Large-Scale Data Processing

Yang Wang
Consistency

- Linearizable
- Sequential
- Causal
- Eventual

Strong

Weak
Consistency

Definition does not depend on implementation: treat system as a black box.
Sequential consistency

• Definition: The result of any execution is the same as if the (read and write) operations by all processes on the data store were executed in some sequential order and the operations of each individual process appear in this sequence in the order specified by its program.
  - There is a global order to execute requests.
  - Every one sees all requests in the same order.
Sequential consistency - Examples

A

B

C

Equal to

W a=1

W b=1

R a=1

R b=1

Equal to
Sequential consistency - Examples

A \[ W_a=1 \]

B \[ W_b=1 \]

C

\[ R_a=0 \]
\[ R_b=0 \]

Equal to

\[ R_a=0 \]
\[ R_b=0 \]
\[ W_a=1 \]
\[ W_b=1 \]
Sequential consistency - Examples

A

W a=1

B

W b=1

C

R a=0

R b=1

Equal to

R a=0

W a=1

W b=1

R b=1
Sequential consistency - Examples

Equal to

W a=1  R a=1  R b=0  W b=1
Sequential consistency - Examples

A

B

C

\[ W_a = 1 \]

\[ W_b = 1 \]

\[ R_a = 1 \]

\[ R_a = 0 \]
Sequential consistency - Examples

A
W a=1

B
W b=1

C
R a=1  R b=0

D
R b=1  R a=0

X
Linearizable

• Definition: sequential consistency + real time constraint

• If a request R1 completes before R2 starts, then R1 must be ordered before R2.

• Concurrent requests can be ordered in either way, but every one must see them in the same order.
Linearizable - Examples

A

B

C

Equal to

W a=1

W b=1

R a=1

R b=1

W a=1

W b=1

R a=1

R b=1
Linearizable - Examples

A

B

C

W a=1

W b=1

R a=0

R b=0

X

X
Linearizable - Examples

A \[ W \ a=1 \]

B \[ W \ a=2 \]

C \[ R \ a=1 \]

✓
Linearizable - Examples

A

B

C

W a=1

W a=2

R a=2

✓
Linearizable - Examples

A \[ W_{a=1} \]

B \[ W_{a=2} \]

C \[ R_{a=1} \]

D \[ R_{a=2} \]
Eventual Consistency

• Definition: If no new updates are made to a given data item, eventually all accesses to that item will return the last updated value

• No global ordering requirement

• Require conflict resolution for concurrent updates
Eventual - Examples

• Version control (e.g. svn, git):
  – Each user makes changes locally
  – A user commits and updates code manually
  – A user may need to resolve conflicts when it updates the repo
  – Different users may see updates in different order
Causal Consistency

• Definition: Respect partial order defined by a user
  – If a user sees a request R, then it must also see all requests R depends on.

• No global ordering requirement
• Require conflict resolution for concurrent updates
Causal - Examples

• Access control in social network
  – Remove your parents from access list
  – Post a photo that you hope your parents never see
Using different consistency models

- Linearizable
- Sequential
- Causal
- Eventual

Easier to use

Lower latency
Using different consistency models

• Linearizable: almost don’t need to worry about anything
• Sequential: may read stale data
• Causal: different clients may see events in different orders; need to resolve conflict
• Eventual: almost no guarantee

Can you think about an application that requires global ordering?
Implementing consistency

Linearizability

U1 \rightarrow R1 \rightarrow R2 \rightarrow U2

Write
Implementing consistency

Linearizability

R1

U1

U2

R1

R2

Read
Implementing consistency

Sequential

U1

R1

Write

U2

R2
Implementing consistency

Sequential

U1

R1

U2

R2

Read
Implementing consistency

Eventual/Causal

Write/Read

Propagate data in background
Consistency vs Latency

CAPS theorem: Impossible to achieve all the following properties together.

- Sequential consistency (or stronger)
- Availability
- Ability to tolerate network partitions
Challenge: Resolve conflicts

- It is application specific
- Example: Number of remaining tickets

```
58
Buy 1
57
Merge 57 57?
56
```

U1 -> Merge -1
```
Update 1
56
```

U2 -> Buy 1
```
56
```
Consistency in reality

• Who are using strong consistency?
  – Online gaming
  – Google

• Who are using weak consistency?
  – Banks (ATMs)
  – Version control
  – Amazon