Overview of Postgres-XC

Koichi Suzuki
Postgres-XC Development Group

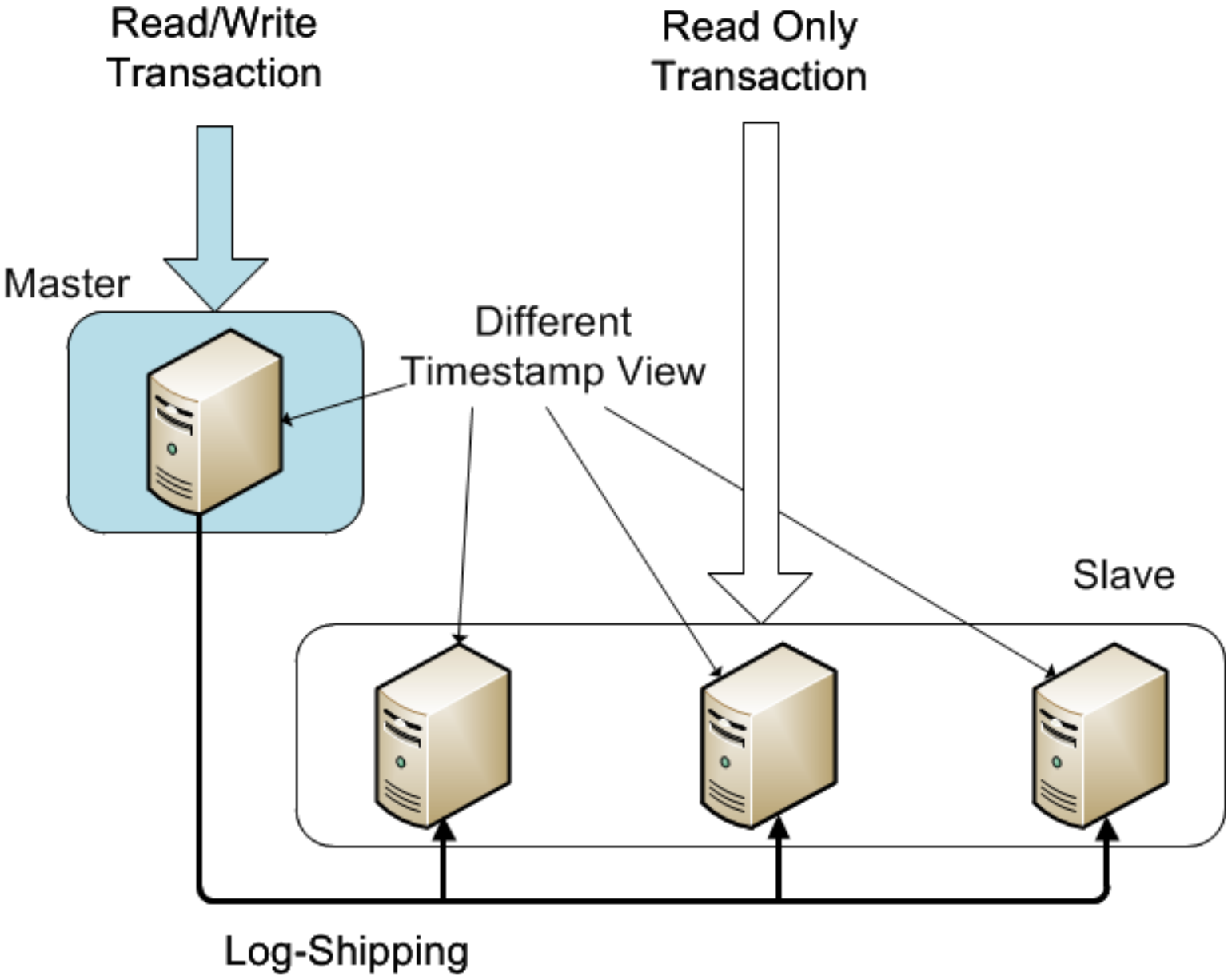
Postgres Conference China 2014
December 12th, 2014
Shenzhen, China

