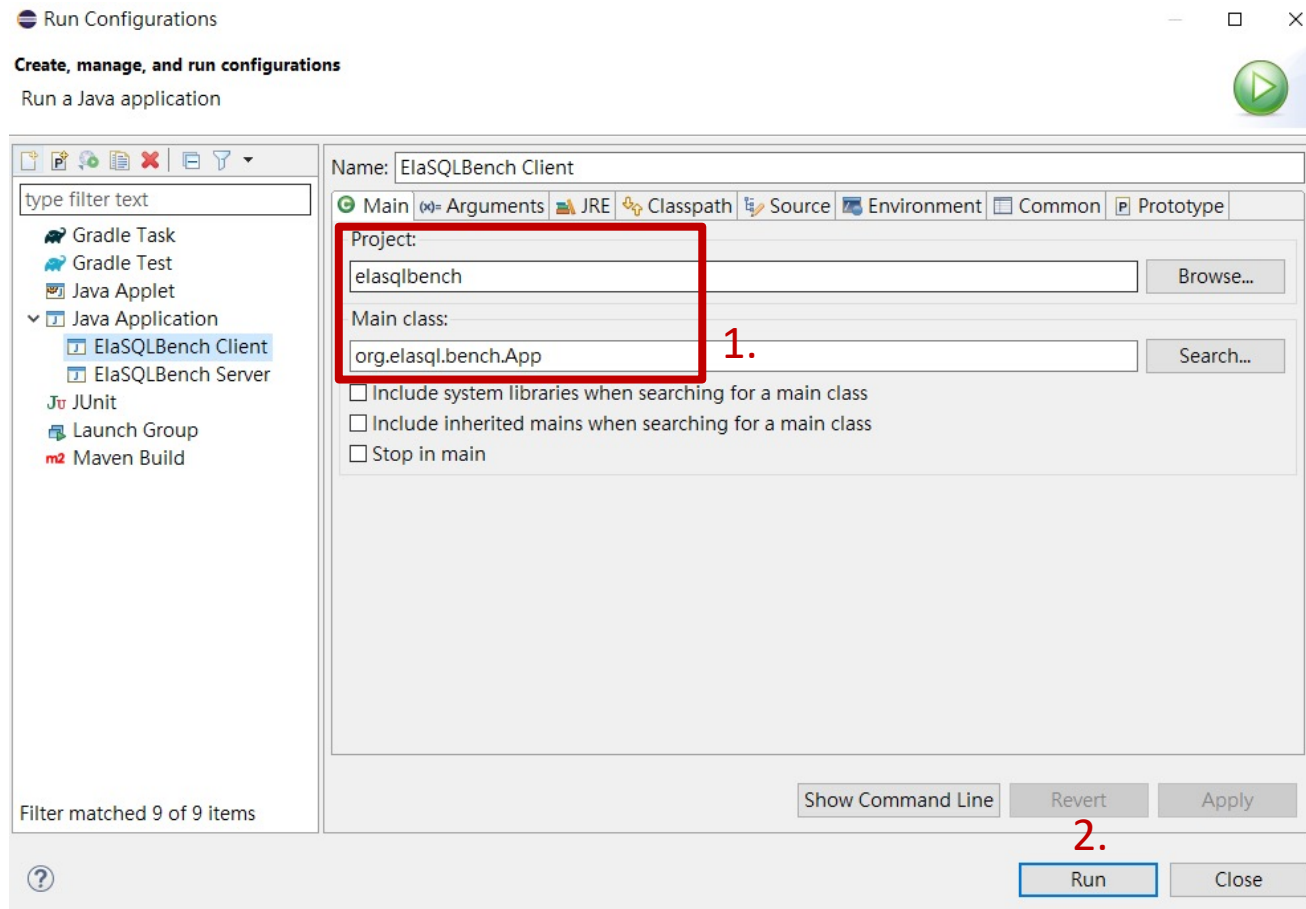
- Steps
 - 1. setup run configurations for JARs via Eclipse
 - 2. Export the project
- Example: package a client JAR
- Example: package a server JAR

A red starburst graphic with a blue outline, containing white text.

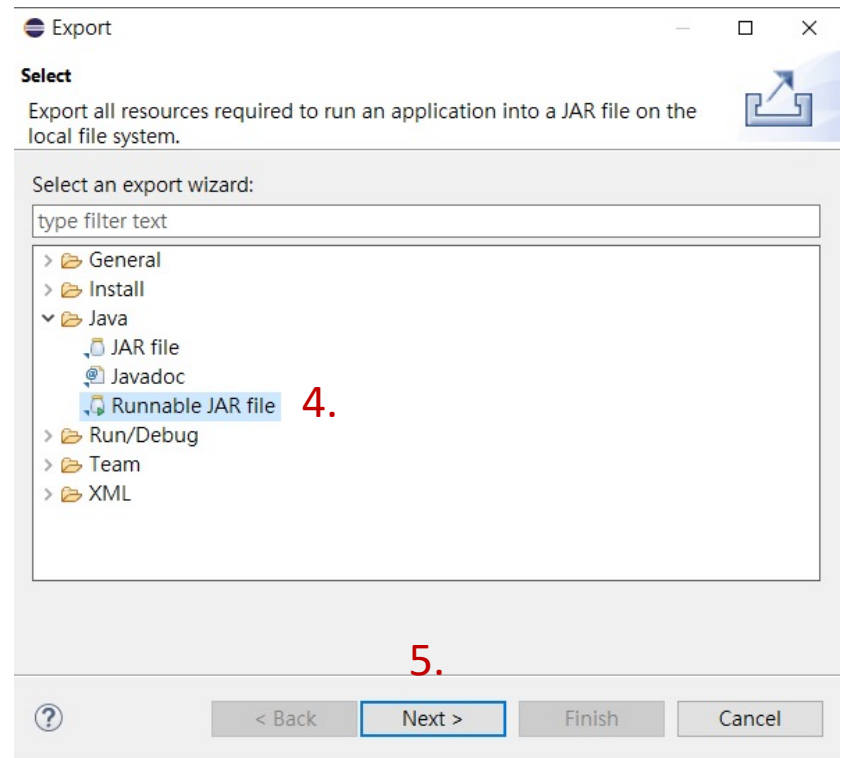
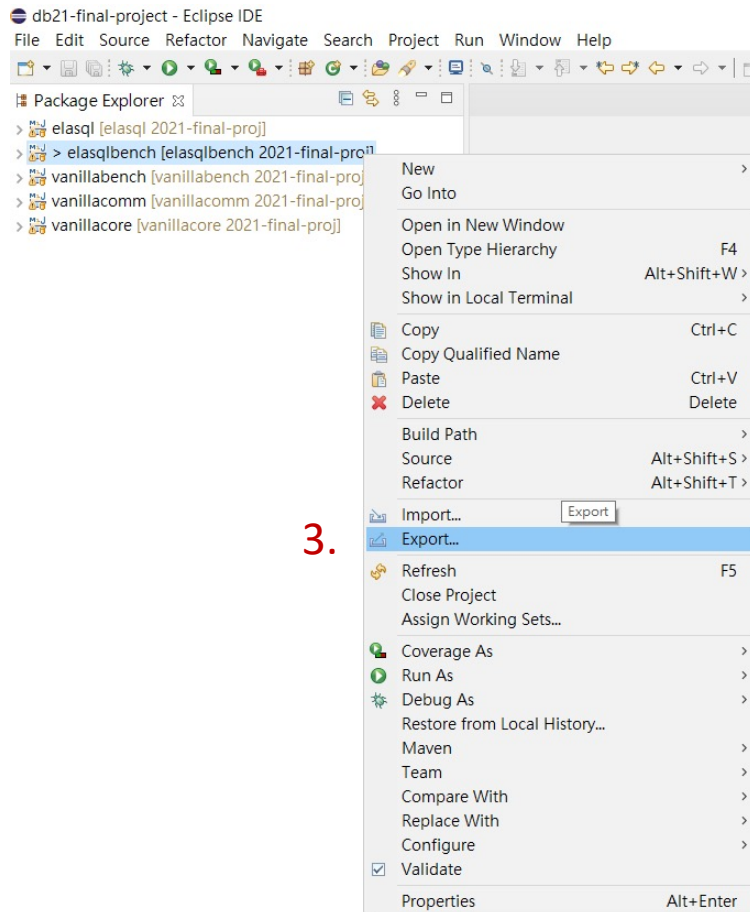
Please ensure the
version of JDK is
1.8

