Peanuts, by Yahoo

Presented by Robert Charles MacDavid

Brian F. Cooper, Raghu Ramakrishnan, Utkarsh Srivastava, Adam Silberstein,
Philip Bohannon, Hans-Arno Jacobsen, Nick Puz, Daniel Weaver and Ramana Yerneni

Motivation

- Relational Database
- Easily Adapt to Changing Loads
- Async Geo-replication for global low latency
- High availability in the face of failures
- **Choose-your-own consistency guarantees**
- Support Single-point requests and Range queries

Choose your Consistency

- Read-Any
- Read-Critical(minimum_version_number)
- Read-Latest
- Write
- Test-and-set-Write(exact_version_number)

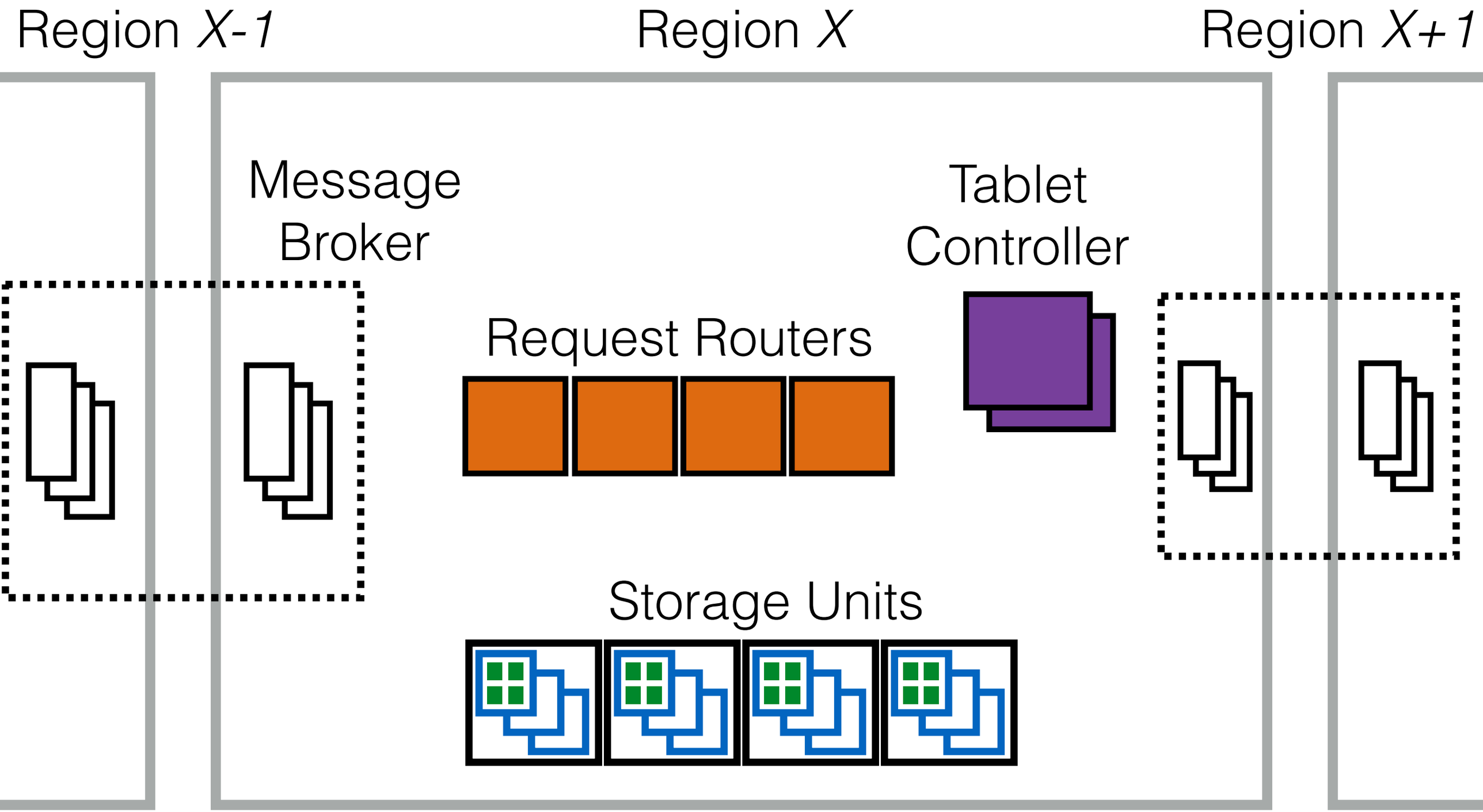
Use Cases

- Social Websites (Flickr)
- Metadata Store of a Distributed Storage System
- Shopping Listings
- Storing User Session State

