

Phase 2

Transaction Management & 2PL

PG-1

Jianhua Li

Shabir Abdul Samadh

Architecture

Middleware contains the Transaction Manager (TM)

- All calls to the middleware are sent to the TM
- The TM maintains a list of Resource Managers (RM)
- Whilst the Car, Flight, Hotel RMs are separate servers the customer is part of the middleware.
- The TM maintains a Transaction object per incoming transaction
- The TM has one single global Lock Manager (LM) to maintain locks on items

Architecture - 2PL

The TM maintains the Lock Manager and a Transaction object per call

- Each Transaction object has a set of different lists:
 - Accessed item list
 - Write list per resource
 - Delete list per resource
- All information about accessed item are kept in these lists
- Each item access is read first time (if LOCK granted) and put to the accessed list.
- Subsequent changes are only reflected in this list
- Writes and deletes are marked in Write & Delete lists
- If aborted these lists are just flushed else written to RM at commit

Performance

Conclusions:

The RTT reduces significantly comparing requests to a single RM against multiple RMs

Ex: RTT in single RM-Read is higher than multiple RM-Read

RTT in single RM-Write is higher than multiple RM-Write

In the case of a mix of Read & Write involved transactions the client seems to come to a exhausted scenario soon where the middleware is able to handle the requests. Hence, they seems to crash sooner.

Comparing Read & Write mixed transaction in 1-RM and Multiple-RM, the 1-RM seems to perform better.