NTT DATA

What is Postgres-XC

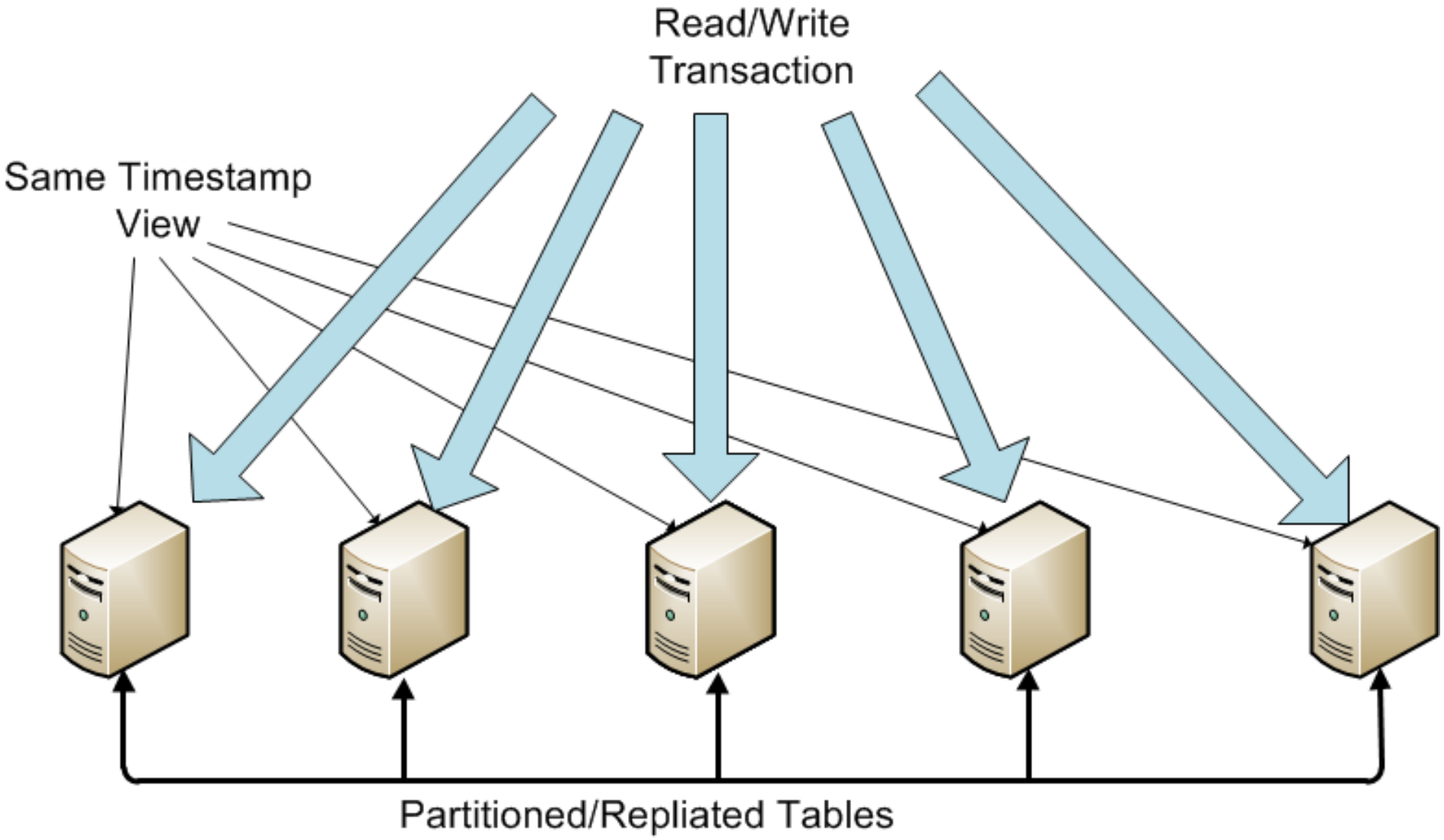
- Symmetric PostgerSQL cluster
  - No master/slave replication
  - No read-only clusters
  - Every node can issue both read/write
  - Every node provides single consistent database view
  - Transparent transaction management
- Not just a replication
  - Each table can be replicated/distributed by sharding
  - Parallel transaction/query execution
    - So both read/write scalability