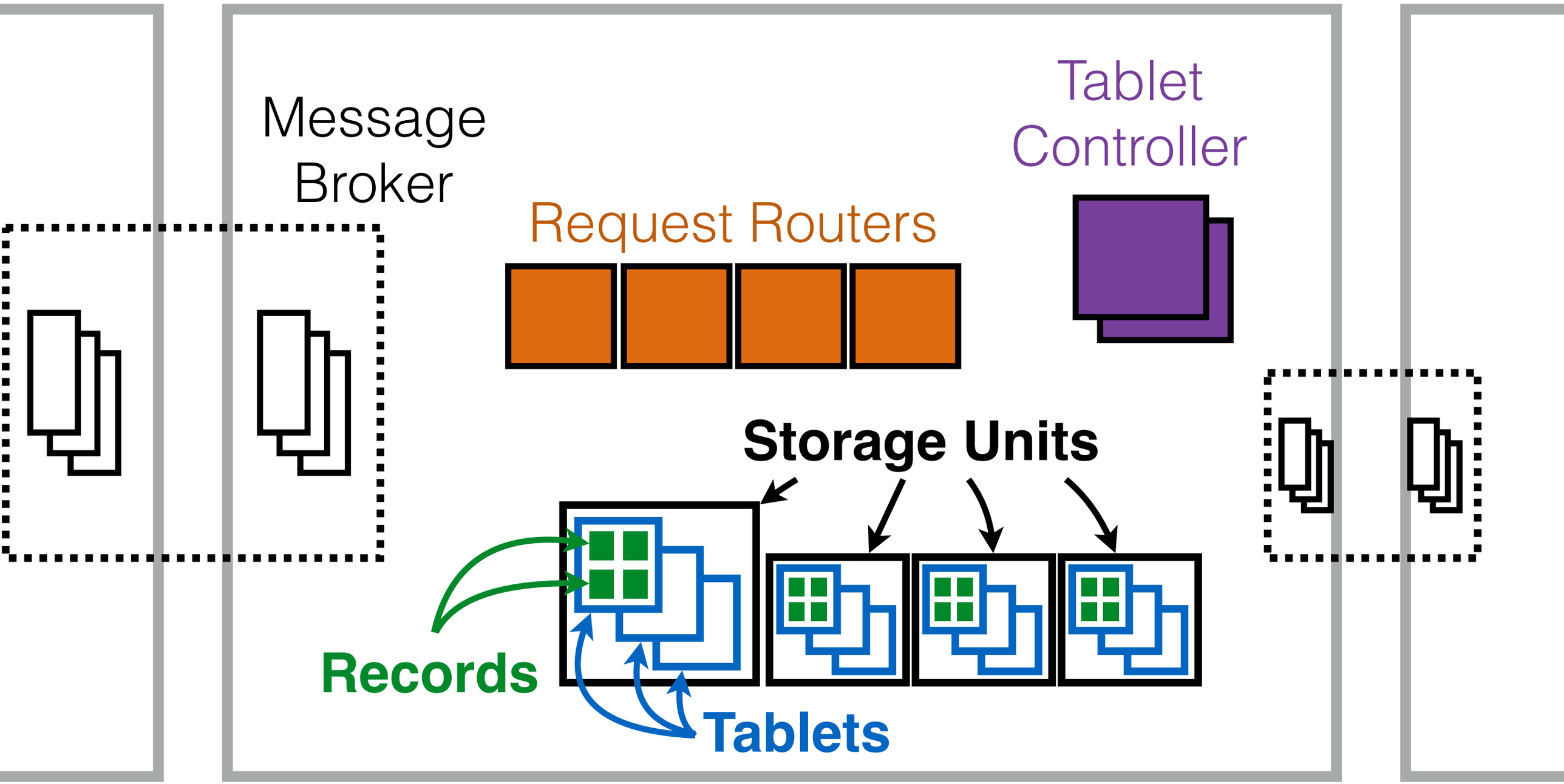
Core Design Ideas

- Publish/Subscribe guaranteed message delivery
- Record-Level Mastering
- No change log, No Archive
- Support single entry requests and range queries

