

Assignment 4 & Benchmarks

Introduction to Database Systems

DataLab

CS, NTHU

Outline

- Assignment 4
- Benchmarks
 - The Micro-benchmark
 - The TPC-C Benchmark
- Guidelines for Experiments
- Example Results
- The Workflow of Benchmarking

Outline

- Assignment 4
- Benchmarks
 - The Micro-benchmark
 - The TPC-C Benchmark
- Guidelines for Experiments
- Example Results
- The Workflow of Benchmarking

Outline

- Assignment 4
- **Benchmarks**
 - The Micro-benchmark
 - The TPC-C Benchmark
- Guidelines for Experiments
- Example Results
- The Workflow of Benchmarking

The Micro-Benchmark

- Two types of transactions.
 - **Read-only** transaction => reads 10 records.
 - **Read-write** transaction => reads and updates 10 records.
 - The ratio is controlled by **RW_TX_RATE**.
- The data set is split into two parts.



- 1 is chosen from hot set, 9 are chosen from cold set.
- The number of hot records is control by **HOT_CONFLICT_RATE**.

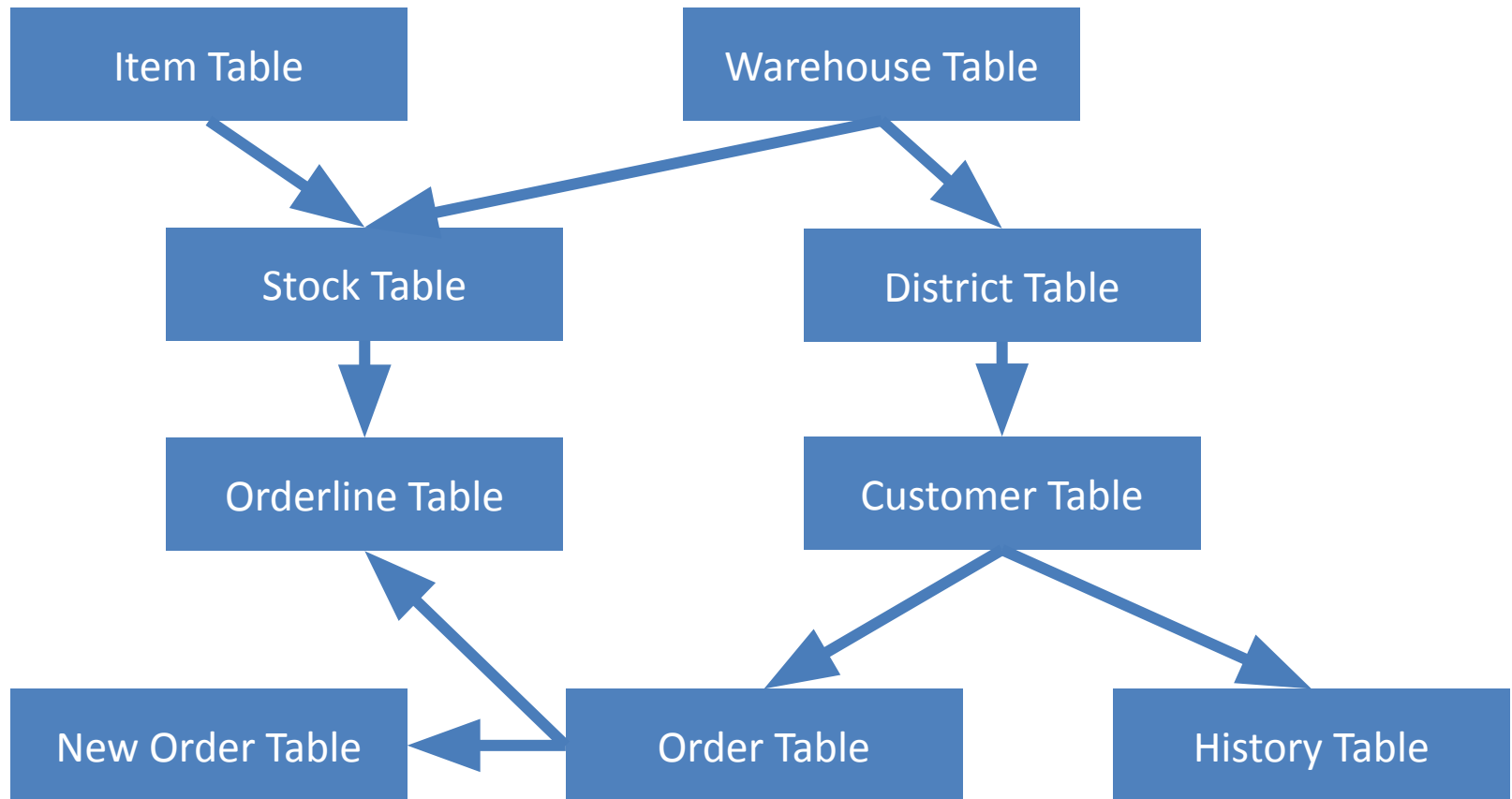
Outline

- Assignment 4
- **Benchmarks**
 - The Micro-benchmark
 - **The TPC-C Benchmark**
- Guidelines for Experiments
- Example Results
- The Workflow of Benchmarking