Example: Package a Client JAR

Client



Client



Client

Runnable JAR File Export

Runnable JAR File Specification

Select a 'Java Application' launch configuration to use to create a runnable JAR.

Launch configuration:

6.

Export destination:

7.

Library handling:

8. ☒ Extract required libraries into generated JAR
☐ Package required libraries into generated JAR
☐ Copy required libraries into a sub-folder next to the generated JAR

☐ Save as ANT script

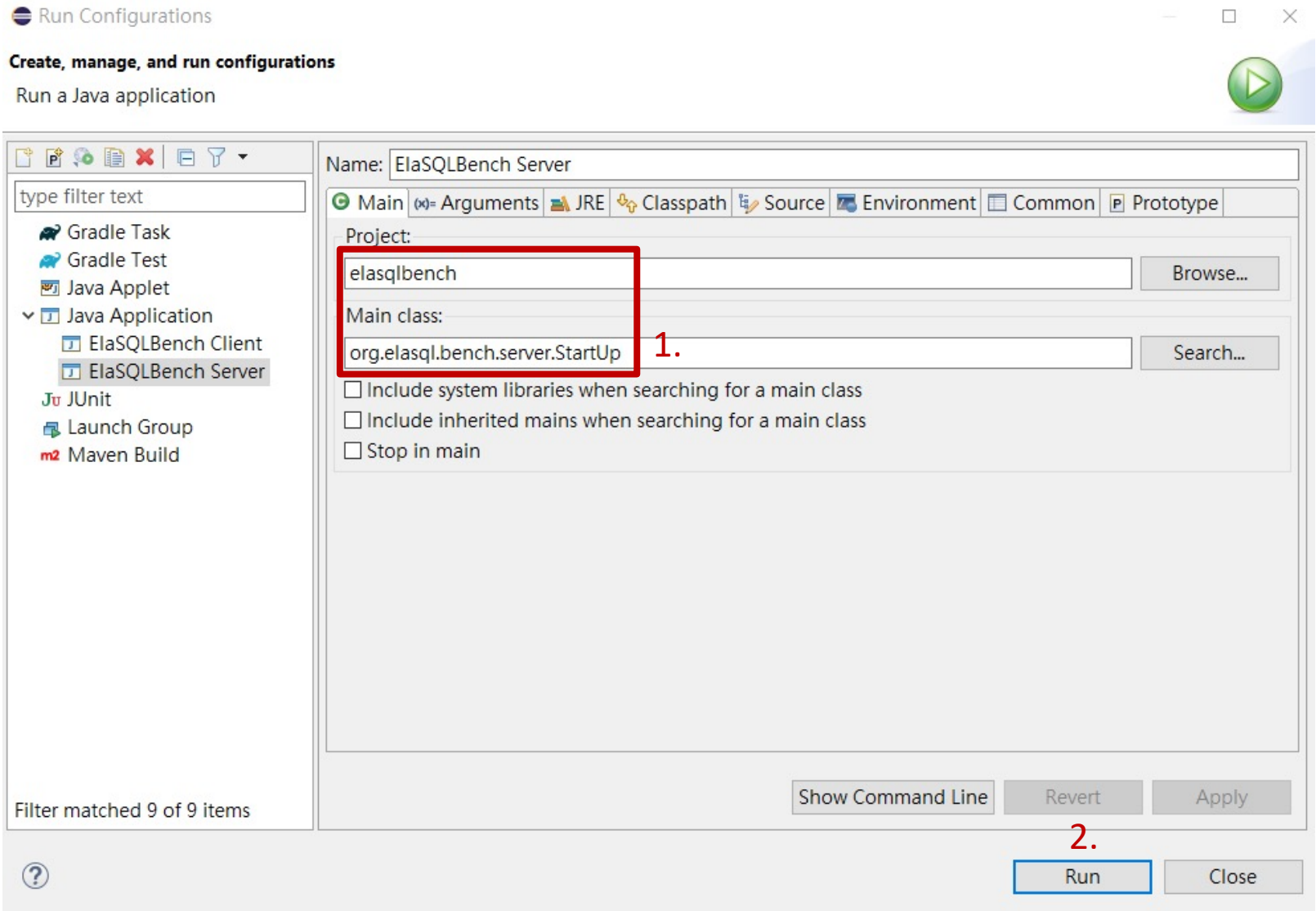
ANT script location:

9.

