PGConfChina 2012

Postgres-XC, write-scalable PostgreSQL Cluster
PostgreSQL usage in NTT group

June 16th, 2012

Koichi Suzuki
NTT DATA Intellilink Corporation

Agenda

NTT DATA

- Postgres-XC
  - What is Postgres-XC
  - Core architecture overview
  - Statement Handling
  - High-availability
  - Release Status
- PostgreSQL in NTT
  - Open source software center
  - Effort to improve PostgreSQL
  - Application Characteristics and Deployment

Self Introduction

NTT DATA

- Postgres-XC leader and core architect, as well as a core developer
  - Whole architecture design
    - Global transaction management and data distribution as a key for write-scalability
- Work for NTT DATA Intellilink
  - Subsidiary of NTT DATA corporation dedicated for system platform
  - Member of NTT group company
  - Providing open-source engineering for NTT Open Source Software Center
- Resources
  - koichi.clarinet@gmail.com (facebook, linkedin)
  - @koichiclarinet (twitter)

Postgres-XC Highlights

NTT DATA

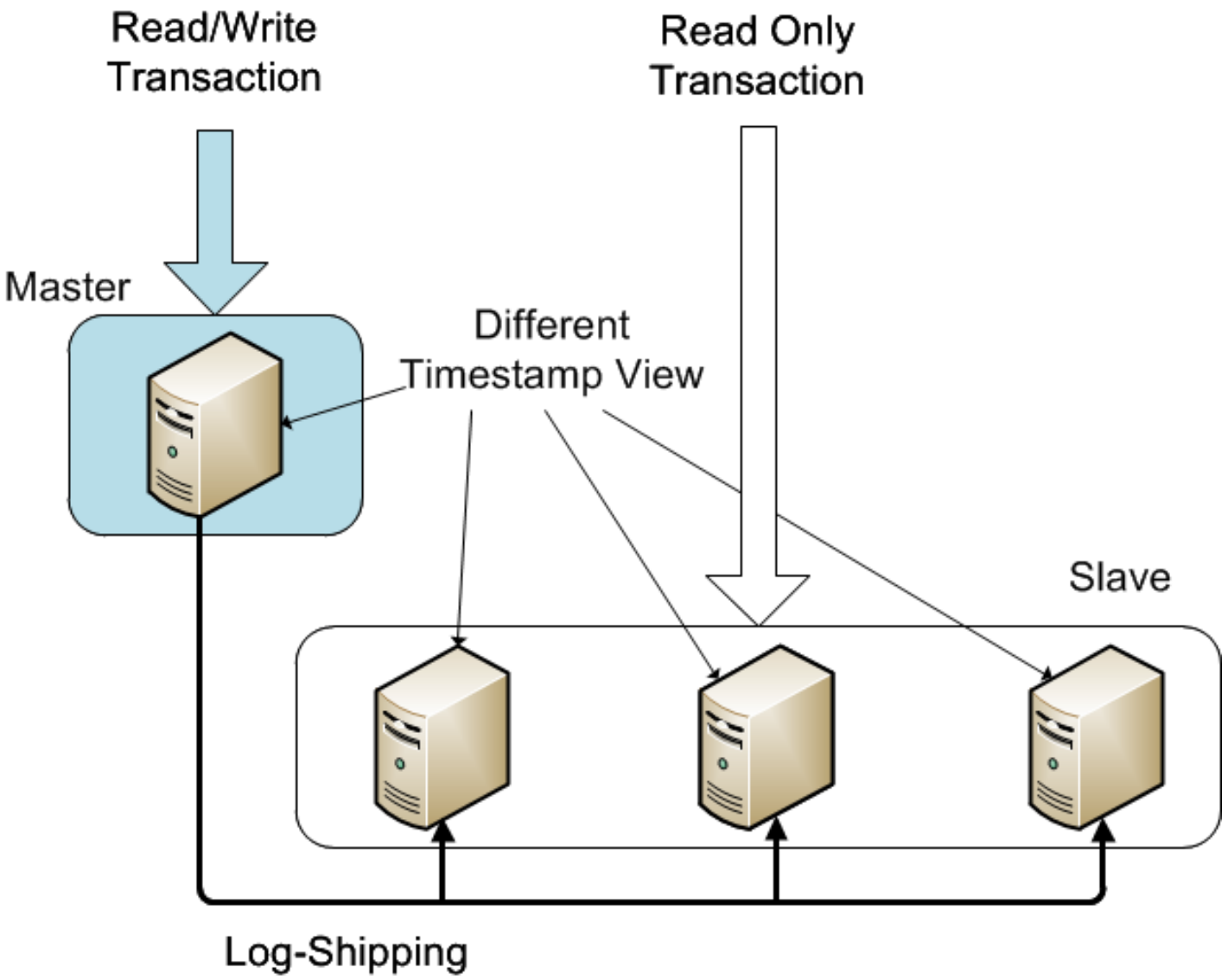
- Cluster software focused on write-scalability
- Based on PostgreSQL (at present, 9.1.4)
  - World's most advanced open source database
  - PostgreSQL license
- Same client APIs as PostgreSQL
  - Ease of application migration from existing PostgreSQL deployment
  - Same drivers, same front end, same SQL queries
- Licensing
  - PostgreSQL license (more or less BSD)
  - Free to use, modify and redistribute for commercial purposes

