

Sirius: Distributing and Coordinating Application Reference Data

Michael Bevilacqua-Linn, Maulan Byron, Peter Cline,
Jon Moore, and Steve Muir
Comcast Cable

@jon_moore

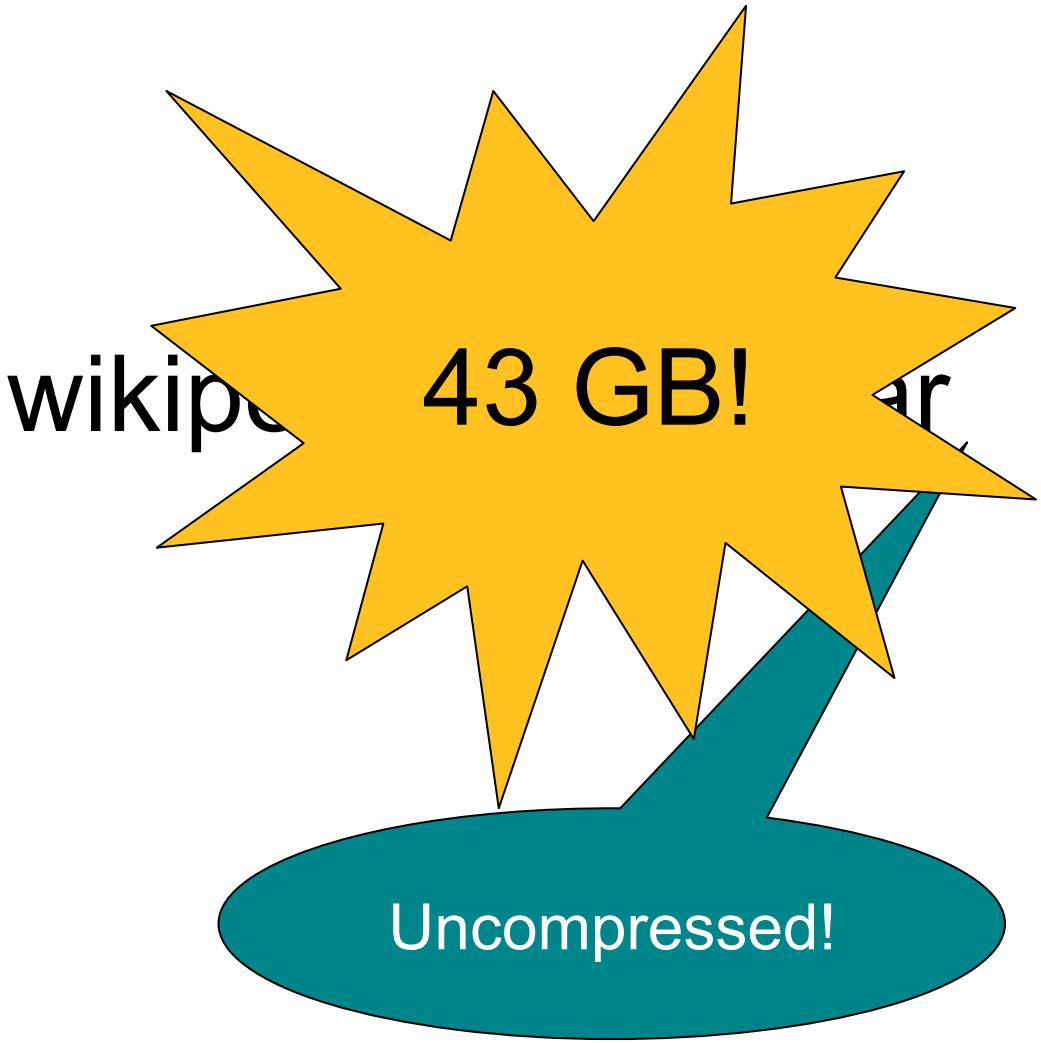
What is “Reference Data” anyway?

COMCAST

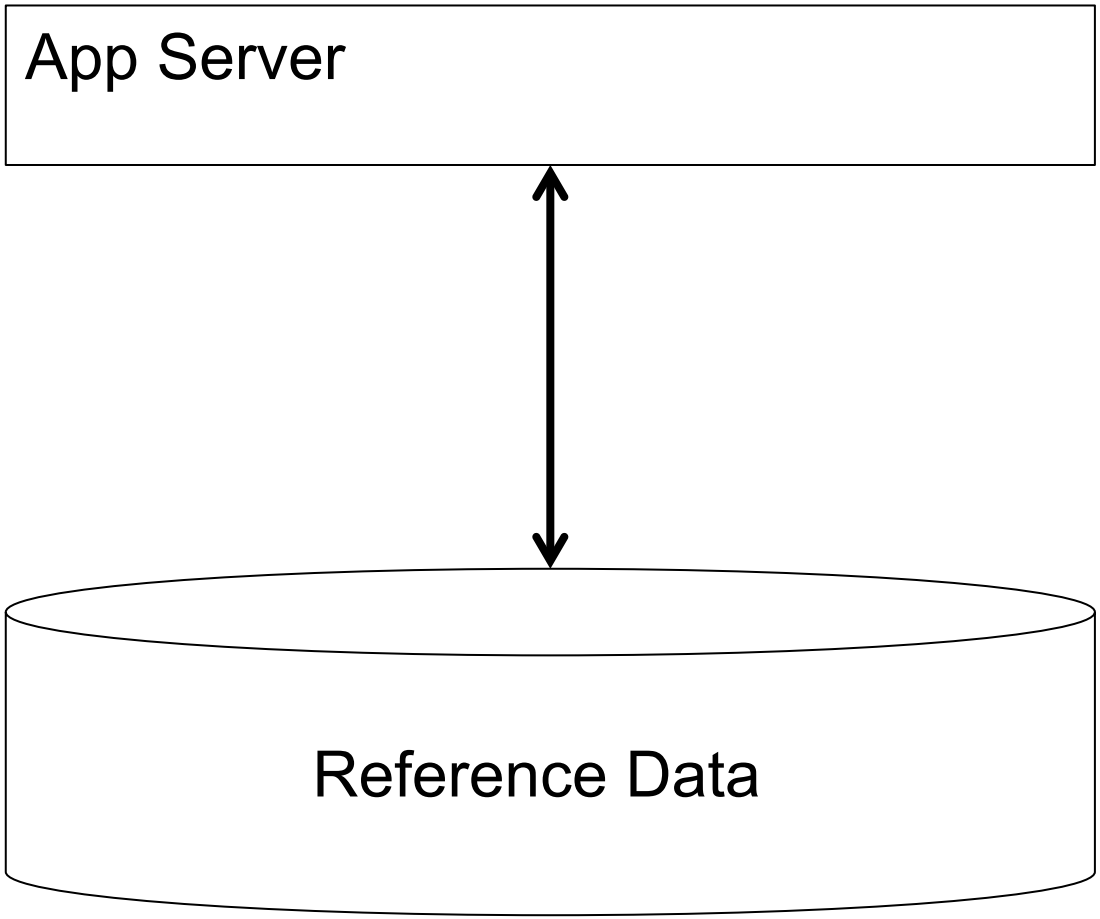


Photo: http://www.flickr.com/photos/37080143@N08/3409059834
by SimCity Research Laboratory: http://www.flickr.com/photos/37080143@N08/
License: http://creativecommons.org/licenses/by-sa/2.0/