Master/Slave with Log Shipping



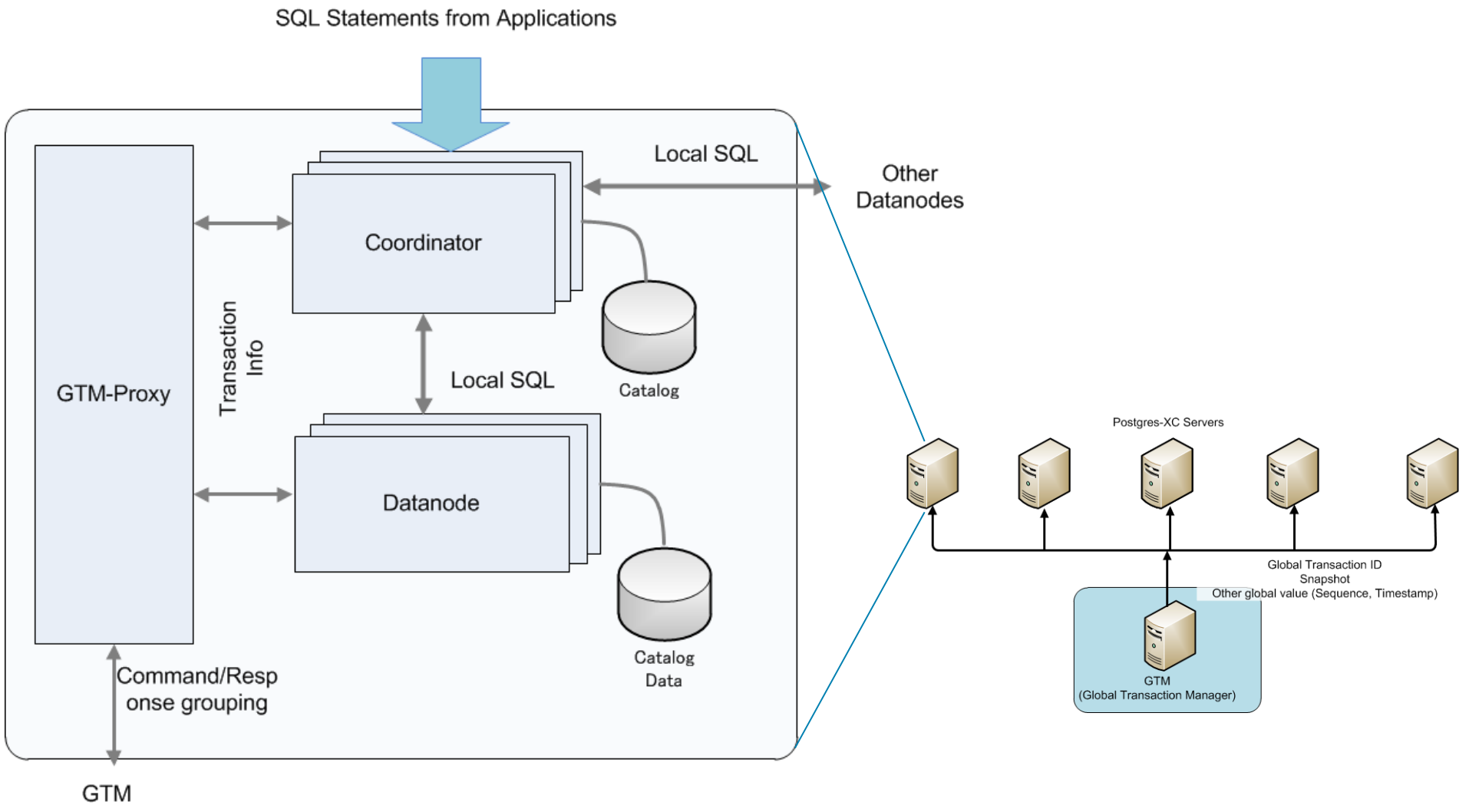
12th Dec., 2014

Postgres-XC Symmetric Cluster



12th Dec., 2014

Architecture and Configuration



12th Dec., 2014

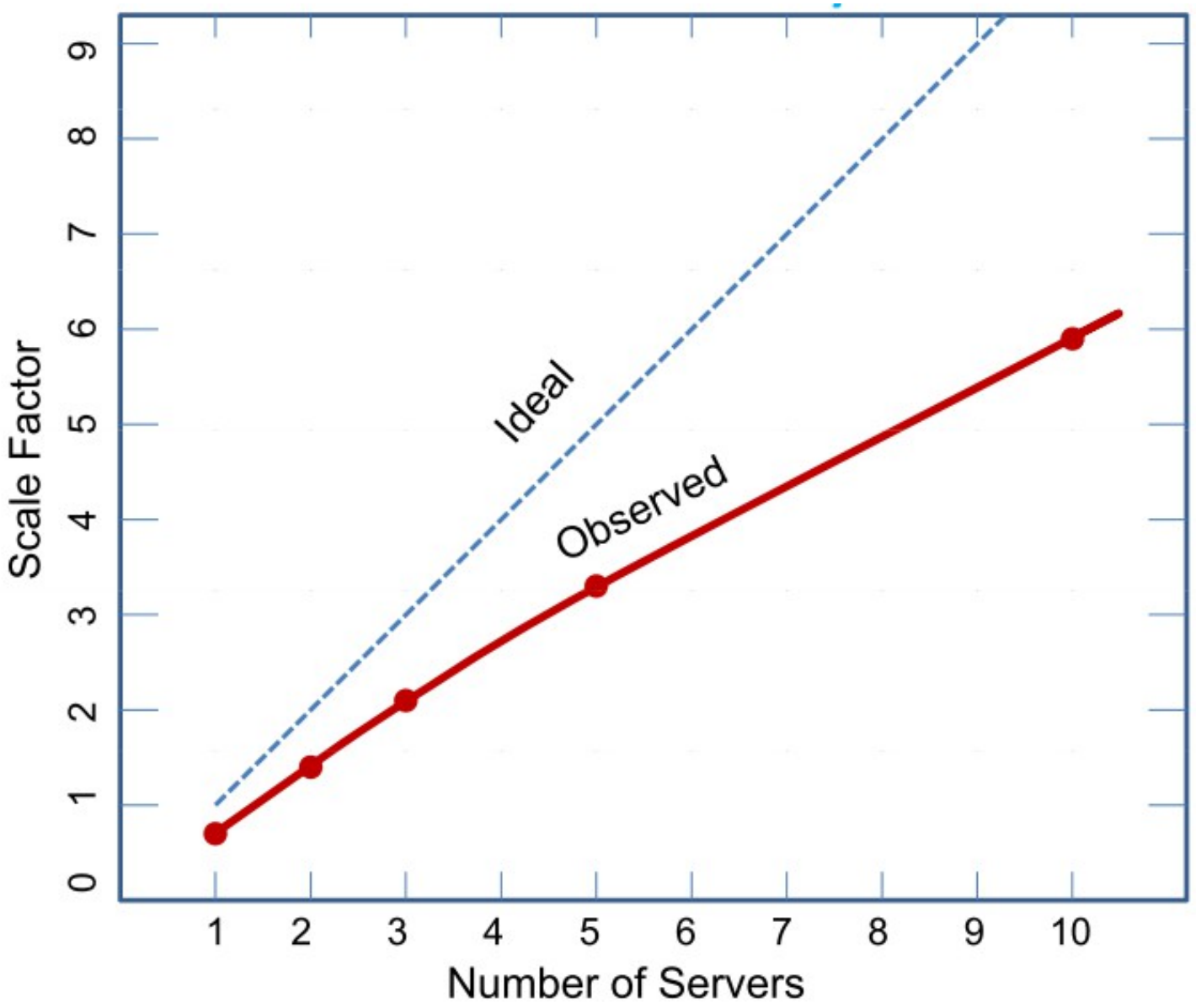
Outline of Components

- GTM (Global Transaction Manager)
  - Distributed MVCC
    - Provide global transaction ID (GXID) to all the transactions
    - Provide global snapshot to all the transactions
  - Sequence
- GTM_Proxy
  - Group communications to GTM and reduce amount of GTM network workload
- Coordinator
  - Handles incoming SQL statements
    - Parse, plan, conduct execution in datanodes and the coordinator.
    - Integrate local results from each datanode involved.
- Datanode
  - Actual data storage
    - Almost vanilla PostgreSQL

Share the binary

12th Dec., 2014

Scalability



DBT-1 (Rev)