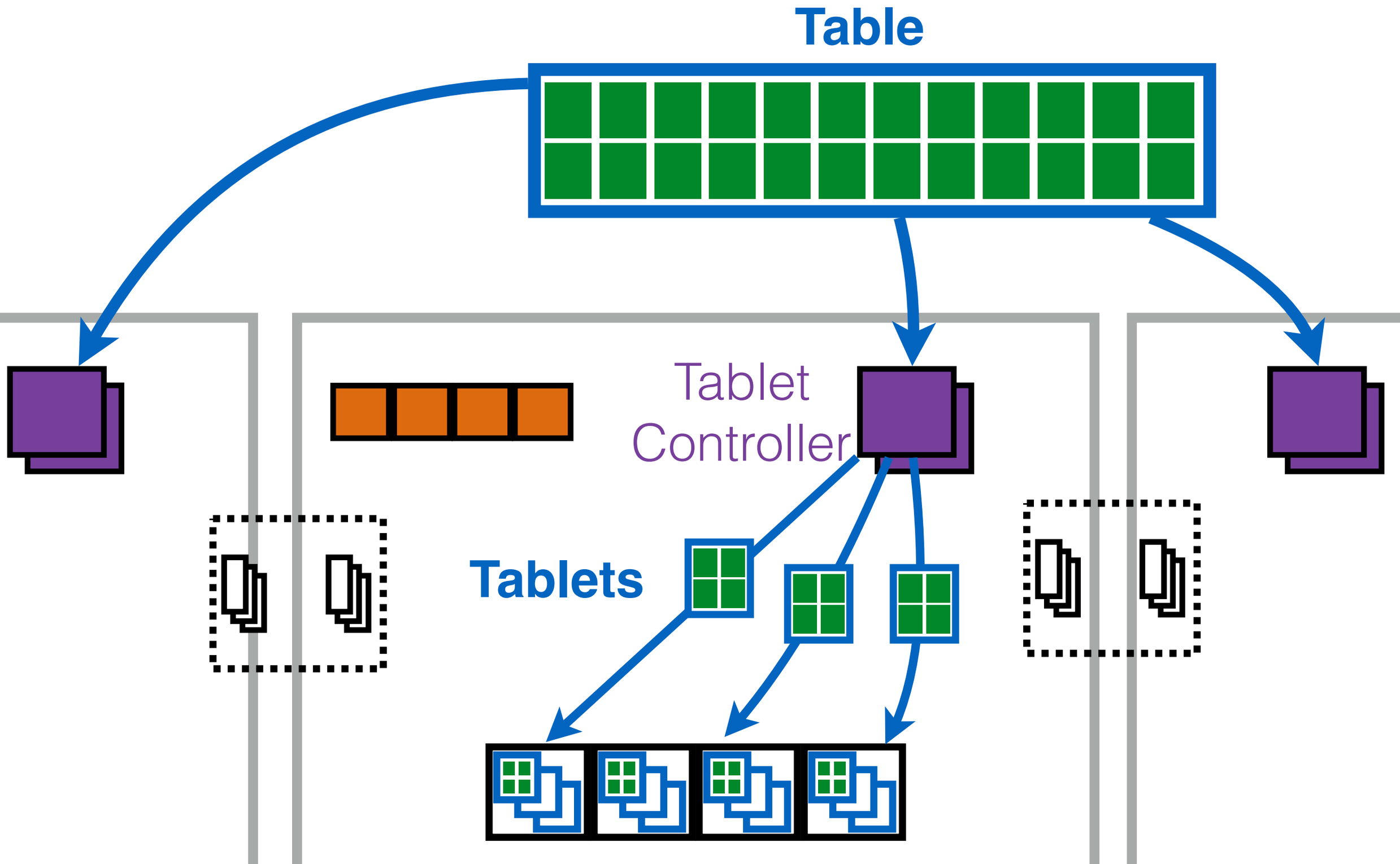
Overall Design



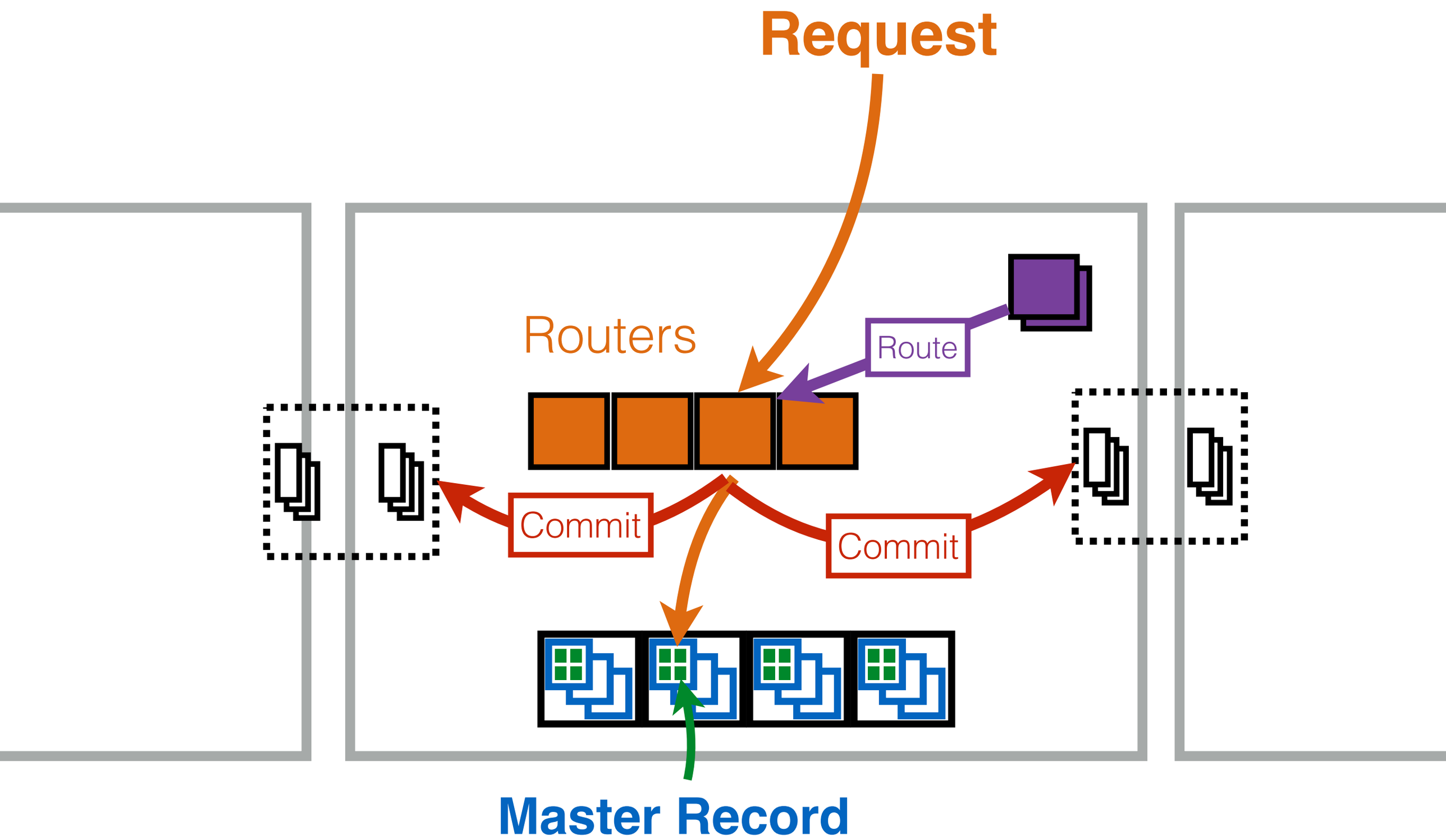
Data Storage