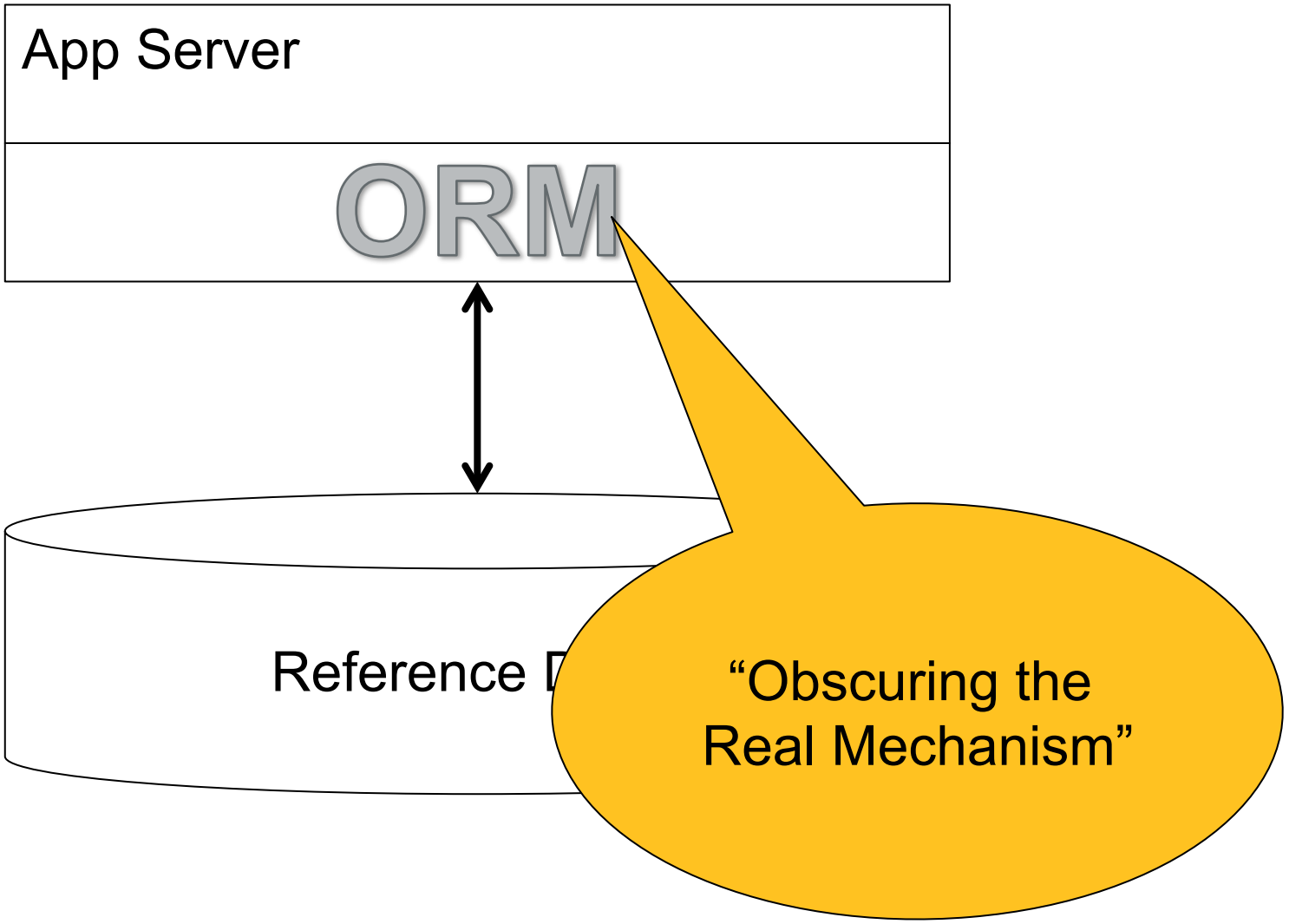
COMCAST



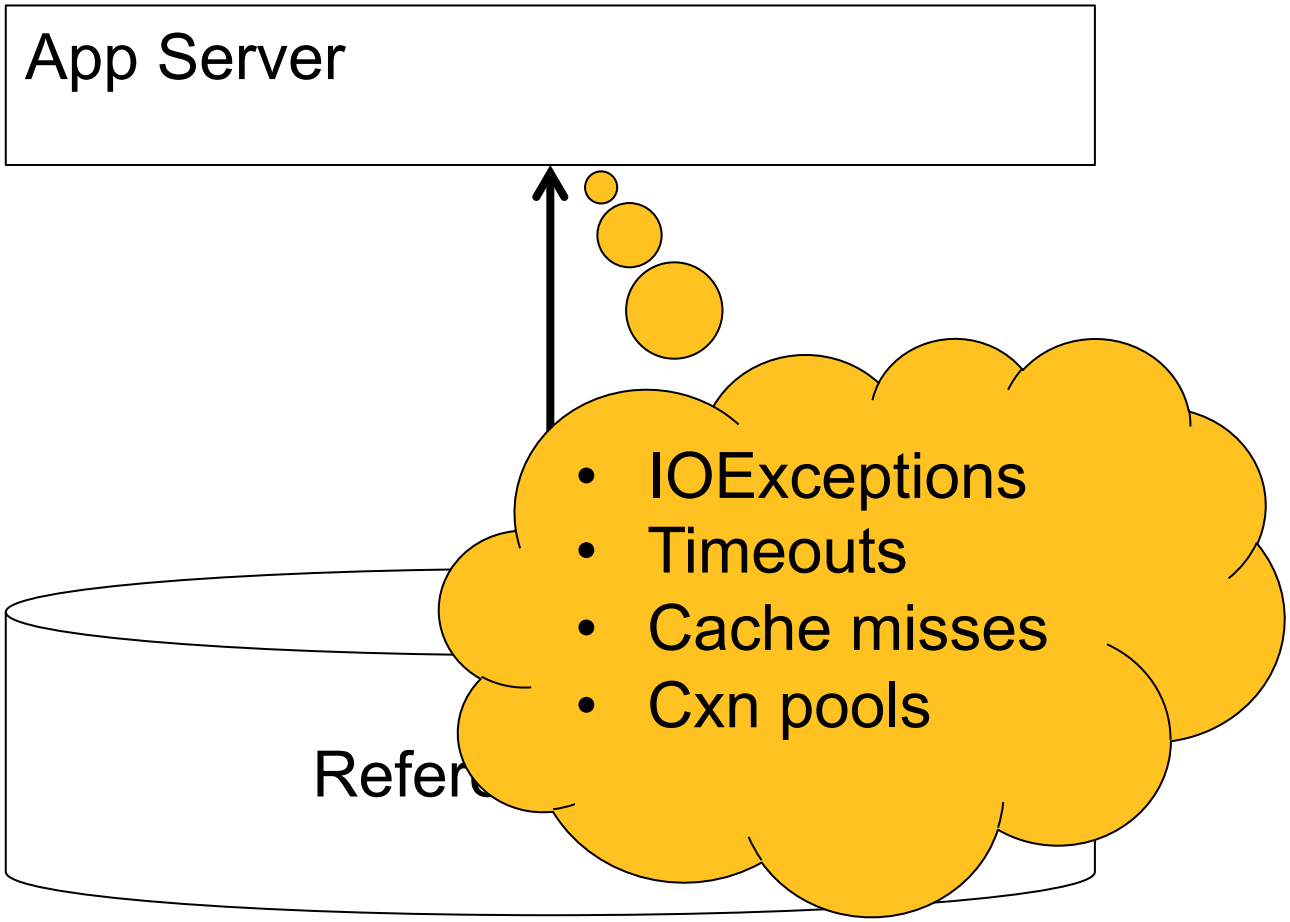
COMCAST



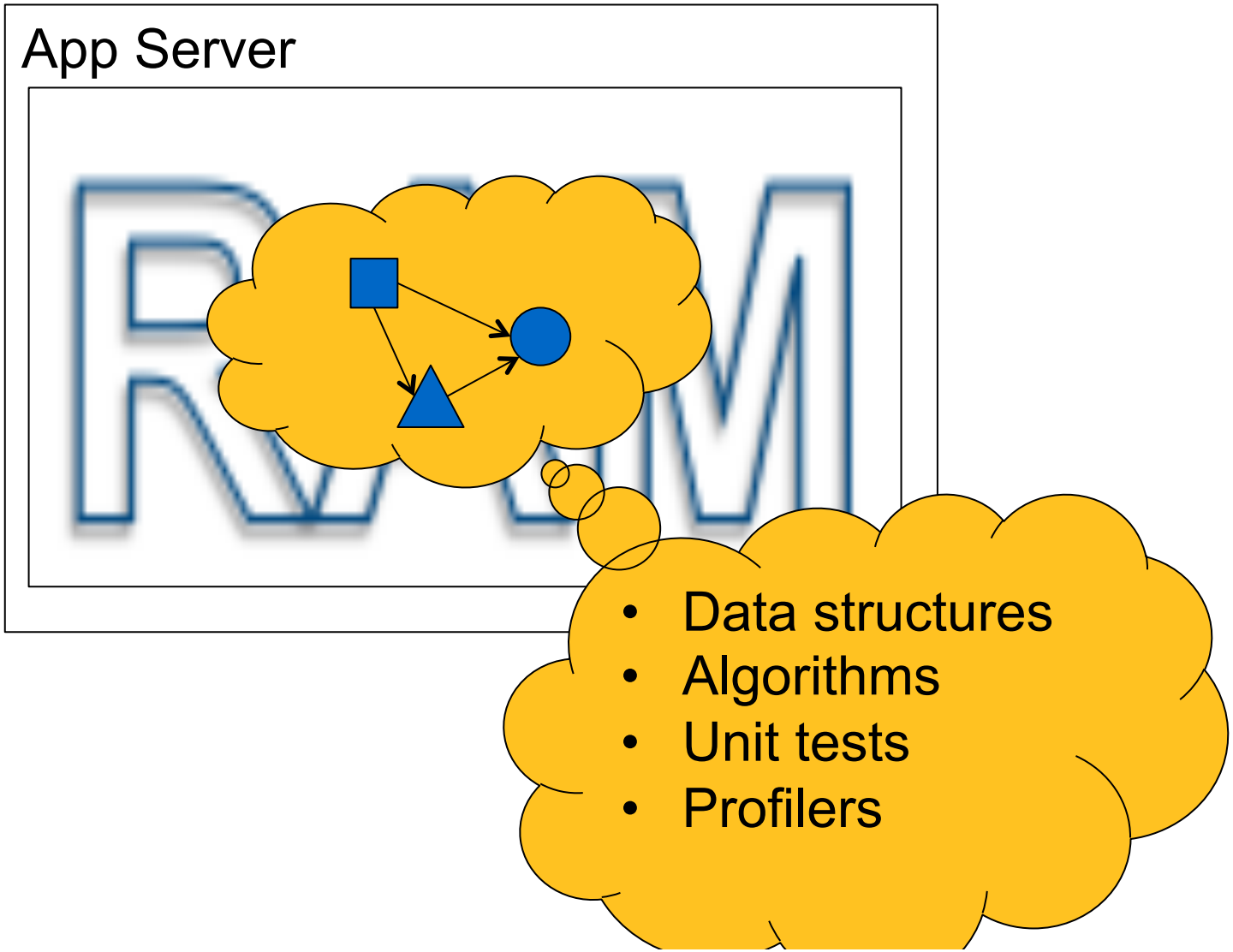
COMCAST