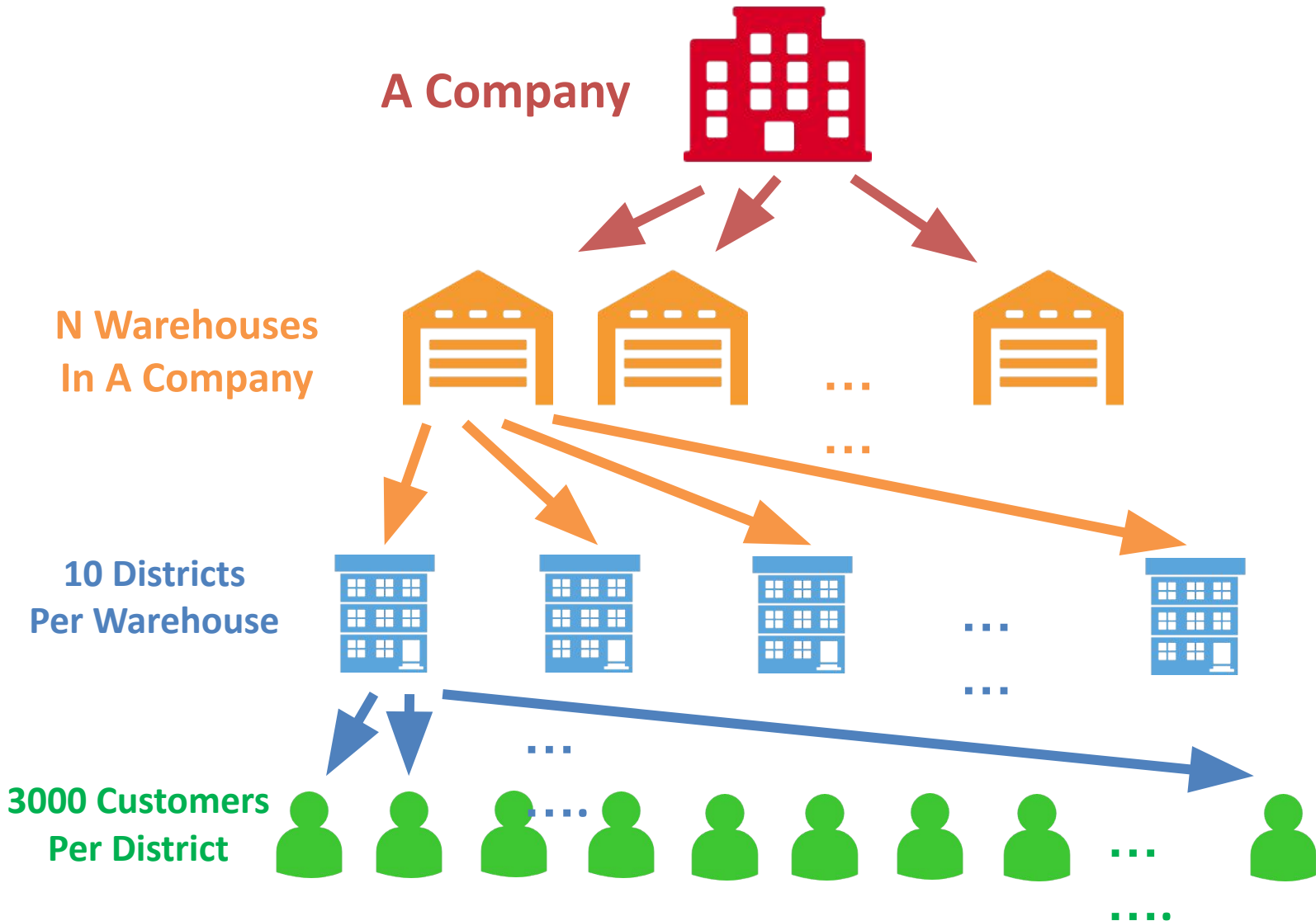
The TPC-C Benchmark

- The TPC-C benchmark is an industry-standard benchmark purposed by TPC (Transaction Processing Council).
 - There are also TPC-A, TPC-B, TPC-E, TPC-H.
- It simulates a warehouse management system.
 - Tree-structured: almost all records are related to a warehouse record.
 - Easy-to-partition: good for a distributed DBMS.

Database Architecture



Warehouses (Tree-Structured)



Orders

A Customer



A New Order



An Order



An Orderline

An Orderline

An Orderline

An Orderline

Types of Transactions

- New Order
 - 23 reads, 11 updates, 12 inserts in average.
- Payment
 - 4 reads, 3 updates, 1 insert.
- Stock Level
- Order Status
- Delivery

Outline

- Assignment 4
- Benchmarks
 - The Micro-benchmark
 - The TPC-C Benchmark
- **Guidelines for Experiments**
- Example Results
- The Workflow of Benchmarking

Guidelines for Experiments

- Think about what settings can highlight your improvement.
- Make sure there is no other CPU-intensive programs running on the testing machines.
- Put the server and the client on different machine if you can.
- Use stored procedures.
- Using a fresh database every time.
- Find best # of RTEs before real experiments.
 - Which give you highest throughput.
- Throughput is a more important indicator for concurrency than latency.
- Draw you results as line plots or histograms in the report.

Outline

- Assignment 4
- Benchmarks
 - The Micro-benchmark
 - The TPC-C Benchmark
- Guidelines for Experiments
- **Example Results**
- The Workflow of Benchmarking

Example Results for the Micro-benchmarks

- Settings
 - RTE = 10
 - RW Tx Rate = 0.5
 - Conflict Rate = 0.001
- Throughputs (txs/min)

Buffer Size	Basic Version	Optimized Version	Speed Up
100000	111558	174521	56%
100	39285	75164	91%

Outline

- Assignment 4
- Benchmarks
 - The Micro-benchmark
 - The TPC-C Benchmark
- Guidelines for Experiments
- Example Results
- **The Workflow of Benchmarking**

The Workflow of Benchmarking

1. Setup properties
2. Run **Start Benchmark Server**
3. Run **Load Testbed**
4. **Copy the Database (Ensure the test data is same)**
5. Run **Launch Benchmark**
6. Record the origin result
7. **Restore the Database**
8. Repeat step 2-5 again
9. Record the result after optimize