Data Sharding

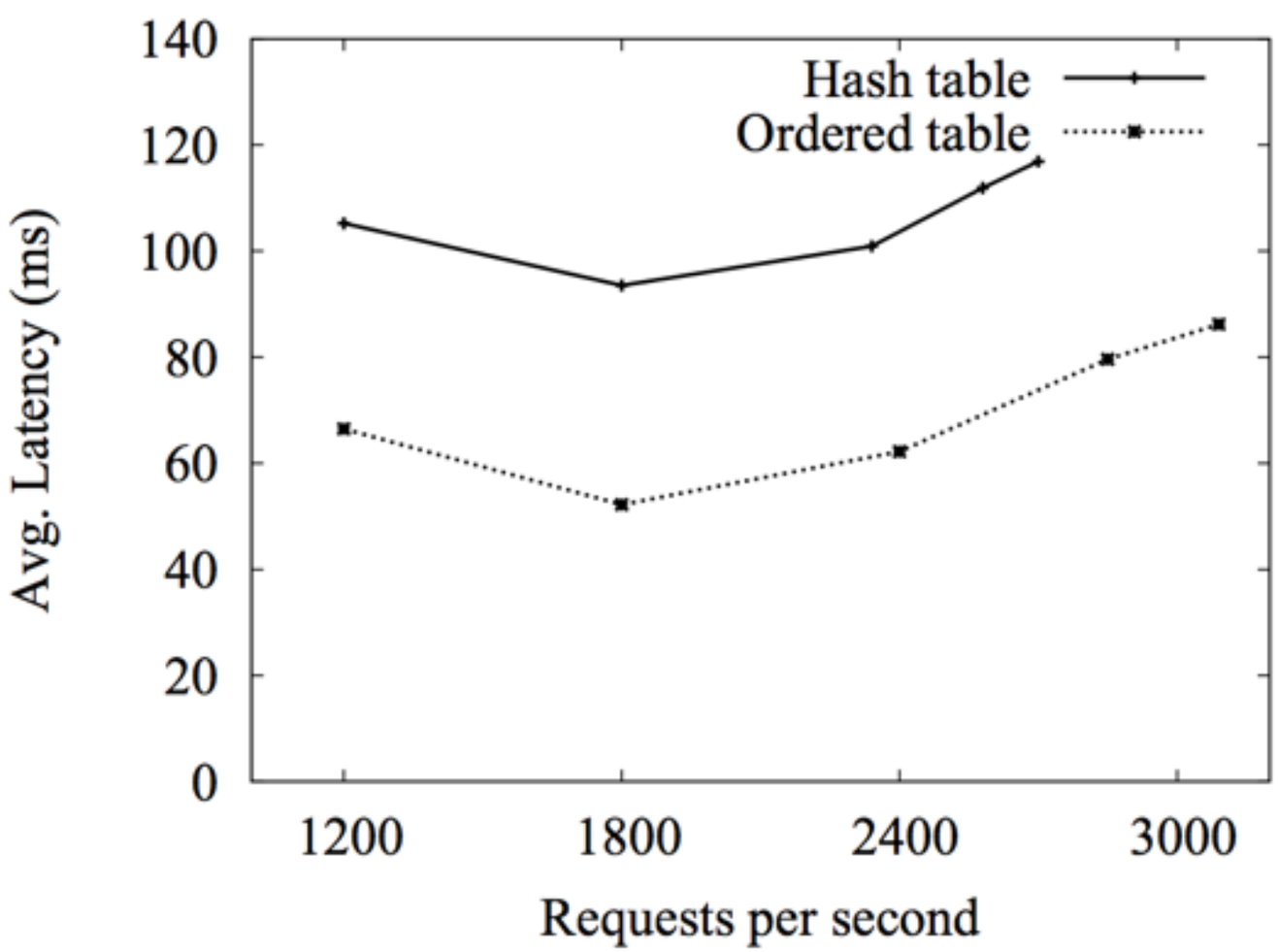


Request Routing



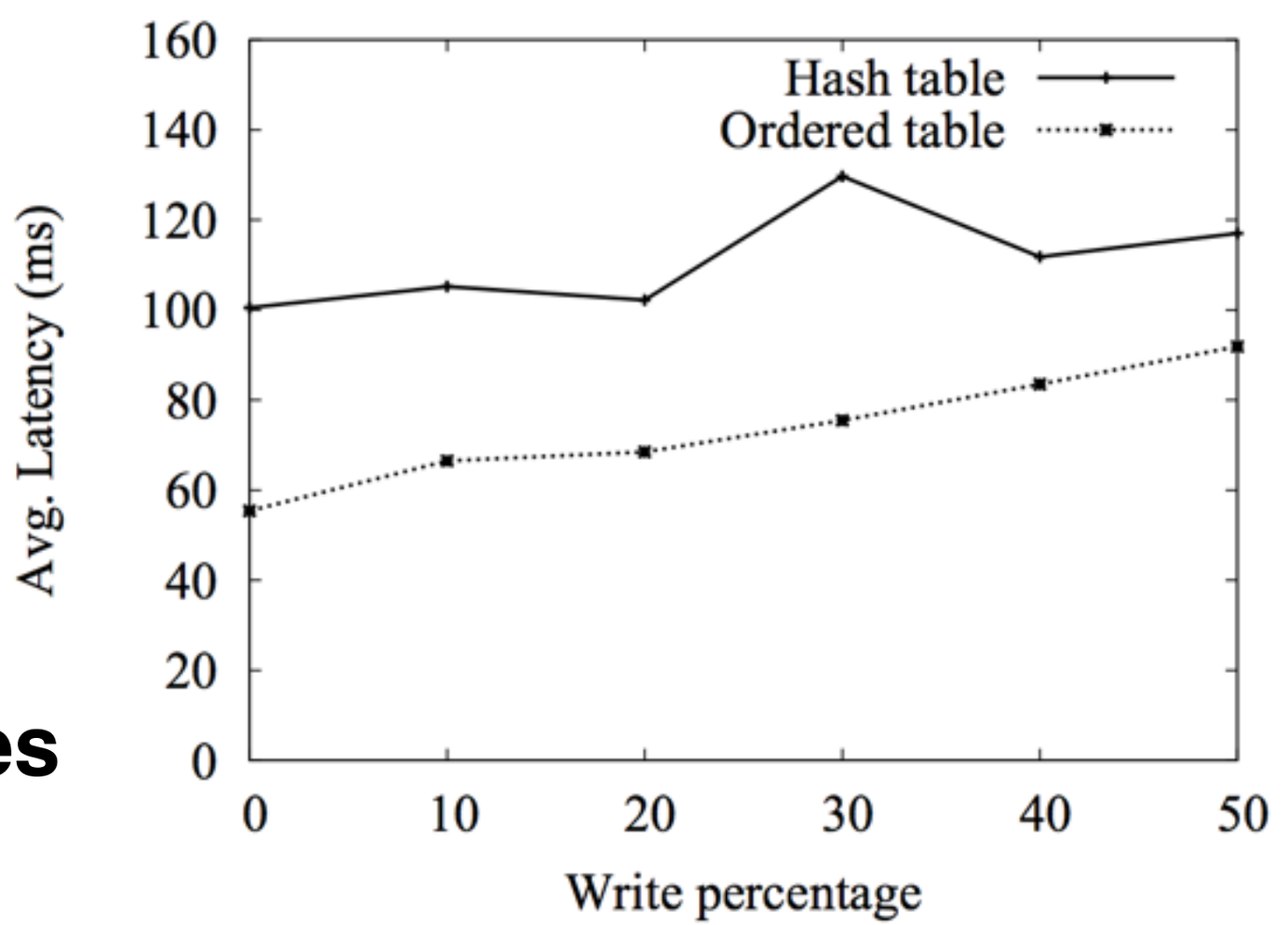
Evaluation Setup

- Three Regions (two West coast, one East)
- MySQL with InnoDB for storage units
- 1 kb records, 128 tablets, 100 clients per region
- Varied Read/Write mix, Distribution Skew, Locality