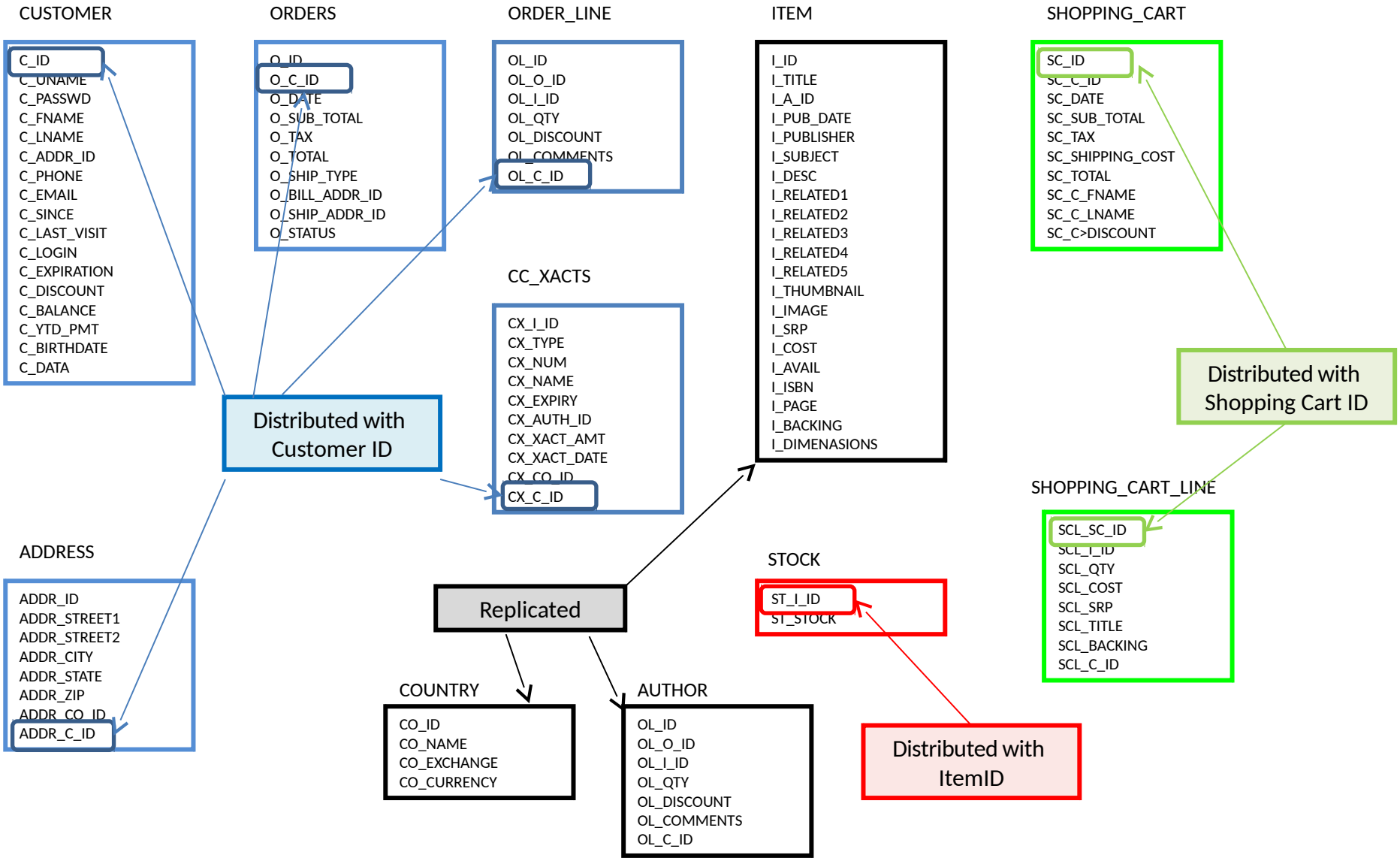
12th Dec., 2014

Scaling-out architecture

- Postgres-XC is designed for throughput scale-out.
- Additional duration for coordinator – datanode communication required.
- Can handle more connection/transaction in parallel.
- For data load, you separate the data to be fed from different coordinators.
  - Key for data load scale-out.
  - Data load from single coordinator may not scale.
    - XC scales up when multiple coordinators/datanodes are involved.