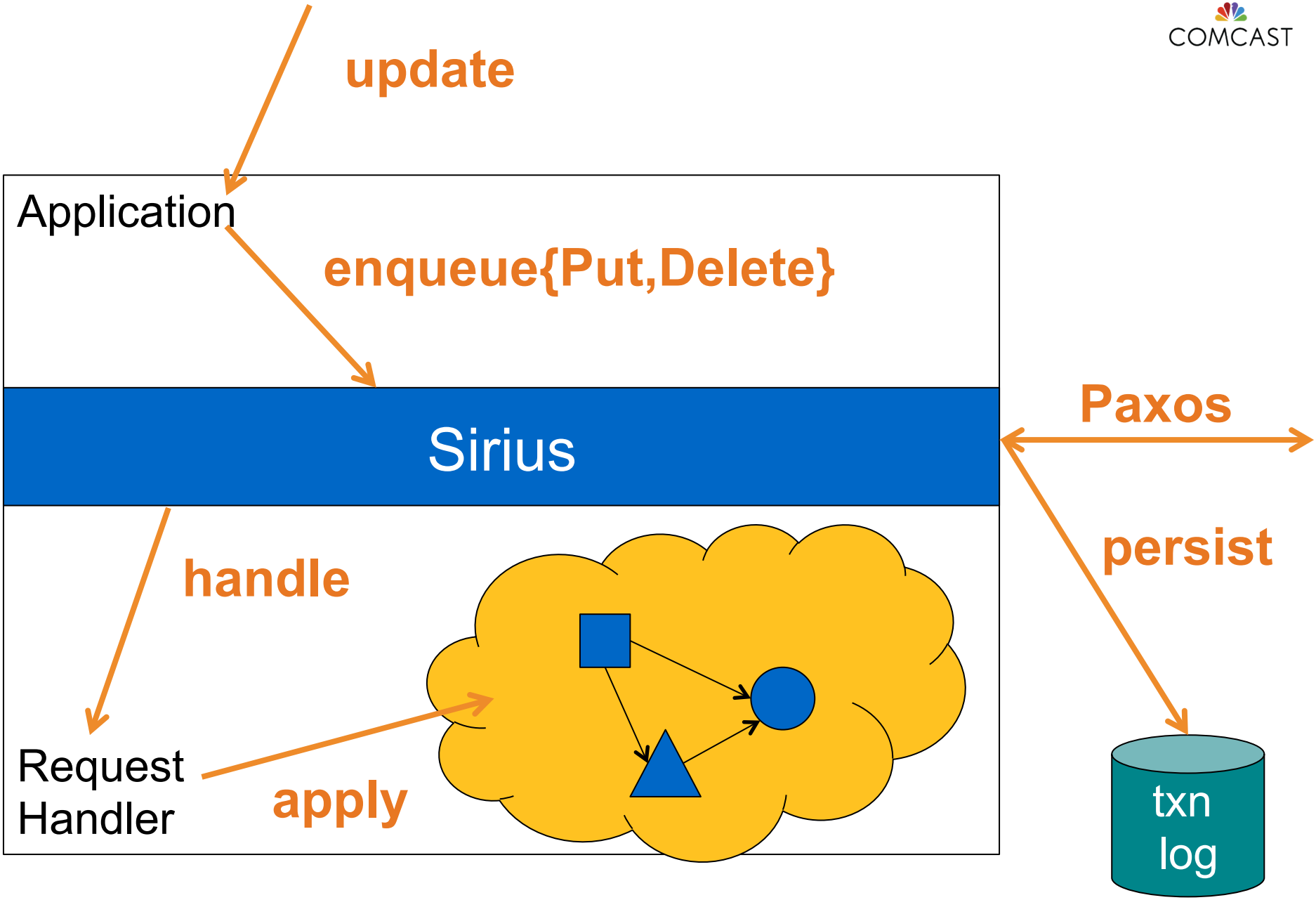


COMCAST

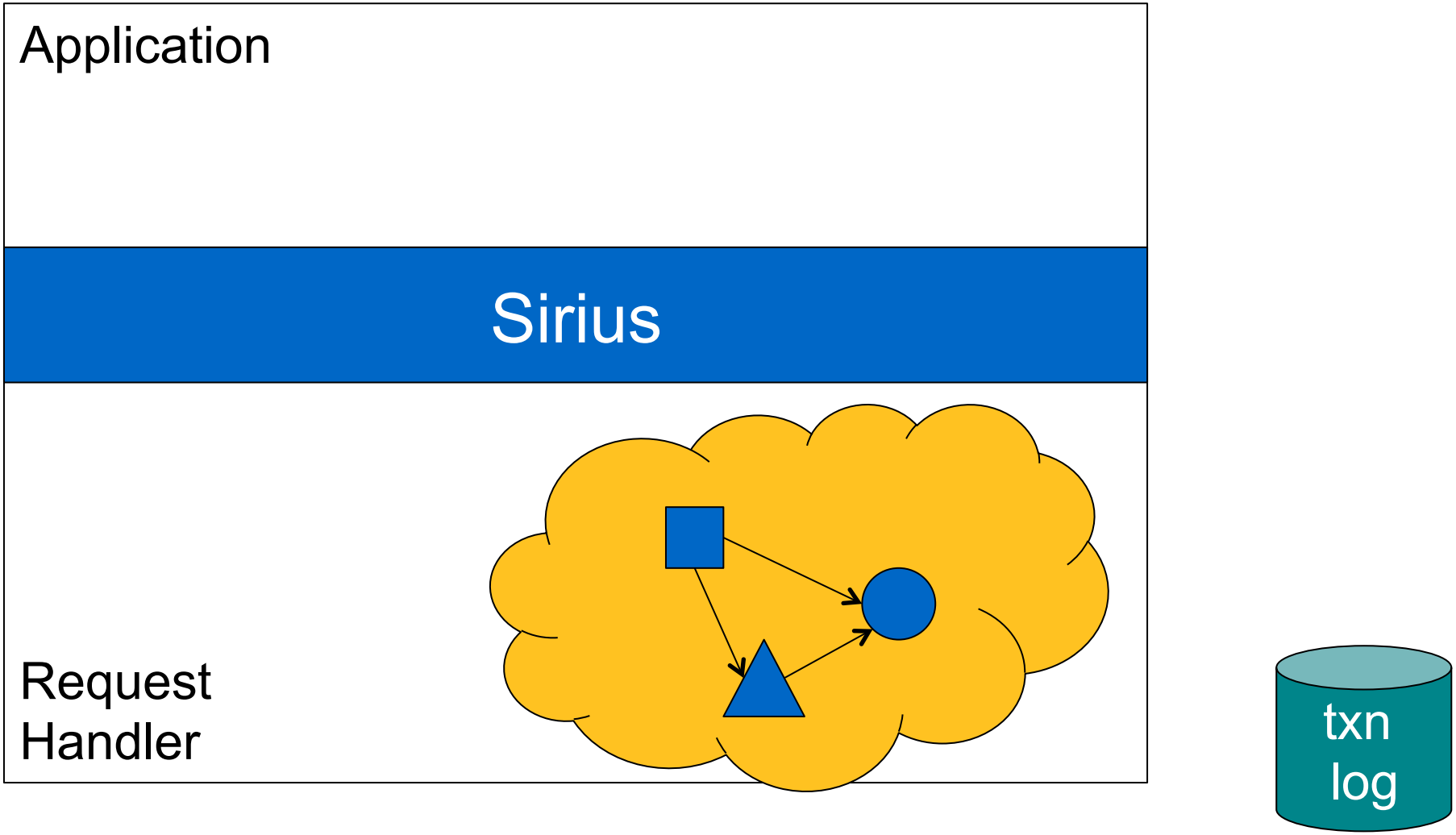


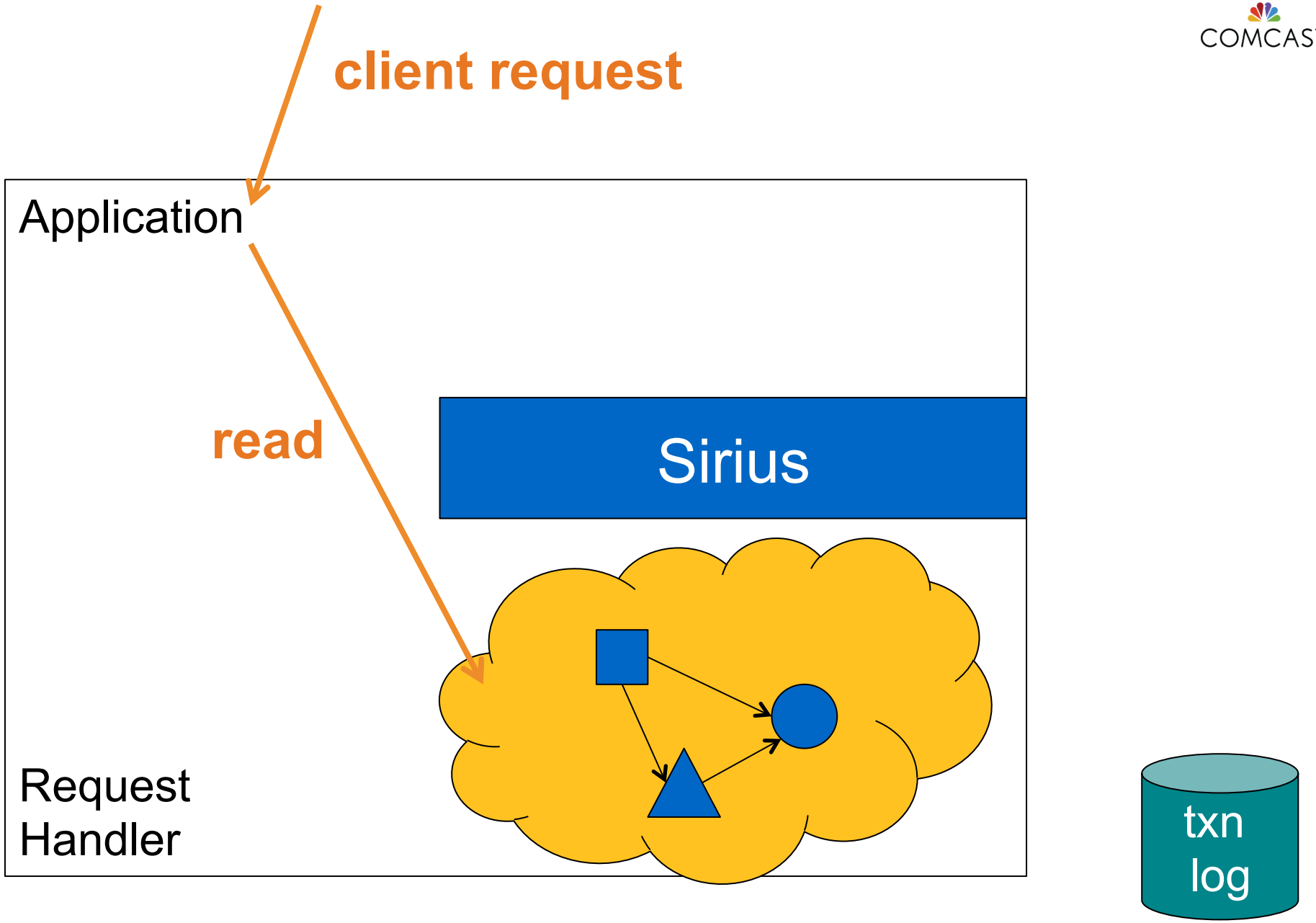
COMCAST


App Server
RVM
• Data structures