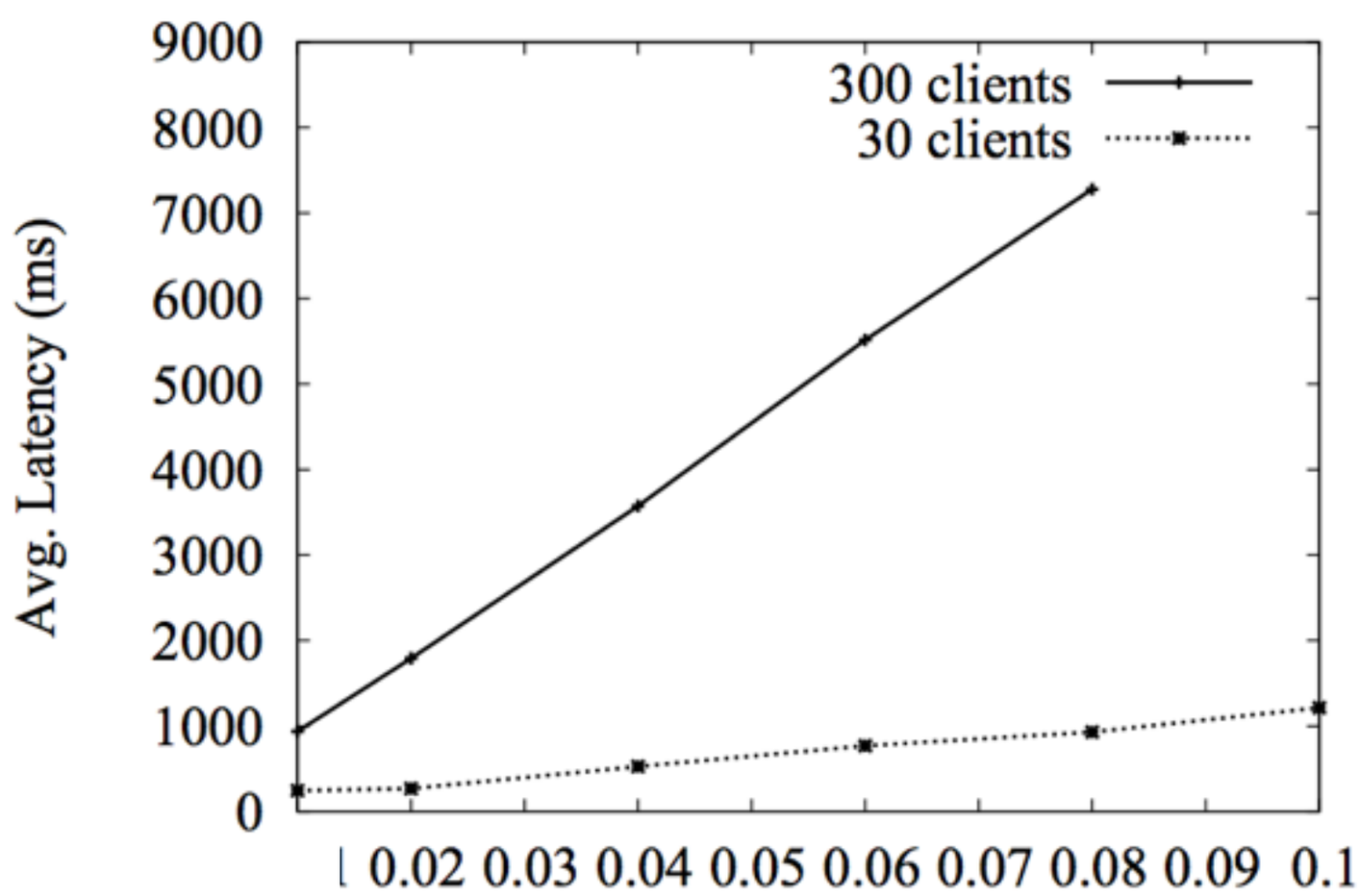


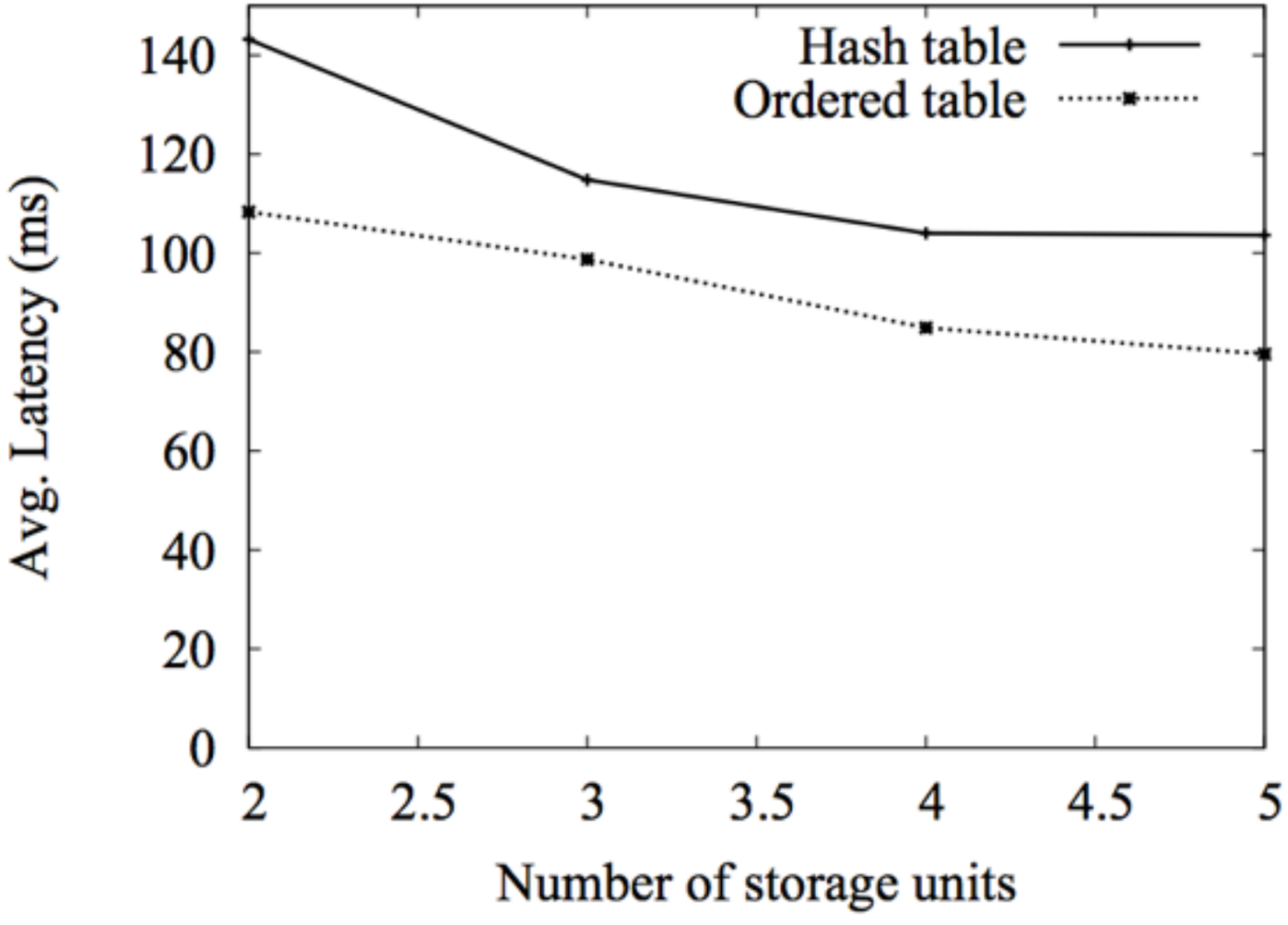
Varying Request Rates

Varying Read/Write Mixes



Varying Range Scan Lengths





Varying Storage Unit Counts

Future Work

- System does not account for varying load on Tablets
- Referential integrity
- Ad-hoc queries
- Bundled updates
- Relaxed consistency
- Have customers share Message Brokers and Storage Units

Opinion

- Strengths - Seems simple enough
- Weaknesses
    - Remote master copy causes slow ops
    - Exposes version numbers to apps
    - Can only range search over Primary Keys
- Unanswered Questions - Doesn’t evaluate the variety of read/writes, and all the future work