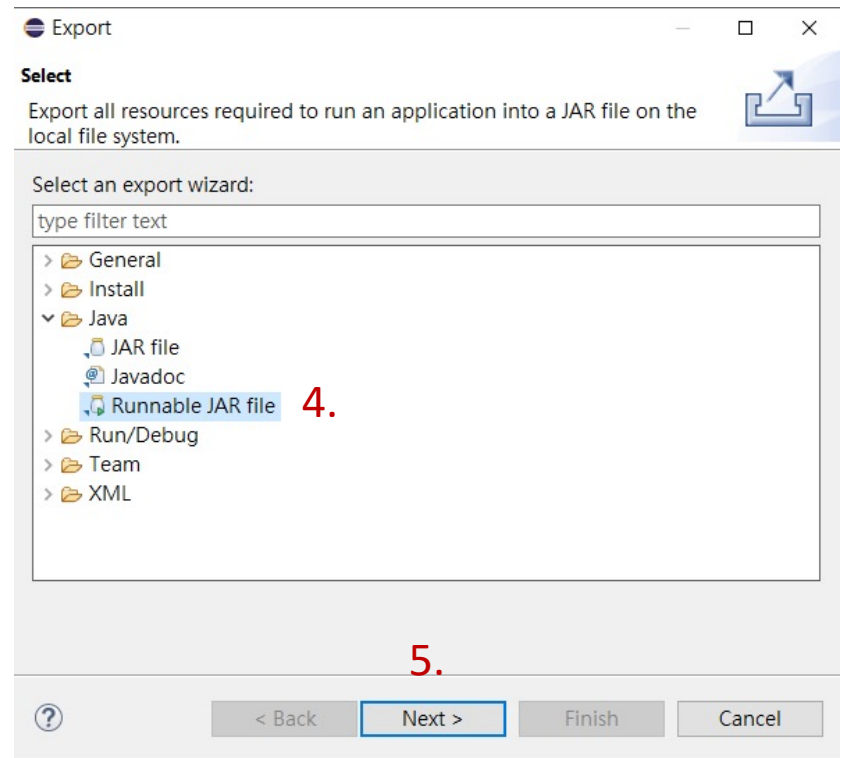
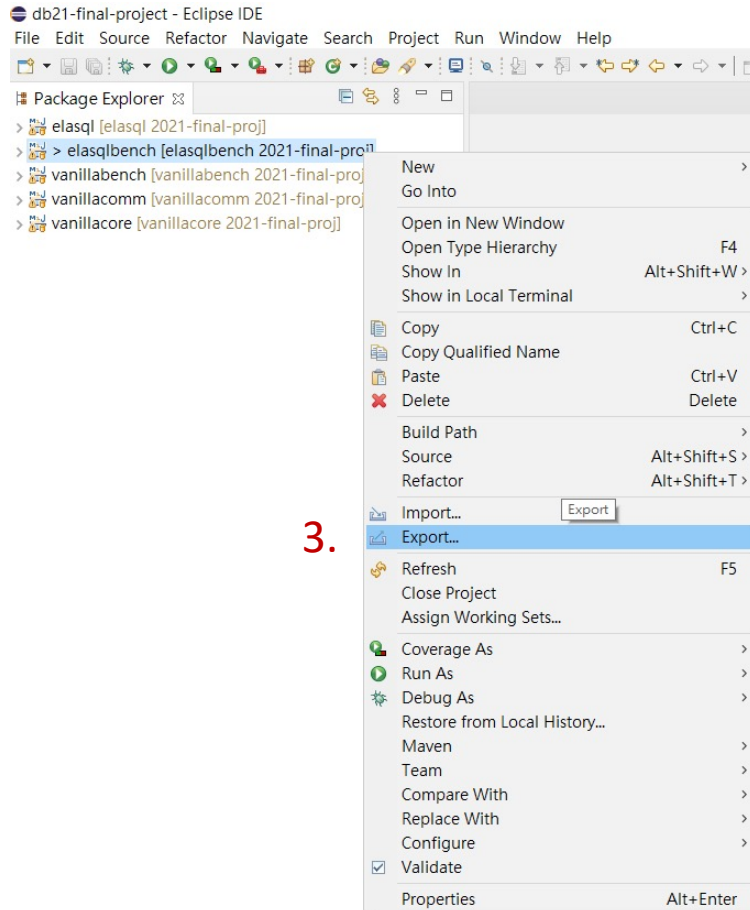
It's ok to ignore the warning message when exporting

Example: Package a Server JAR

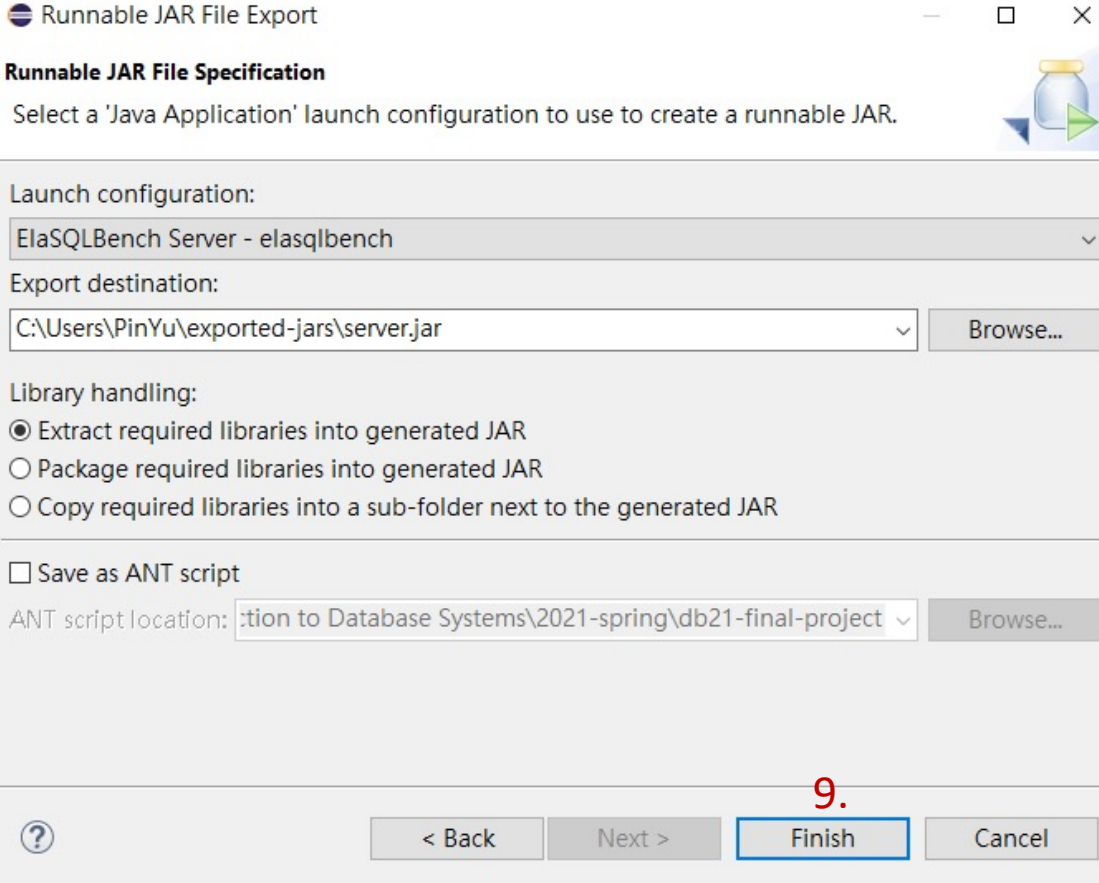
Server



Server



Server

The image shows a 'Runnable JAR File Export' dialog box from an IDE. It has a title bar with a standard icon and window controls. Below the title bar is a section titled 'Runnable JAR File Specification' with a sub-instruction: 'Select a 'Java Application' launch configuration to use to create a runnable JAR.' To the right of this text is a small icon of a jar with a green arrow. The main area of the dialog is divided into several sections. The first section is 'Launch configuration:' with a dropdown menu showing 'ElaSQLBench Server - elasqlbench'. The second section is 'Export destination:' with a text field containing 'C:\Users\PinYu\exported-jars\server.jar' and a 'Browse...' button. The third section is 'Library handling:' with three radio buttons: 'Extract required libraries into generated JAR' (which is selected), 'Package required libraries into generated JAR', and 'Copy required libraries into a sub-folder next to the generated JAR'. The fourth section is 'Save as ANT script' with an unchecked checkbox. Below this is 'ANT script location:' with a text field containing 'tion to Database Systems\2021-spring\db21-final-project' and a 'Browse...' button. At the bottom of the dialog are four buttons: a help button (question mark in a circle), '< Back', 'Next >', and 'Finish' (which is highlighted with a blue border). A red number '9.' is placed above the 'Finish' button.