Combining sharding and replication

DBT-1 example



12th Dec., 2014

How XC Scales both Read/Write

- Transaction Tables → Sharding
  - Only one write
  - Parallel writes in datanodes
- Master Tables → Replication
  - Relatively static: Not significant many-writes overhead
  - Local join with transaction tables → Most join operation ca be done locally in datanodes

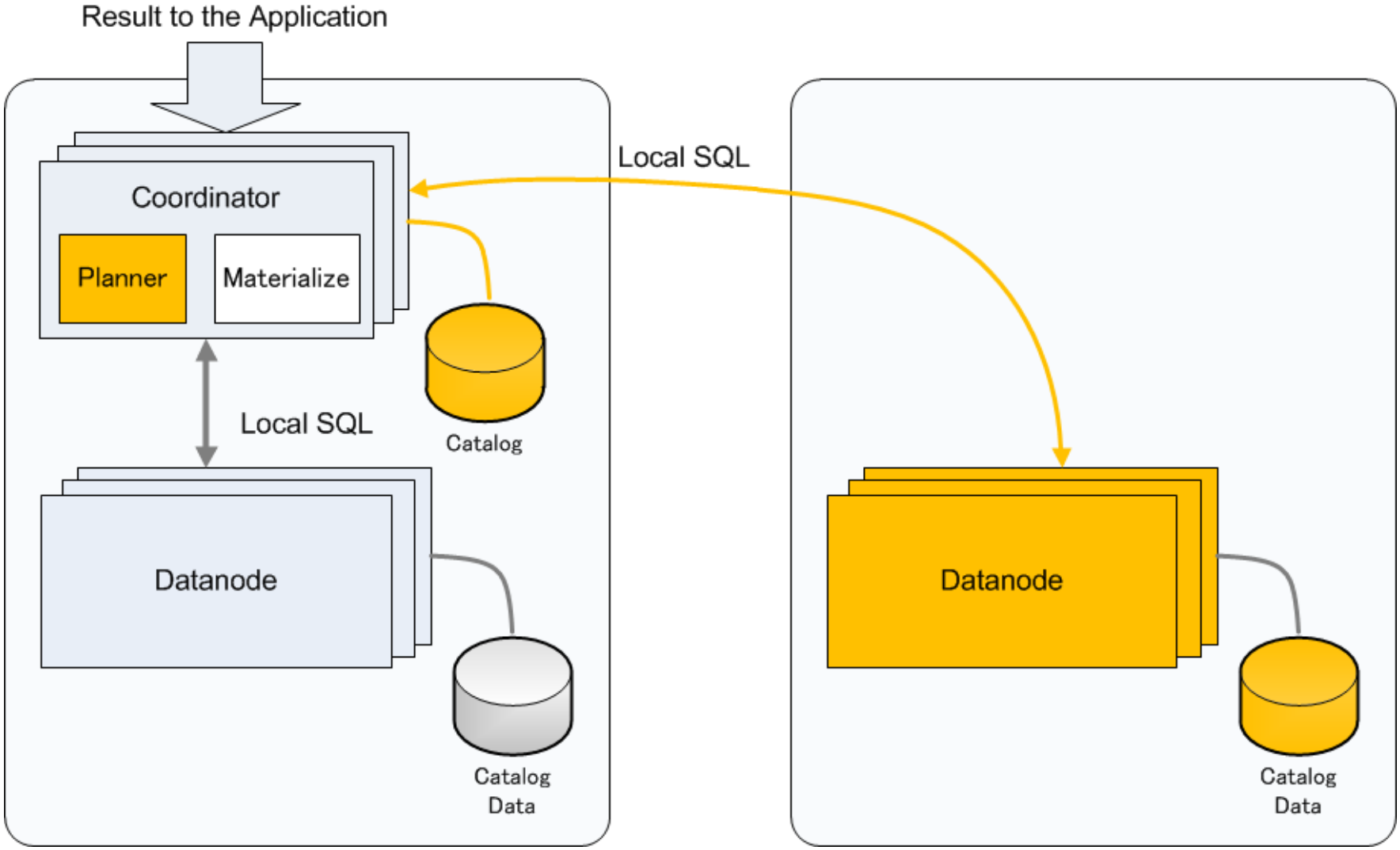
Postgres-XC API

- Binary compatible with PostgreSQL
  - Limited support for ODBC
  - JDBC may have a couple of restrictions
- Compatible statements to PostgreSQL
  - Slight difference
    - CREATE TABLE, etc.
    - Constraints can be enforced only locally in each datanodes.
  - Extra
    - Coordinator/datanode membership management, etc.
      - CREATE/ALTER/DROP NODE, EXECUTE DIRECT...
    - Extension in aggregates
      - Combiner functions
    - Maintain consistency in point-in-time-recovery
      - CREATE BARRIER
- No load balancing so far
- You should notice
  - OID is local to each node