• Algorithms
• Unit tests
• Profilers


update
Application
enqueue{Put,Delete}
Sirius
Paxos
handle
persist
Request
Handler
apply
txn
log

COMCAST




client request
Application
read
Sirius
Request
Handler
txn
log

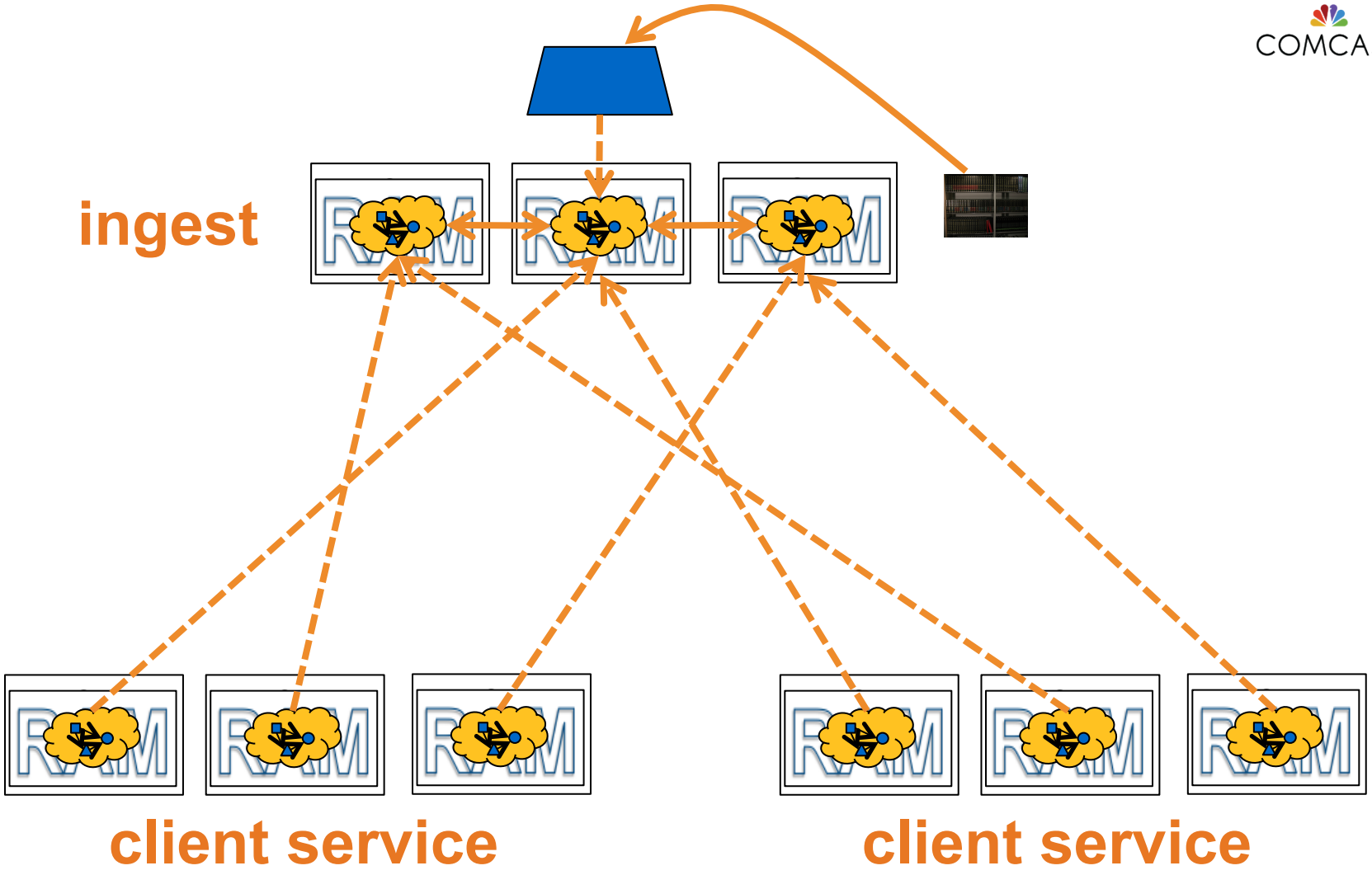
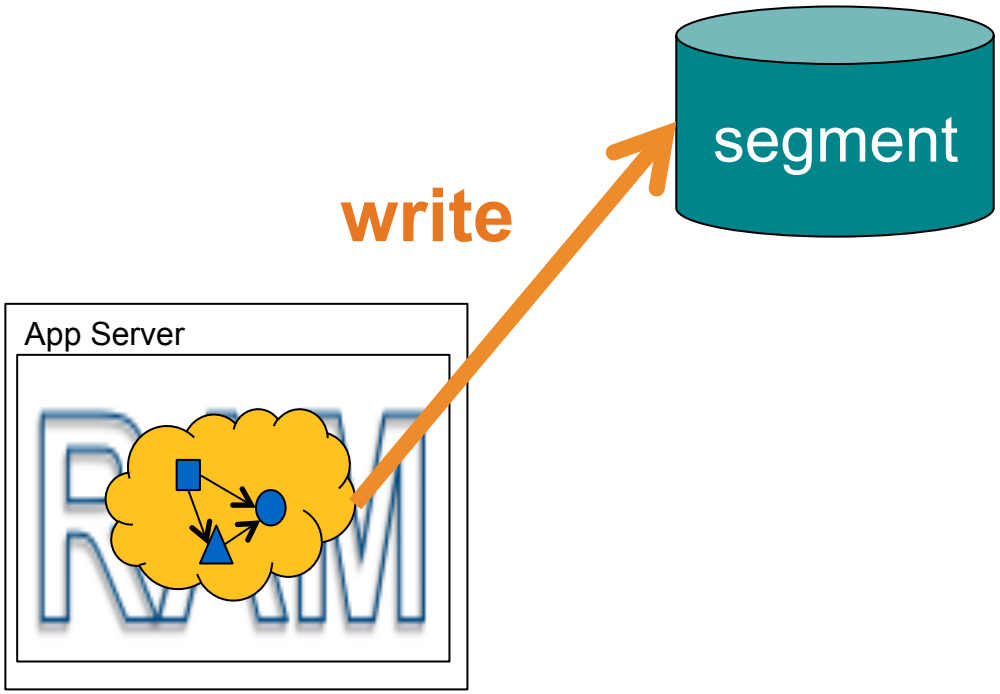
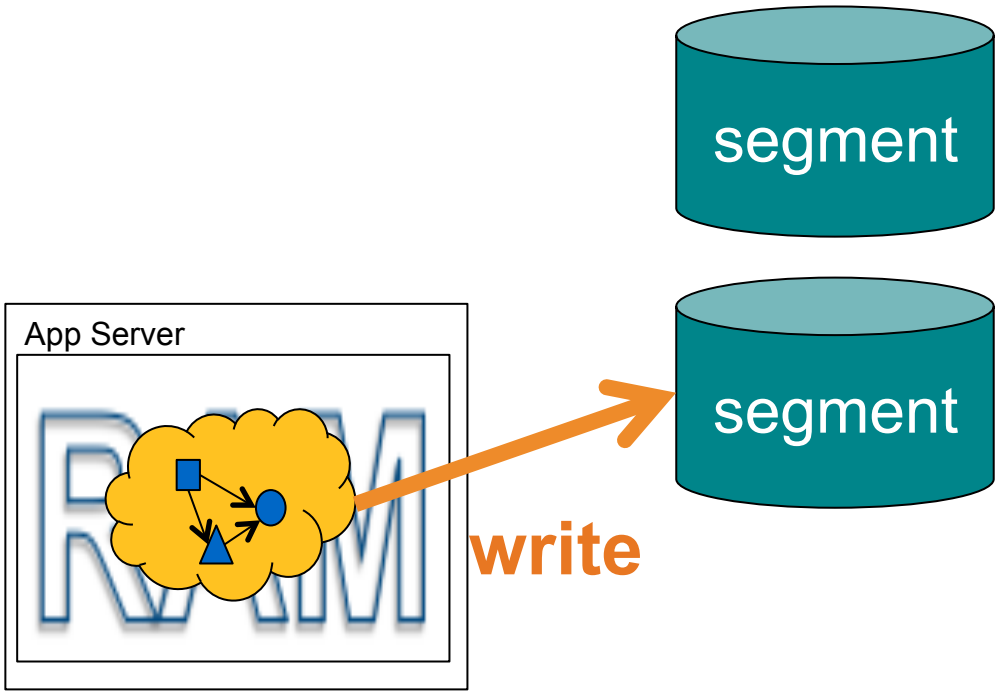