Runnable JAR File Export

Runnable JAR File Specification

Select a 'Java Application' launch configuration to use to create a runnable JAR.

Launch configuration:

6. ElaSQLBench Server - elasqlbench

Export destination:

7. C:\Users\PinYu\exported-jars\server.jar Browse...

Library handling:

8. ☒ Extract required libraries into generated JAR
☐ Package required libraries into generated JAR
☐ Copy required libraries into a sub-folder next to the generated JAR

☐ Save as ANT script

ANT script location: tion to Database Systems\2021-spring\db21-final-project Browse...

9. Finish

< Back Next > Cancel

It's ok to ignore the warning message when exporting

Copy Properties Files

- Copy 6 properties files from ElasqlBench to the export destination

```
▼ MJD > elasqlbench [elasqlbench 2021-final-proj]
  > src/main/java
  ▼ src/main/resources
    ▼ java
      ▼ util
        ▼ logging
          logging.properties
    ▼ org
      ▼ elasql
        > elasql.properties
        > elasqlbench.properties
      ▼ vanilladb
        ▼ bench
          vanillabench.properties
        ▼ comm
          vanilladbcomm.properties
        ▼ core
          vanilladb.properties
```

Copy Google-workloads Csv

- Copy google-workloads-2min-3days.csv to the export destination

Workspace (D:) > Courses > Introduction to Database Systems > 2021-spring > db21-final-project				
名稱	修改日期	類型	大小	
.git	2021/5/31 上午 02...	檔案資料夾		
.metadata	2021/5/29 下午 02...	檔案資料夾		
elasql	2021/5/30 下午 11...	檔案資料夾		
elasqlbench	2021/5/30 下午 11...	檔案資料夾		
vanillabench	2021/5/30 下午 11...	檔案資料夾		
vanillacomm	2021/5/30 下午 11...	檔案資料夾		
vanillacore	2021/5/30 下午 11...	檔案資料夾		
.gitignore	2021/5/24 下午 04...	GITIGNORE 檔...	1 KB	
google-workloads-2min-3days.csv	2021/5/31 上午 01...	CSV 檔案	439 KB	

Result

- There will be 2 jars, 6 properties files and 1 csv file in the export destination

OS (C:) > 使用者 > PinYu > exported-jars			
名稱	修改日期	類型	大小
 client.jar	2021/5/31 上午 01...	Executable Jar...	3,224 KB
 elasql.properties	2021/5/30 下午 08...	PROPERTIES ...	5 KB
 elasqlbench.properties	2021/5/30 下午 03...	PROPERTIES ...	5 KB
 google-workloads-2min-3days.csv	2021/5/31 上午 01...	CSV 檔案	439 KB
 logging.properties	2021/5/29 下午 01...	PROPERTIES ...	3 KB
 server.jar	2021/5/31 上午 01...	Executable Jar...	3,224 KB
 vanillabench.properties	2021/5/29 下午 01...	PROPERTIES ...	5 KB
 vanilladb.properties	2021/5/29 下午 03...	PROPERTIES ...	7 KB
 vanilladbcomm.properties	2021/5/29 下午 01...	PROPERTIES ...	2 KB

Outline

- What is ElaSQLBench & ElaSQL
- Recap: Target Workloads
- Let's Run a Benchmark on ElaSQL
 - Package code
 - Load Data
 - Workload configuration
 - Start Benchmarking
- Schedule: Next?

Load Data

- For simplicity, we demonstrate it on a single machine
- 3 server processes (2 normal server + 1 sequencer)
- 1 client process

Load Data

- Modify the properties

Files	Properties
vanilladb.properties	org.vanilladb.core.storage.buffer.BufferMgr.BUFFER_POOL_SIZE=128000 # set it true if your OS is Linux org.vanilladb.core.storage.file.io.ioAllocator.USE_O_DIRECT=true
vanillabench.properties	org.vanilladb.bench.BenchmarkParameters.BENCH_TYPE=4
elasql.properties	org.elasql.server.Elasql.SERVICE_TYPE=1 org.elasql.remote.groupcomm.client.BatchSpcSender.BATCH_SIZE=1
elasqlbench.properties	org.elasql.bench.benchmarks.ycsb.ElasqlYcsbConstants.DATABASE_MODE=1 org.elasql.bench.benchmarks.ycsb.ElasqlYcsbConstants.INIT_RECORD_PER_PART=1000000
vanillacomm.properties	org.vanilladb.comm.view.ProcessView.SERVER_VIEW=0 127.0.0.1 42961, 1 127.0.0.1 42962, 2 127.0.0.1 42963 org.vanilladb.comm.view.ProcessView.CLIENT_VIEW=0 127.0.0.1 30000

3 servers including sequencers is a minimum requirement

1 client is enough when loading the data

You could increase YCSB table size by increasing INIT_RECORD_PER_PART

Run Commands to Start Servers/Clients

- A machine is represented by “ID IP PORT”
- Because we run these processes on a single machine, IP is 127.0.0.1(localhost)

```
# The views of the machine
# A machine is represented by "ID IP PORT"
# Each machine is split by a comma (,)
org.vanilladb.comm.view.ProcessView.SERVER_VIEW=0 127.0.0.1 42961, 1 127.0.0.1 42962, 2 127.0.0.1 42963
org.vanilladb.comm.view.ProcessView.CLIENT_VIEW=0 127.0.0.1 30000
```

Run Commands to Start Servers

- Copy the script below and save it as server.sh
 - Argument \$1: Database Name
 - Argument \$2: ID
 - Argument \$3: isSequencer (0 -> not sequencer, 1 -> is sequencer)

```
java \  
-Dorg.elasql.config.file=elasql.properties \  
-Dorg.elasql.bench.config.file=elasqlbench.properties \  
-Dorg.vanilladb.comm.config.file=vanilladbcomm.properties \  
-Dorg.vanilladb.bench.config.file=vanillabench.properties \  
-Dorg.vanilladb.core.config.file=vanilladb.properties \  
-Djava.util.logging.config.file=logging.properties \  
-jar server.jar \  
$1 \  
$2 \  
$3 \
```

