Advanced Enterprise Computing - Lecturenotes SoSe2016

Julius Hülsmann

10. Mai 2016

Inhaltsverzeichnis

1 Replication and State Management (25.04. - 09.05.)

1.1 Motivation and Background

1.1.1 Replication

Definition - Replication Process of maintaining multiple Copies of an Entity (Data / Process / File ...)

Advantages of Replication in General

- *System Availability / Fault tolerance* in case
  A Server fails
  B Data is corrputed.
- *Performance / Scalability*
  A Workloads are spread across distributed Replicas
  B Geodistribution for processing demands in client's proximity

Disadvantages of Replication in General

- Consistency vs. Performance

Kinds of Replication In general there are the following kinds of „physical“ Replication. We do only consider (B).
!!!!!!!!!!!!!!!ueberpruefen am Ende!!!!!!!!!!!!!!!

Replication Strategies PAGE 17
Synchronous vs. Asynchronous

Synchronous / eager

Asynchronous / lazy

Primary Copy vs. Update Everywhere

Primary Copy / master

Update Everywhere / group

Name explanation Protokolltitle implementation Atomacity either perform Transaction entirely or roll-back atomic commitement protocol 2PC

ACID

Atomiticity:
Consistency: does not mean Data-Consistency but that the transaction produces consistent changes.
Isolation: Transactions are isolated from one another
Durability: Once the transaction is ready (commits) it remains.

Both the Atomitcity and the Isolation are managed by the **Transaction Manager**

- Aquires locks on behalf of the transactions
- guarantees serializable execution. Therefore, the scheduler only needs to enforce 2PL behavior because serializability is automatically guaranteed in case the transaction follow 2PL

What happens to ACID in case of Replication? Atomicity can be guaranteed using 2PC (but expensive) Problem: Serialization order must be the same at all replicas.
Synchronous

1. 1 propagate Data to everybody
2. Wait until everybody responded
3. commit

- ACID properties apply to all copy updates (no Inconsistencies) - High response time (high execution time, response time)
- Availability (in case one Copy fails)

Asynchronous

1. Update local copy
2. Commit
3. propagate Data to everybody

- Response Time
- Availability
- Data inconsistency (local read does not always 1 Update local copy return the latest value) - No guarantee that the changes arrive at each copy - Replication is not guaranteed (it is possible that the changes do not reach the replicas at all)

Primary Copy

-
-

Update everywhere

-
-

6 Lecturenotes

Lecture 05? start @ 81 Für Donnerstag paper mitbringen und Paxos anschauen.
2016-05-09

Paxos (Represent as State-machine) - P. 77
Proposer
Phase 1 - Proposer choses Number largr than any value chosen before by Propposer. - Broadcast the integer *prepare(n)*, e.g. prepare(50)
Acceptors a) Not respond at all b) *recject* Reject, in case a higher value has been accepted. 50 ¡ something b) *prommise(n)* in case 50 ¿ everything. Also Send everything that has already been accepted.
If prposer receives majority of prommise respons, -¿ proceed to Phase 2 ELSE -¿ Phase 1
Phase 2 - Check whether any ¡n, value¿ have been returned. - YES: take max n's value - accept (n, value)

Xtensions Paxos **Multi-paxos** Determine Leader once Stay in phase 2, attatch the leader identifier Leader is the one to accept values
Purpose: Optimize Speed (get rid of the first phase, Master-Slave setup)
Fast Paxos
Generalized Paxos - Assumption: The execution order does not matter.

CRDT Conflict free / Communitive replicated Datatypes
Some operations are commutative, others not.
State- Based vs. Operation based.
theoretically it is possible to converge them but ... practice
IDEA INTEGER - example: e.g. not store int values but operations (increment / decrement))
SET - example
State - based Set

7 Begriffe und Abkürzungen

Replication Strategy to maintain mutiple copies of an entity on multiple Servers.

Replica

CRDT *conflict-free replicated data*

Paxos

Commit In case a Transaction commits, it is ready.

Concurrency control protocol guarantees isolation of Transactions

2PL Two phase locking (one concurrency control protocol)

Snapshot Isolation other cuncurrency control protocol implementation

atomic commitement protocol guarantees atomaticity

2PC Two phase Commit

Transaction Manager Middleware Component; Manages Atomacity and Isolation of Transactions

ACID Atomacity + Consistency + Isolation + Durability

serializability a plan of executing multiple transactions in pseudo- parallel is called serializable in case the parallel execution comes to the same result as executing the transactions one after the other