Photo: http://www.flickr.com/photos/37080143@N08/3409059834
by SimCity Research Laboratory: http://www.flickr.com/photos/37080143@N08/
License: http://creativecommons.org/licenses/by-sa/2.0/

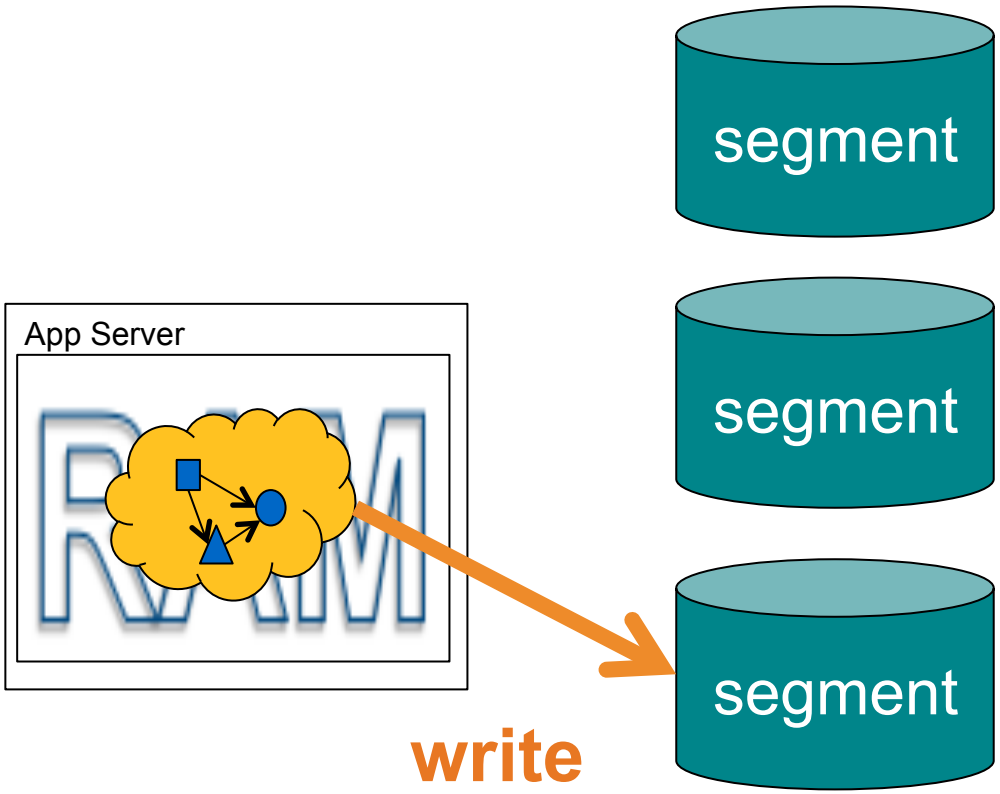
COMCAST



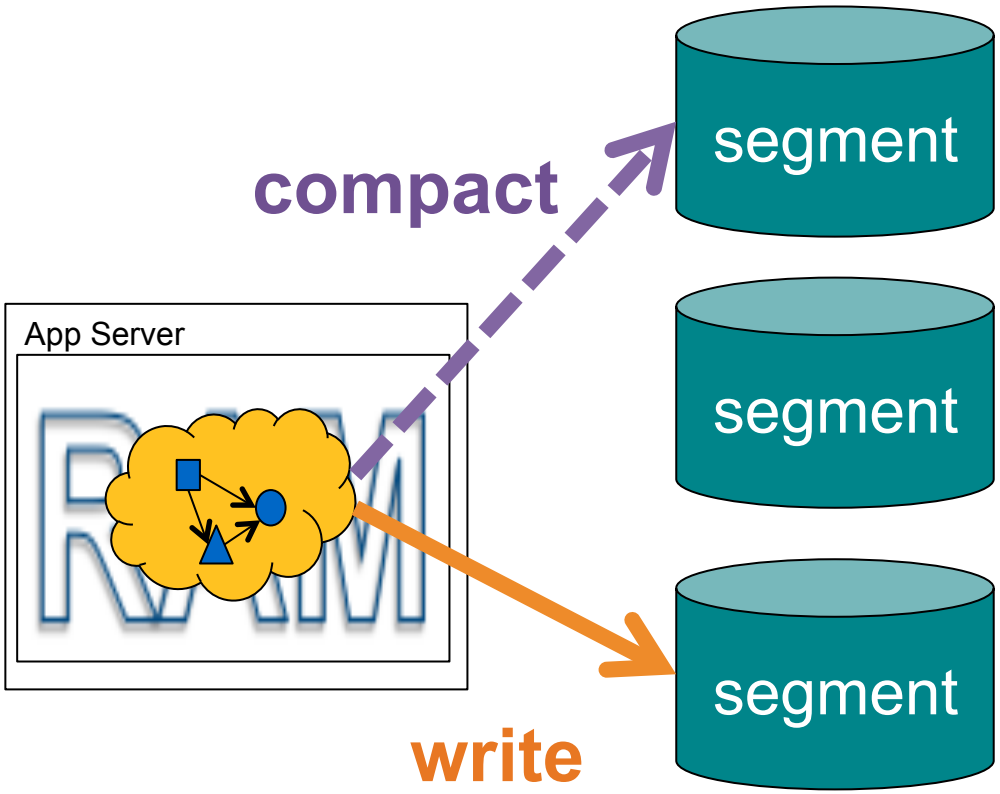
COMCAST



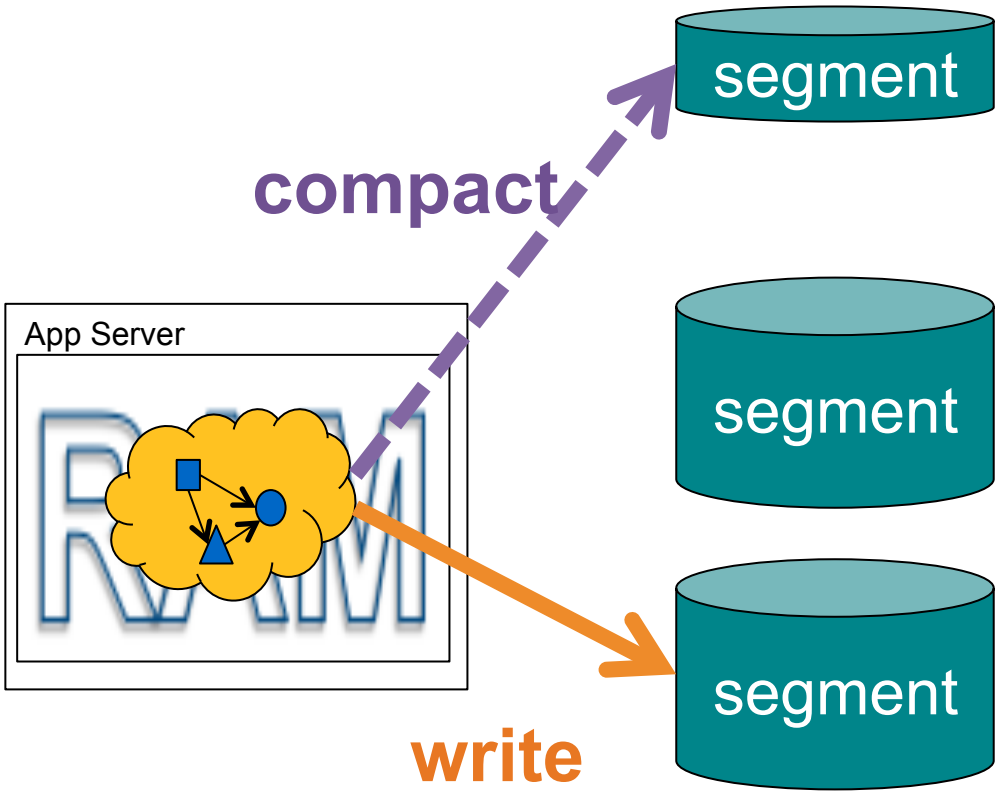
COMCAST



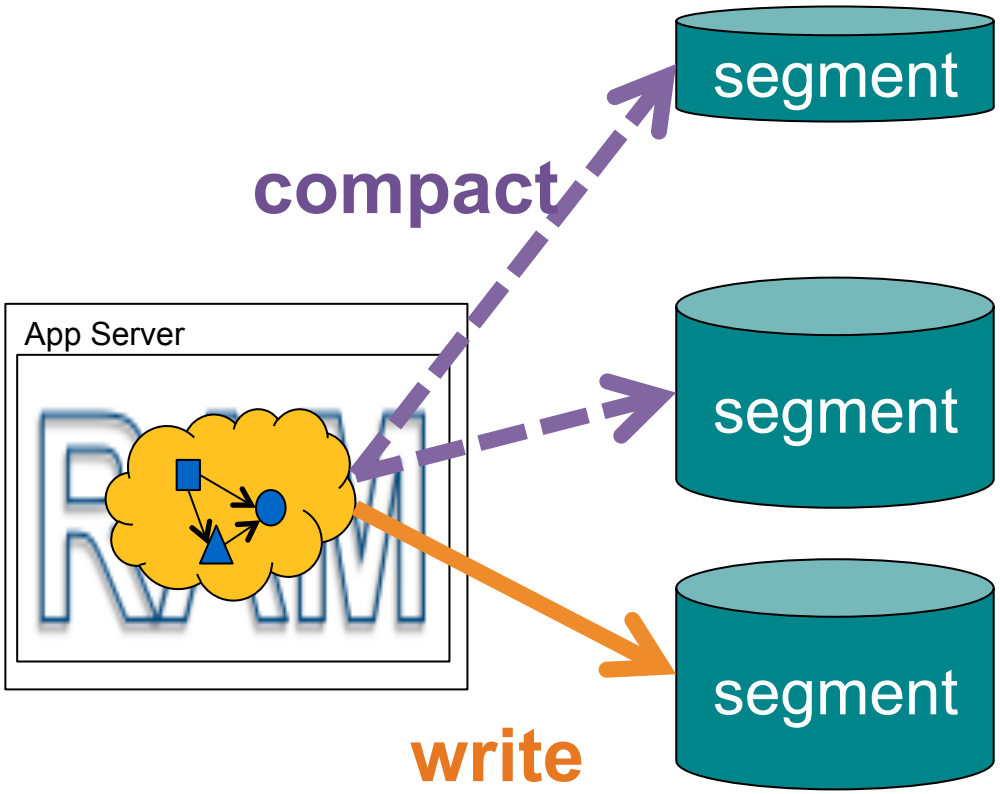
COMCAST



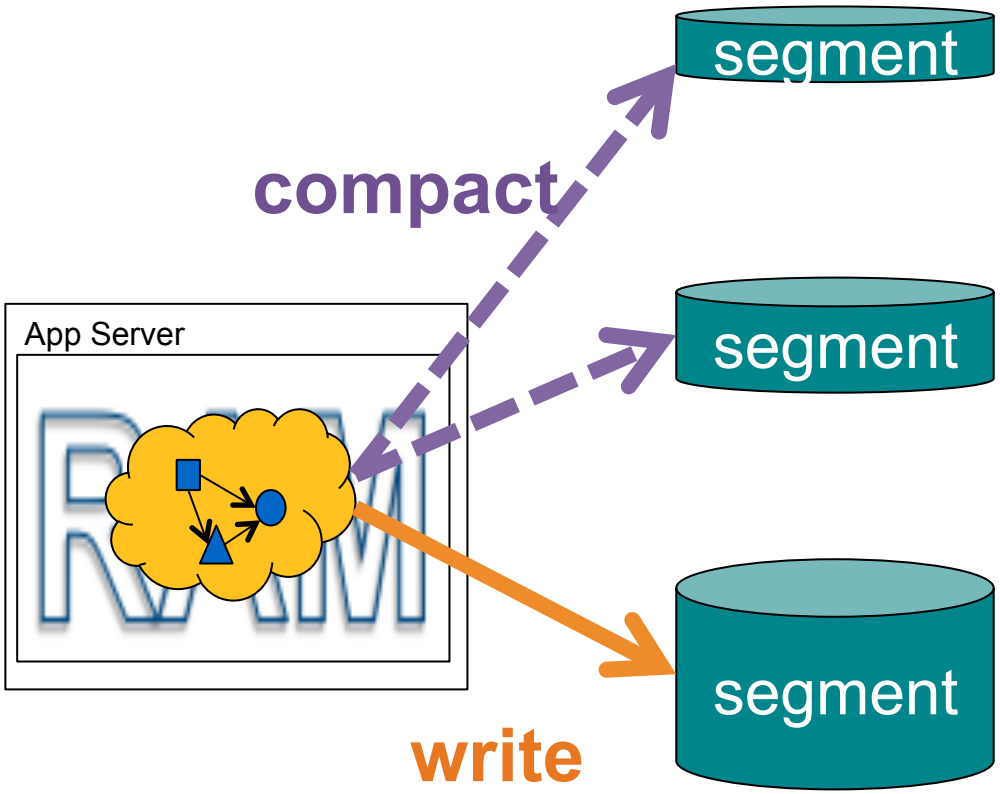
COMCAST



COMCAST



COMCAST



Implementation

COMCAST

```
for ever
  switch receive()
    case ⟨request, p⟩ :
      propose(p);
    case ⟨decision, s, p⟩ :
      decisions := decisions ∪ {⟨s, p⟩};
      while ∃p′ : ⟨slot_num, p′⟩ ∈ decisions do