12th Dec., 2014

Join Example (1)

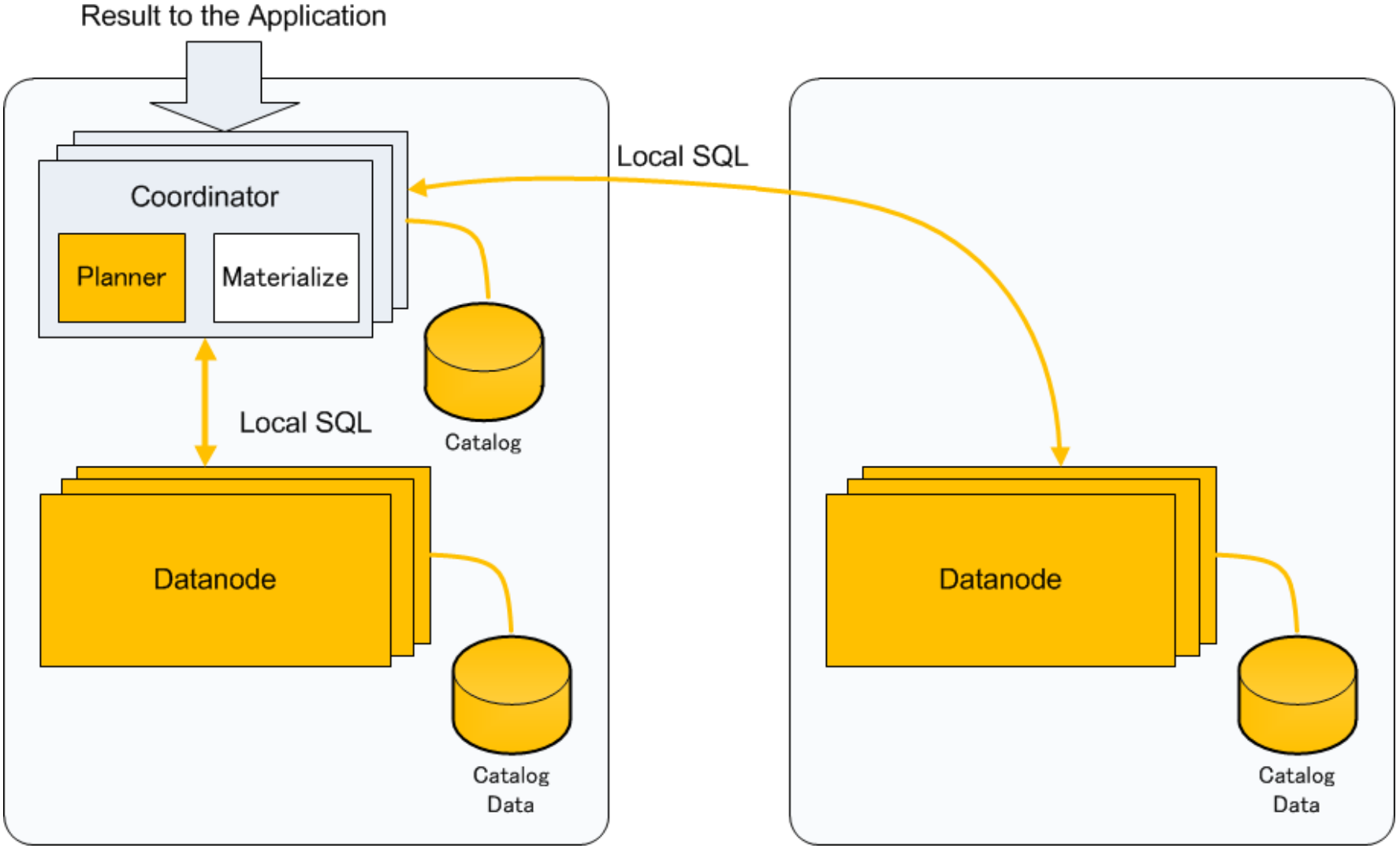
- Replicated Table and Partitioned Table
  - Can determine which datanode to go from WHERE clause



12th Dec., 2014

Join Example (2)

- Replicated Table and Partitioned Table
  - Cannot determine which datanode to go



12th Dec., 2014

Thank You Very Much!

koichi@dbms.gmail.com