香港中文大學(深圳)
The Chinese University of Hong Kong, Shenzhen

SCHOOL OF
DATA SCIENCE
數據科學學院

CSC3170
Tutorial 7

School of Data Science

The Chinese University of Hong Kong, Shenzhen

Outline

- Index Concurrency (Latches)
- Sqlite3

Index Concurrency

- Allow multiple threads to safely access data structures.
- Latches
  - protect single workers (e.g., threads)
  - Several implementations.
  - Hash Table Latching.

Latches

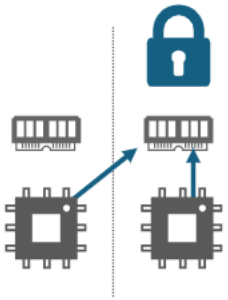
- Test-and-set Spinlock
- Blocking OS Mutex
- Reader-Writer Locks

Test-and-set Spin Latch (TAS)

- Very efficient (single instruction to latch/unlatch).
- Non-scalable, not cache friendly, not OS friendly.
- Example: std::atomic<T>

std::atomic<bool>

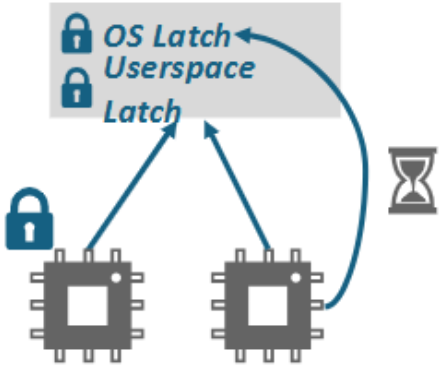
```
std::atomic_flag latch;
 ⋮
while (latch.test_and_set(…)) {
   // Retry? Yield? Abort?
}
```



Blocking OS Mutex

- Simple to use.
- Non-scalable (about 25ns lock/unlock)
- Example: std::mutex -> pthread_mutex -> futex

```
std::mutex m;
 ⋮
m.lock();
// Do something special...
m.unlock();
```



Reader-Writer Latches

- **Read Mode**
  - Multiple threads can read the same object at the same time.
  - A thread can acquire the read latch if another thread has it in read mode.
- **Write Mode**
  - Only one thread can access the object.
  - A thread cannot acquire a write latch if another thread has it in any mode.

Compatibility Matrix

	Read	Write
Read	✓	X
Write	X	X

Hash Table Latching

- Page/Block Latches
- Slot Latches

Sqlite3

- Official Link: https://docs.python.org/zh-cn/3.7/library/sqlite3.html
- Create a database:

```
import sqlite3
conn = sqlite3.connect('test.db')
```

- Create a table:

```
conn.execute('CREATE TABLE users (id INTEGER PRIMARY KEY, name TEXT, age INTEGER)')
conn.commit()  # 提交更改
```

- Insert data:

```
conn.execute("INSERT INTO users (name, age) VALUES (?, ?)", ('Bob', 25))
conn.commit()
```

Sqlite3

- Look up in database:

```
cursor = conn.execute("SELECT * FROM users")
for row in cursor:
    print(row)
```

- Close the database
  - conn.close()

For other languages (C++/Java)

- C++: https://dev.mysql.com/downloads/connector/cpp/

- Java: https://dev.mysql.com/downloads/connector/j/

Q&A