Run Commands to Start Servers

- Open a Gitbash and run
 - Start server0



```
bash server.sh db0 0 0
```

- Start server1

```
bash server.sh db1 1 0
```

- Start sequencer (Actually, sequencer is for communication only and it won't create any database file)

```
bash server.sh dbseq 2 1
```



Now we have
opened three
servers

```
bash server.sh [db-name] [ID] [isSequencer]
```

Run Commands to Start Servers

- Check if “ElaSQL server ready” is printed in the console of sequencer

```
PinYu@DESKTOP-Q36P8AM MINGW64 ~/exported-jars
$ bash server.sh db3 2 1
五月 31, 2021 1:40:47 上午 org.vanilladb.core.util.PropertiesLoader getPropertyAsLong
警告: can't find property: org.vanilladb.bench.BenchmarkParameters.RTE_SLEEP_TIME, using default value: 0
五月 31, 2021 1:40:47 上午 org.vanilladb.core.util.PropertiesLoader getPropertyAsString
警告: can't find property: org.vanilladb.bench.BenchmarkParameters.SERVER_IP, using default value: 127.0.0.1
五月 31, 2021 1:40:47 上午 org.vanilladb.bench.BenchmarkParameters <clinit>
資訊: Using YCSB benchmarks
五月 31, 2021 1:40:47 上午 org.elasql.bench.server.ElasqlStartup startup
資訊: initializing benchmarker server...
五月 31, 2021 1:40:47 上午 org.elasql.bench.server.ElasqlStartup getTPartSpFactory
資訊: using YCSB stored procedures for T-Part
五月 31, 2021 1:40:47 上午 org.elasql.migration.MigrationComponentFactory <init>
資訊: using MGCRA as migration algorithm.
五月 31, 2021 1:40:47 上午 org.elasql.server.Elasql init
資訊: ElaSQL initializing...
五月 31, 2021 1:40:47 上午 org.elasql.server.Elasql init
資訊: using HERMES type service
五月 31, 2021 1:40:47 上午 org.elasql.server.Elasql init
資訊: initializing using Sequencer mode
五月 31, 2021 1:40:47 上午 org.elasql.remote.groupcomm.server.ConnectionMgr waitForServersReady
資訊: wait for all servers to start up comm. module
五月 31, 2021 1:40:47 上午 org.vanilladb.comm.server.VanillaCommServer run
資訊: Starts the network service
五月 31, 2021 1:40:47 上午 org.vanilladb.comm.protocols.totalorderappl.TotalOrderApplicationSession handleChannelInit
資訊: Socket registration request sent.
五月 31, 2021 1:40:47 上午 org.vanilladb.comm.protocols.totalorderappl.TotalOrderApplicationSession handleRegisterSocketEvent
資訊: Socket registration completed. (/127.0.0.1:42963)
五月 31, 2021 1:41:01 上午 org.vanilladb.comm.server.VanillaCommServer onAllProcessesReady
資訊: All processes are ready.
五月 31, 2021 1:41:01 上午 org.elasql.storage.metadata.PartitionMetaMgr <init>
資訊: Using 'Notification Partition Plan (underlayer: YCSB range partition (each range has 1000000 records))'
五月 31, 2021 1:41:01 上午 org.elasql.bench.server.ElasqlStartup startup
資訊: ElaSQL server ready
```

Run Commands to Start Clients

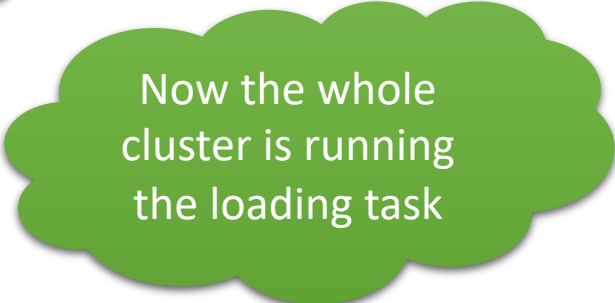
- Copy the script below and save it as client.sh
 - Argument \$1: ID
 - Argument \$2: BenchType (1: load, 2: benchmark)

```
java \  
-Dorg.elasql.config.file=elasql.properties \  
-Dorg.elasql.bench.config.file=elasqlbench.properties \  
-Dorg.vanilladb.comm.config.file=vanilladbcomm.properties \  
-Dorg.vanilladb.bench.config.file=vanillabench.properties \  
-Dorg.vanilladb.core.config.file=vanilladb.properties \  
-Djava.util.logging.config.file=logging.properties \  
-jar server.jar \  
$1 \  
$2 \
```

Run Commands to Start Servers

- After ElaSQL server ready, ppen a Gitbash and run
 - Start client0

```
bash cleint.sh 0 1
```



Now the whole
cluster is running
the loading task

```
bash client.sh [ID] [benchType]
```

The Scree Shots When Loading Successfully

- On the server console

```
五月 31, 2021 2:04:23 上午 org.elasql.bench.server.procedure.calvin.ycsb.YcsbTestbedLoaderProc generateRecords  
資訊: 950000 YCSB records has been populated.  
五月 31, 2021 2:04:31 上午 org.elasql.bench.server.procedure.calvin.ycsb.YcsbTestbedLoaderProc generateRecords  
資訊: 1000000 YCSB records has been populated.  
五月 31, 2021 2:04:31 上午 org.elasql.bench.server.procedure.calvin.ycsb.YcsbTestbedLoaderProc generateRecords  
資訊: Populating YCSB table completed.  
五月 31, 2021 2:04:31 上午 org.elasql.bench.server.procedure.calvin.ycsb.YcsbTestbedLoaderProc executeSql  
資訊: Loading completed. Flush all loading data to disks...  
五月 31, 2021 2:04:31 上午 org.vanilladb.core.storage.tx.recovery.CheckpointTask createCheckpoint  
資訊: Start creating checkpoint  
五月 31, 2021 2:04:51 上午 org.vanilladb.core.storage.tx.recovery.CheckpointTask createCheckpoint  
資訊: A checkpoint created  
五月 31, 2021 2:04:51 上午 org.elasql.bench.server.procedure.calvin.ycsb.YcsbTestbedLoaderProc executeSql  
資訊: Loading procedure finished. 1000000 YCSB records are loaded.
```