        if ∃p″ : ⟨slot_num, p″⟩ ∈ proposals ∧
                  p″ ≠ p′ then
          propose(p″);
        end if
        perform(p′);
      end while;
  end switch
end for
```

```
def receive = {
    case Request(command: Command) =>
      propose(command)

    case decision @ Decision(slot, decisionCommand) =>

      decisions.put(slot, decisionCommand)
      reproposeIfClobbered(slot, decisionCommand)

      try {
        performFun(decision)
      } catch {
        case t: Throwable =>
          logger.warning("…")
      }

    case decisionHint @ DecisionHint(decisionHintSlotNum) =>
      slotNum = decisionHintSlotNum + 1

      outstandingProposals.filter((k, _) => k > decisionHintSlotNum)
      decisions.filter((k, _) => k > decisionHintSlotNum)

      localLeader forward decisionHint

    case Reap =>
      reapStagnantProposals()
  }
```

Robbert van Renesse. “Paxos Made Moderately Complex.” March, 2011.
http://www.cs.cornell.edu/courses/CS6452/2012sp/papers/paxos-complex.pdf

Conclusions

COMCAST

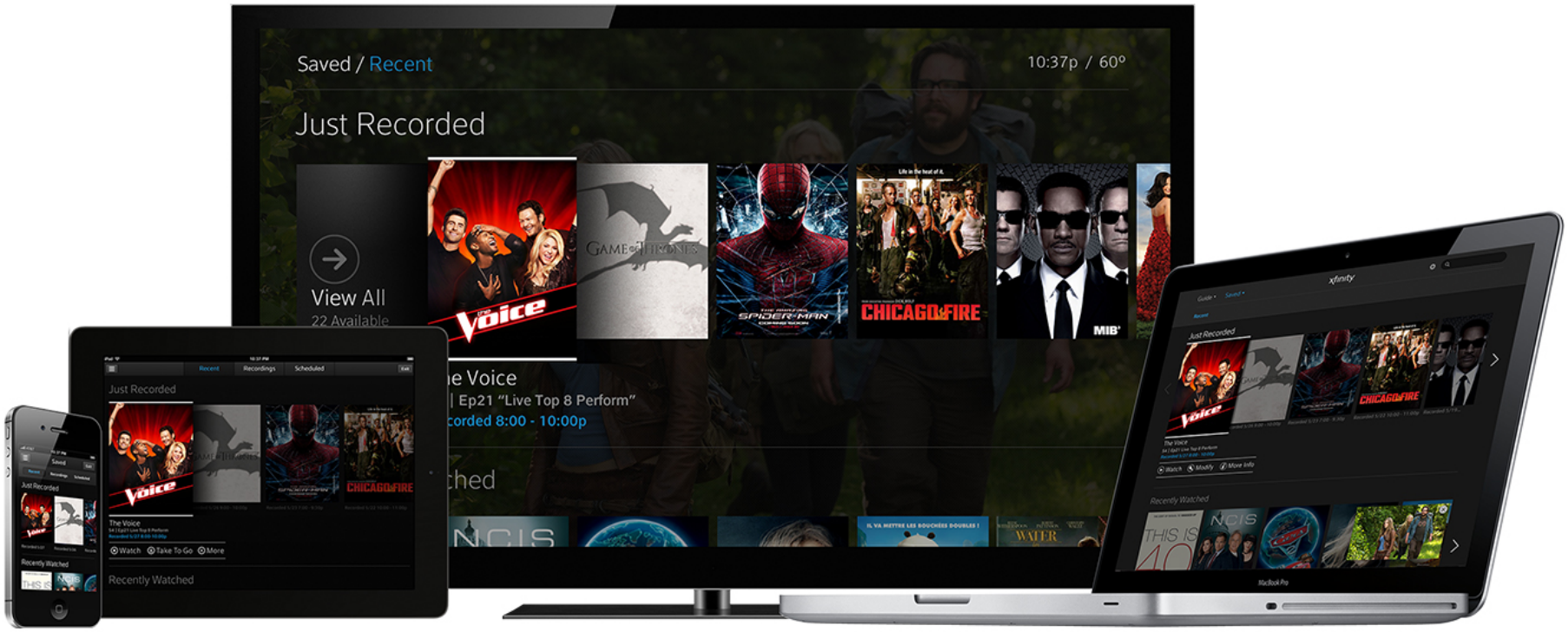


Photo: Copyright Comcast

Sirius provides access to reference data with:

COMCAST

- Native, arbitrary data structures
- Eventually consistent replication
- Persistence and replay
- Convenient library interface

http://comcast.github.io/sirius/