AntidoteDB
The developer-friendly open-source cloud database

Just-Right Consistency

Marc Shapiro, UPMC-LIP6 & Inria
Annette Bieniusa, U. Kaiserslautern
Nuno Preguiça, Valter Balegas, U. Nova Lisboa
Christopher Meiklejohn, U. Catholique de Louvain

Geo-distributed DB

+ fault-tolerance
+ low latency read
Updates?

Aysnchronous updates

Available under Partition
Concurrency anomalies
Eventual Consistency: hell for developers!
Synchronous updates

consistent under Partition
Conservative: one at a time
Simple programming model
Expensive infrastructure
Not available under partition

FMK Fælles Medicinkort

FMK invariants

Geo-distrib: invariants?

EC does not maintain!
CP is overkill!
The developer-friendly open-source alternative

Preserve sequential data invariant patterns

Transactionally Causal Consistency (TCC)
- Maintains AP-compatible invariants
- Strongest form of consistency that is AP

CAP-sensitive: Synchronise only when strictly necessary for application \( \Rightarrow \) tools
No specialised infrastructure, no lock-in

AP data model: CRDTs

Concurrent, asynchronous updates
- Standard register model: assignments \( \Rightarrow \) CP
- AP \( \Rightarrow \) concurrent updates merged

CRDT: register, counter, set, map, sequence
- Extends sequential type
- Encapsulates convergent merge

CRDT API

CRDT: register, counter, set, map, sequence
- Extends sequential type
- Encapsulates convergent merge

antidote: update_objects([[
    {meds, antidote_crdt_set, bucket},
    add, {"Causatin", client1}
]], TxId1).
Relatve order: AP-Compatible

create-p before add-p

Relative-order invariant pattern:
- "Patient record points to valid prescription"
  - \(x \text{ valid} \land x \text{ points to } y \Rightarrow y \text{ valid}\)
- \(\text{admin-login-enabled} \Rightarrow \text{non-default-password}\)
- Make RHS true; then LHS true

Transmit in the right order!
AP-compatible: Causal Consistency

Joint update: AP-Compatible

create-p updates doctor, patient & pharmacy record

Transmit joint updates together
- write-atomic
  - Read from common set of txns
  - snapshot property
  - All-or-Nothing (A of ACID)

AP-compatible

Txn causal Consistency

AntidoteDB.org

Transactional Causal Consistency (TCC) = strongest AP model
Guarantees AP-compatible invariant patterns
Antidote: first industrial-strength TCC database
- alpha
Guarantees Relative-Order and Joint-Update invariant patterns

Transaction API

\{ok, T1\} = antidote:start_transaction (ignore, []).
...
antidote:update_objects ([[
  \{meds, antidote_crdt_set, bucket\},
  add, \{"Causatin", client1\}
  ]], T1).
...
antidote:commit_transaction (T1).
**CAP-sensitive invariants**

\[
\text{process-p (..., nb)} \{
    \text{if } cnt \geq nb \quad \text{// precondition at source}
    \text{cnt} -= nb \quad \text{// at every replica}
\} \quad \geq 0
\]

Precondition \textit{stable} w.r.t. concurrent \textsl{add-med}

Concurrency OK

\[\text{cnt} += 3\]

\[\text{add-med (..., 3)}\]

**CAP-sensitive invariants**

\[
\text{process-p (..., nb)} \{
    \text{if } cnt \geq nb \quad \text{// precondition at source}
    \text{cnt} -= nb \quad \text{// at every replica}
\} \quad \geq 0
\]

Precondition \textit{not stable} w.r.t. concurrent \textsl{process-p}

\[\text{cnt} -= 2\]

\[\text{pp(..., 2)}\]

\[\text{cnt} \geq 1\]

\[\text{cnt} \geq 0\]

**CISE Static Analysis**
Bounded Counter

Specific but common case
Shared counter:
• \( x \geq 0 \)
• increment (n)
• decrement (n) \( \text{// precondition } x \geq n \)

Escrow:
• Local share, decrement \( share -= n \)
• Synchronise only when \( share < n \)
• Donate share

Encapsulated, proven correct (CISE)

Mostly AP

CISE tools

Static analysis of any application:
• Operations, invariants
• Does each individual op maintain invariant?
• Do concurrent updates converge?
• Is precondition of \( u \) stable w.r.t. concurrent \( v \)?
  If not:
  ‣ Change specification (invariant)
  ‣ or Synchronise
  ‣ Designer decision, per pair \((u, v)\)

Ex: medication count= \( inc||inc, inc||dec, dec||dec \)

Just-Right Consistency

Methodology for provably ensuring
As Available as Possible, Consistent Enough
TCC \( \Rightarrow \) AP-compatible invariants
CAP-sensitive invariants: Bounded Ctr, CISE

AntidoteDB: CRDTs, Causal Consistency, Transactions, Bounded Counter

\( \Rightarrow \) AP-compatible

Rich interface with concurrent abstract data types (CRDTs)
Geo-replication
Low latency, Available under Partition
Erlang / Riak Core
New, growing community
Active development:
• Secondary Indexes, configurable back-ends and protocols, partial replication, clients in many languages, SQL interface, etc.