The Screenshots When Loading Successfully

- On the client console

```
PinYu@DESKTOP-Q36P8AM MINGW64 ~/exported-jars
$ bash client.sh 0 1
五月 31, 2021 2:02:15 上午 org.vanilladb.core.util.PropertiesLoader getPropertyA
sLong
警告: can't find property: org.vanilladb.bench.BenchmarkParameters.RTE_SLEEP_T
IME, using default value: 0
五月 31, 2021 2:02:16 上午 org.vanilladb.core.util.PropertiesLoader getPropertyA
sString
警告: can't find property: org.vanilladb.bench.BenchmarkParameters.SERVER_IP,
using default value: 127.0.0.1
五月 31, 2021 2:02:16 上午 org.vanilladb.bench.BenchmarkParameters <clinit>
資訊: Using YCSB benchmarks
五月 31, 2021 2:02:16 上午 org.vanilladb.comm.client.VanillaCommClient run
資訊: Starts the network service
五月 31, 2021 2:02:16 上午 org.vanilladb.comm.protocols.p2pappl.P2pApplicationSe
ssion handleChannelInit
資訊: Socket registration request sent.
五月 31, 2021 2:02:16 上午 org.elasql.remote.groupcomm.client.BatchSpcSender run
資訊: start batching-request worker thread (batch size = 1)
五月 31, 2021 2:02:16 上午 org.vanilladb.core.util.PropertiesLoader getPropertyA
sString
警告: can't find property: org.vanilladb.bench.StatisticMgr.OUTPUT_DIR, using de
fault value: null
五月 31, 2021 2:02:16 上午 org.elasql.bench.ElasqlBench loadTestbed
資訊: loading the testbed of the benchmark...
五月 31, 2021 2:02:16 上午 org.vanilladb.comm.protocols.p2pappl.P2pApplicationSe
ssion handleRegisterSocket
資訊: Socket registration completed. (/127.0.0.1:30000)
五月 31, 2021 2:04:51 上午 org.elasql.bench.ElasqlBench loadTestbed
資訊: loading procedure finished.
```


The Scree Shots When Loading Successfully

- In db1

> OS (C:) > 使用者 > PinYu > db1				
名稱	修改日期	類型	大小	
elastql.log	2021/5/31 上午 02...	文字文件	0 KB	
fldcat.tbl	2021/5/31 上午 02...	TBL 檔案	0 KB	
idx_ycsb_dir.idx	2021/5/31 上午 02...	IDX - Subtitle ...	0 KB	
idx_ycsb_leaf.idx	2021/5/31 上午 02...	IDX - Subtitle ...	0 KB	
idxcat.tbl	2021/5/31 上午 02...	TBL 檔案	0 KB	
idxkeycat.tbl	2021/5/31 上午 02...	TBL 檔案	0 KB	
tblcat.tbl	2021/5/31 上午 02...	TBL 檔案	0 KB	
vanilladb.log	2021/5/31 上午 02...	文字文件	0 KB	
viewcat.tbl	2021/5/31 上午 02...	TBL 檔案	0 KB	
ycsb.tbl	2021/5/31 上午 02...	TBL 檔案	1,333,340 KB	

The Scree Shots When Loading Successfully

- In db2

OS (C:) > 使用者 > PinYu > db2

名稱	修改日期	類型	大小
elasql.log	2021/5/31 上午 02...	文字文件	4 KB
fldcat.tbl	2021/5/31 上午 02...	TBL 檔案	8 KB
idx_ycsb_dir.idx	2021/5/31 上午 02...	IDX - Subtitle ...	0 KB
idx_ycsb_leaf.idx	2021/5/31 上午 02...	IDX - Subtitle ...	0 KB
idxcat.tbl	2021/5/31 上午 02...	TBL 檔案	4 KB
idxkeycat.tbl	2021/5/31 上午 02...	TBL 檔案	4 KB
tblcat.tbl	2021/5/31 上午 02...	TBL 檔案	8 KB
vanilladb.log	2021/5/31 上午 02...	文字文件	0 KB
viewcat.tbl	2021/5/31 上午 02...	TBL 檔案	4 KB
ycsb.tbl	2021/5/31 上午 02...	TBL 檔案	1,333,340 KB

Outline

- What is ElaSQLBench & ElaSQL
- Recap: Target Workloads
- **Let's Run a Benchmark on ElaSQL**
 - Package code
 - Load Data
 - **Workload configuration**
 - Start Benchmarking
- Schedule: Next?

Recap: Target Workloads

- We prepare three workloads for testing:
 - The Hotspot Workload
 - The Google Workload
 - The Hot Counter Workload
- For more details, please refer to the [introduction slides of final project](#).

Workload Configuration

- 3 configurations we need to handle.
 - Hotspot Workload
 - Google Workload
 - Hot Counter Workload

Workload Configuration – Hotspot Workload

Files	Properties
vanillabench.properties	org.vanilladb.bench.BenchmarkParameters.BENCH_TYPE=4 org.vanilladb.bench.BenchmarkParameters.BENCHMARK_INTERVAL=900000 org.vanilladb.bench.BenchmarkParameters.RTE_SLEEP_TIME= 0 org.vanilladb.bench.BenchmarkParameters.NUM_RTES=200 org.vanilladb.bench.StatisticMgr.GRANULARITY=10000
elasql.properties	org.elasql.server.Elasql.SERVICE_TYPE=3 org.elasql.remote.groupcomm.client.BatchSpcSender.BATCH_SIZE=20 # T-Part & Hermes org.elasql.schedule.tpart.TPartPartitioner.ROUTING_BATCH=200 # Hermes org.elasql.schedule.tpart.hermes.FusionTable.EXPECTED_MAX_SIZE=1000000 org.elasql.schedule.tpart.hermes.HermesNodeInserter.IMBALANCED_TOLERANCE=0.1
elasqlbench.properties	org.elasql.bench.benchmarks.ycsb.ElasqlYcsbConstants.DATABASE_MODE=1 org.elasql.bench.benchmarks.ycsb.ElasqlYcsbConstants.WORKLOAD_TYPE=3 org.elasql.bench.benchmarks.ycsb.ElasqlYcsbConstants.RW_TX_RATE=0.5 org.elasql.bench.benchmarks.ycsb.ElasqlYcsbConstants.TX_RECORD_COUNT=2 org.elasql.bench.benchmarks.ycsb.ElasqlYcsbConstants.ADD_INSERT_IN_WRITE_TX=0 org.elasql.bench.benchmarks.ycsb.ElasqlYcsbConstants.ZIPFIAN_PARAMETER=0.99 org.elasql.bench.benchmarks.ycsb.ElasqlYcsbConstants.TENANTS_PER_PART=4 org.elasql.bench.benchmarks.ycsb.ElasqlYcsbConstants.ENABLE_HOTSPOT=true org.elasql.bench.benchmarks.ycsb.ElasqlYcsbConstants.HOTSPOT_HOTNESS=0.9 org.elasql.bench.benchmarks.ycsb.ElasqlYcsbConstants.HOTSPOT_CHANGE_PERIOD= 600

Workload Configuration – Google Workload

After finishing the previous workload (hotspot), please modify these properties to fit the google workload

Files	Properties
elasqlbench.properties	<code>org.elasql.bench.benchmarks.ycsb.ElasqlYcsbConstants.WORKLOAD_TYPE=2</code> <code>org.elasql.bench.benchmarks.ycsb.ElasqlYcsbConstants.DIST_TX_RATE=0.5</code> <code>org.elasql.bench.benchmarks.ycsb.ElasqlYcsbConstants.USE_DYNAMIC_RECORD_COUNT=false</code> <code>org.elasql.bench.benchmarks.ycsb.ElasqlYcsbConstants.GOOGLE_TRACE_FILE=/path/google-workloads-2min-3days.csv</code> <code>org.elasql.bench.benchmarks.ycsb.ElasqlYcsbConstants.GOOGLE_TRACE_LENGTH=2160</code>

Workload Configuration – Hot Counter Workload

After finishing the previous workloads, please modify these properties to fit the hot counter workload

Files	Properties
elasqlbench.properties	<code>org.elasql.bench.benchmarks.ycsb.ElasqlYcsbConstants.WORKLOAD_TYPE=4</code> <code>org.elasql.bench.benchmarks.ycsb.ElasqlYcsbConstants.HOT_COUNT_PER_PART=1</code> <code>org.elasql.bench.benchmarks.ycsb.ElasqlYcsbConstants.HOT_UPDATE_RATE_IN_RW_TX=0.1</code>

Common Properties

- For the 3 workloads, they are all the same
 - `org.vanilladb.bench.BenchmarkParameters.BENCH_TYPE=4`
 - `org.elasql.server.Elasql.SERVICE_TYPE=3`

Particular Properties

- This properties determines which workloads we are going to run
 - `org.elasql.bench.benchmarks.ycsb.ElasqlYcsbConstants.WORKLOAD_TYPE`
- In Google Workload, we need to include a special file called **google-workloads-2min-3days.csv**
 - This file has kept the real cpu usage from Google's datacenter.
 - ElaSQL will reproduce the similar behaviors based on this file.

Some Properties You Can Modify

- It might work well to run 3 clients in one machine (vanillacomm.properties)
- The more powerful your computer is, the larger RTE numbers you can set (vanillabench.properties)
 - 200 might be a good number, please find a better number for your clusters
- BATCH_SIZE (elasql.properties)
 - 20 might be a good number
 - There are some restrictions on this properties, please see the next slide
- ROUTING_BATCH (elasql.properties)
 - 100~500 might be a good range, please find a better number for your clusters
 - There are some restrictions on this properties, please see the next slide

Restriction!

- *org.elasql.remote.groupcomm.client.BatchSpcSender.BATCH_SIZE* must be smaller than *org.vanilladb.bench.BenchmarkParameters.NUM_RTES*
- *org.elasql.schedule.tpart.TPartPartitioner.ROUTING_BATCH* must be smaller than (*org.vanilladb.bench.BenchmarkParameters.NUM_RTES* * Client Nums)
- Please do not modify these properties
 - `USE_DYNAMIC_RECORD_COUNT`
 - `ADD_INSERT_IN_WRITE_TX`
 - `ZIPFIAN_PARAMETER`
 - `GOOGLE_TRACE_LENGTH`
- You could modify the other properties excluding the above properties. But default value might be good enough.

Outline

- What is ElaSQLBench & ElaSQL
- Recap: Target Workloads
- Let's Run a Benchmark on ElaSQL
 - Package code
 - Load Data
 - Workload configuration
 - Start Benchmarking
- Schedule: Next?

Start Benchmarking – Google Workload

- Open a Gitbash and run
 - Start server0

```
bash server.sh db0 0 0
```

- Start server1

```
bash server.sh db1 1 0
```

- Start sequencer

```
bash server.sh dbseq 2 1
```

- After ElaSQL server readyt, Start client0

```
bash cleint.sh 0 2
```

Start Benchmarking – Google Workload

Client console

```
PinYu@DESKTOP-Q36P8AM MINGW64 ~/exported-jars
$ bash client.sh 0 2
五月 31, 2021 1:41:08 上午 org.vanilladb.core.util.PropertiesLoader getPropertyA
sLong
警告: can't find property: org.vanilladb.bench.BenchmarkParameters.RTE_SLEEP_T
IME, using default value: 0
五月 31, 2021 1:41:08 上午 org.vanilladb.core.util.PropertiesLoader getPropertyA
sString
警告: can't find property: org.vanilladb.bench.BenchmarkParameters.SERVER_IP,
using default value: 127.0.0.1
五月 31, 2021 1:41:08 上午 org.vanilladb.bench.BenchmarkParameters <clinit>
資訊: Using YCSB benchmarks
五月 31, 2021 1:41:08 上午 org.vanilladb.comm.client.VanillaCommClient run
資訊: Starts the network service
五月 31, 2021 1:41:08 上午 org.vanilladb.comm.protocols.p2pappl.P2pApplicationSe
ssion handleChannelInit
資訊: Socket registration request sent.
五月 31, 2021 1:41:08 上午 org.elasql.remote.groupcomm.client.BatchSpCsender run
資訊: start batching-request worker thread (batch size = 20)
五月 31, 2021 1:41:08 上午 org.vanilladb.core.util.PropertiesLoader getPropertyA
sString
警告: can't find property: org.vanilladb.bench.StatisticMgr.OUTPUT_DIR, using de
fault value: null
五月 31, 2021 1:41:08 上午 org.elasql.bench.ElasqlBench benchmark
資訊: checking the database on the server...
五月 31, 2021 1:41:08 上午 org.elasql.bench.ElasqlBench benchmark
資訊: database check passed.
五月 31, 2021 1:41:08 上午 org.elasql.bench.ElasqlBench benchmark
資訊: creating 200 emulators...
五月 31, 2021 1:41:08 上午 org.vanilladb.comm.protocols.p2pappl.P2pApplicationSe
ssion handleRegisterSocket
資訊: Socket registration completed. (/127.0.0.1:30000)
五月 31, 2021 1:41:09 上午 org.elasql.bench.benchmarks.ycsb.rte.SingleTableGoogl
eParamGen <clinit>
資訊: Use single-table Google YCSB generators (Read-write tx ratio: 0.500000, di
stributed tx ratio: 0.500000, 2 records/tx, 1 remote records/dist. tx, data size
: 20000000, google trace file: C:\Users\PinYu\exported-jars\google-workloads-2min
-3days.csv, google trace length: 2160)
五月 31, 2021 1:41:09 上午 org.elasql.bench.ElasqlBench benchmark
資訊: waiting for connections...
五月 31, 2021 1:41:10 上午 org.elasql.bench.ElasqlBench benchmark
資訊: start benchmarking.
五月 31, 2021 1:41:10 上午 org.elasql.bench.ElasqlBench benchmark
資訊: warm up period finished.
五月 31, 2021 1:41:10 上午 org.elasql.bench.ElasqlBench benchmark
資訊: start recording results...
```

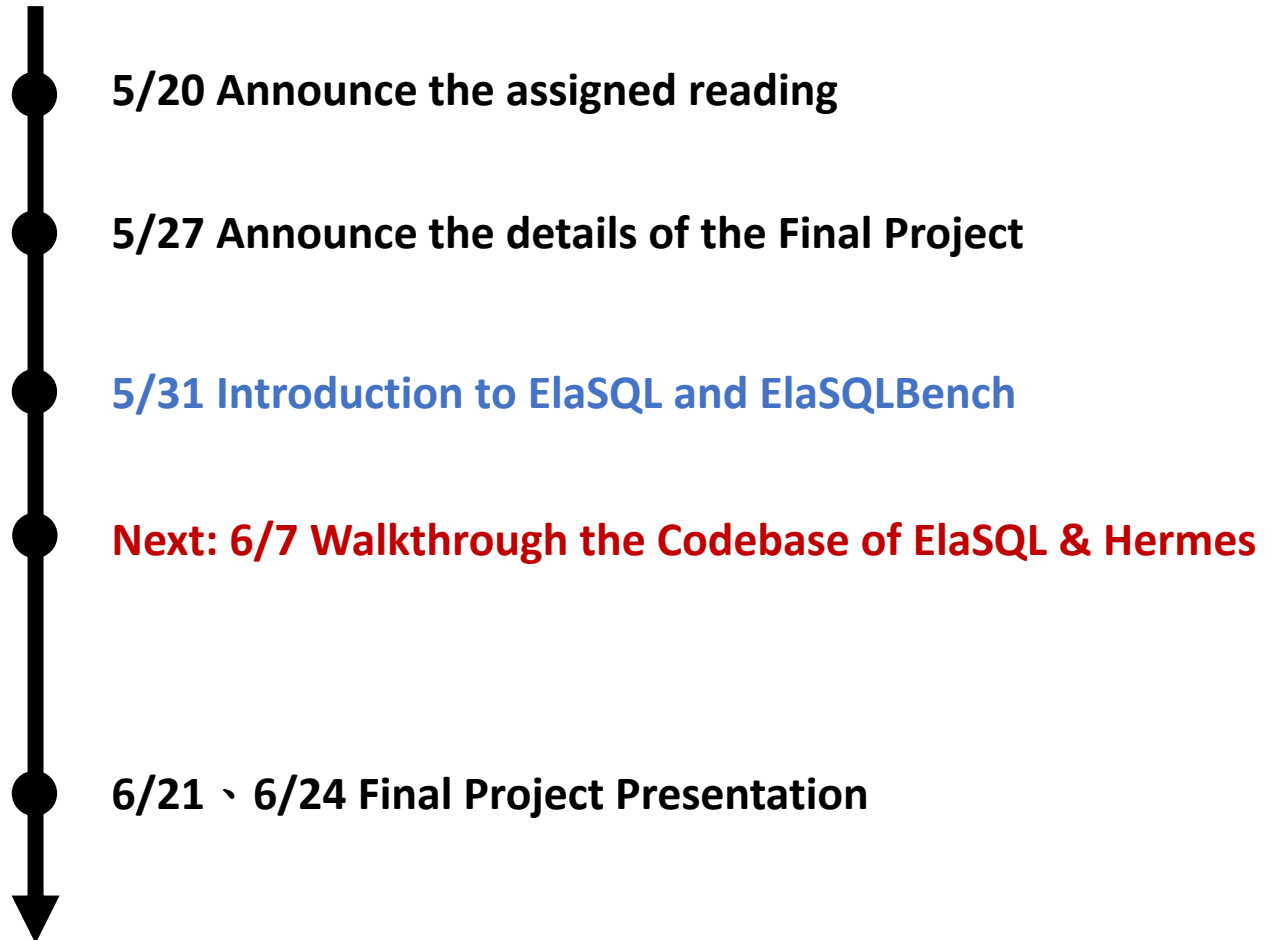
Client console

```
資訊: start benchmarking.
五月 31, 2021 1:41:10 上午 org.elasql.bench.ElasqlBench benchmark
資訊: warm up period finished.
五月 31, 2021 1:41:10 上午 org.elasql.bench.ElasqlBench benchmark
資訊: start recording results...
Not replaying. Current replay point: -86
Not replaying. Current replay point: -85
Not replaying. Current replay point: -80
Not replaying. Current replay point: -75
Not replaying. Current replay point: -70
Not replaying. Current replay point: -65
Not replaying. Current replay point: -60
Not replaying. Current replay point: -55
Not replaying. Current replay point: -50
Not replaying. Current replay point: -45
Not replaying. Current replay point: -40
Not replaying. Current replay point: -35
Not replaying. Current replay point: -30
Not replaying. Current replay point: -25
Not replaying. Current replay point: -20
Not replaying. Current replay point: -15
Not replaying. Current replay point: -10
Not replaying. Current replay point: -5
Replaying. Current replay point: 0
Replaying. Current replay point: 4
Replaying. Current replay point: 9
Replaying. Current replay point: 14
Replaying. Current replay point: 19
Replaying. Current replay point: 24
Replaying. Current replay point: 29
Replaying. Current replay point: 34
Replaying. Current replay point: 39
Replaying. Current replay point: 44
Replaying. Current replay point: 49
Replaying. Current replay point: 54
Replaying. Current replay point: 59
Replaying. Current replay point: 64
Replaying. Current replay point: 69
Replaying. Current replay point: 74
Replaying. Current replay point: 79
Replaying. Current replay point: 84
Replaying. Current replay point: 89
Replaying. Current replay point: 94
Replaying. Current replay point: 99
Replaying. Current replay point: 104
Replaying. Current replay point: 109
Replaying. Current replay point: 114
Replaying. Current replay point: 119
Replaying. Current replay point: 124
Replaying. Current replay point: 129
Replaying. Current replay point: 134
Replaying. Current replay point: 139
Replaying. Current replay point: 144
Replaying. Current replay point: 149
Replaying. Current replay point: 154
Replaying. Current replay point: 159
Replaying. Current replay point: 164
Replaying. Current replay point: 169
Replaying. Current replay point: 174
Replaying. Current replay point: 179
Replaying. Current replay point: 184
Replaying. Current replay point: 189
Replaying. Current replay point: 194
Replaying. Current replay point: 199
Replaying. Current replay point: 204
五月 31, 2021 1:46:10 上午 org.elasql.bench.ElasqlBench benchmark
資訊: benchmark preiod finished. Stopping RTEs...
五月 31, 2021 1:46:11 上午 org.vanilladb.bench.StatisticMgr outputReport
資訊: Finnish creating tpcc benchmark report
五月 31, 2021 1:46:11 上午 org.elasql.bench.ElasqlBench benchmark
資訊: benchmark process finished.
```


Outline

- What is ElaSQLBench & ElaSQL
- Recap: Target Workloads
- Let's Run a Benchmark on ElaSQL
 - Package code
 - Load Data
 - Workload configuration
 - Start Benchmarking
- **Schedule: Next?**

Timeline





Good Luck !