Postgres-XC History

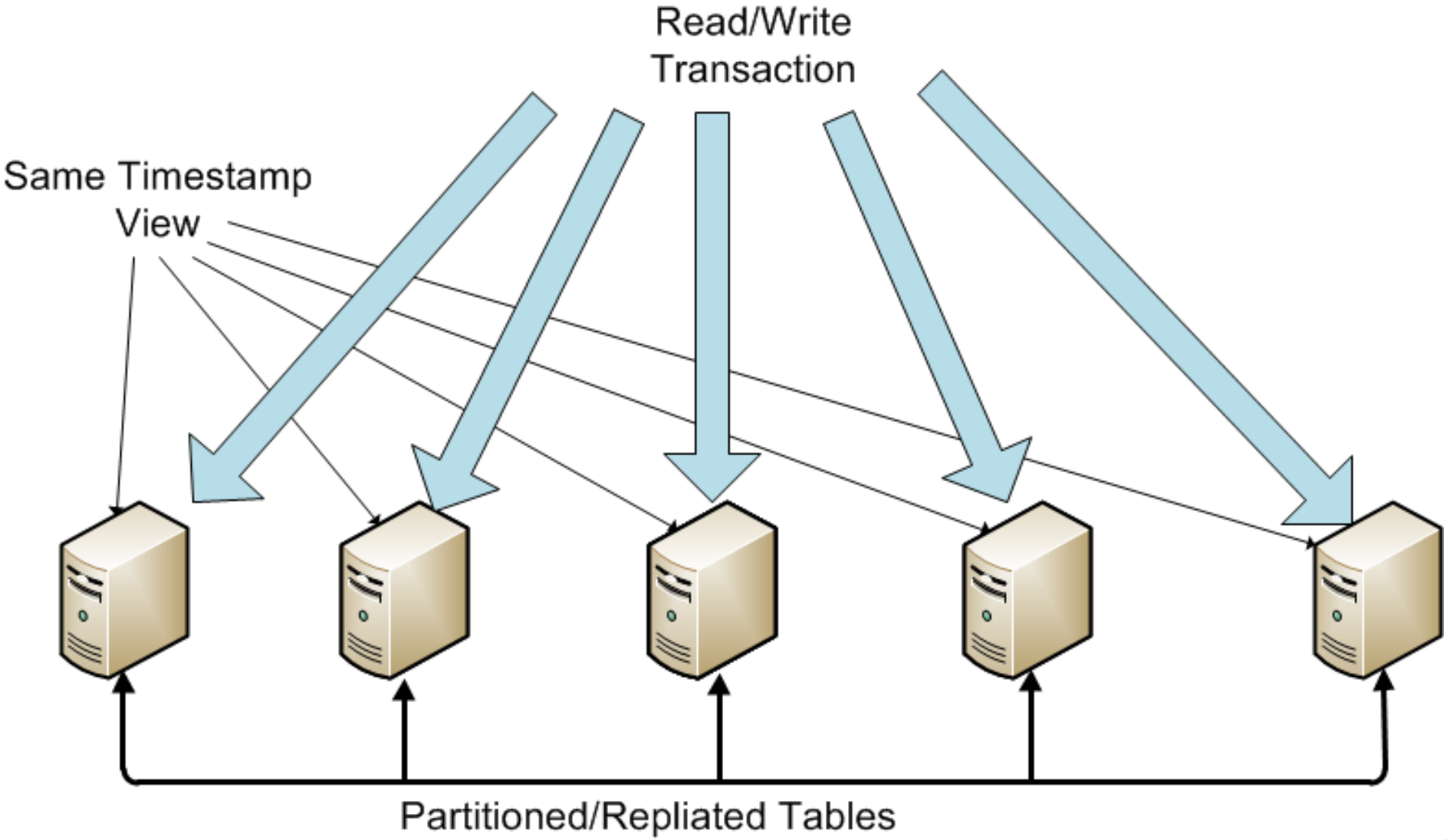
NTT DATA

- Work begun in 2003 as research project
  - NTT DATA, NTT DATA Intellilink
  - Co-worked with ISCAS (Beijing) at the final stage of research
- Restarted at 2009 as development work
  - NTT Open Source Software Center and EnterpriseDB
  - Goal to build a PostgreSQL based clustering solution which can serve as an alternative to Oracle RAC
  - Development is community-based, with resources gathered from NTT and EnterpriseDB
  - Licensing terms changed from GPL to PostgreSQL license (same as Postgres) in 2011

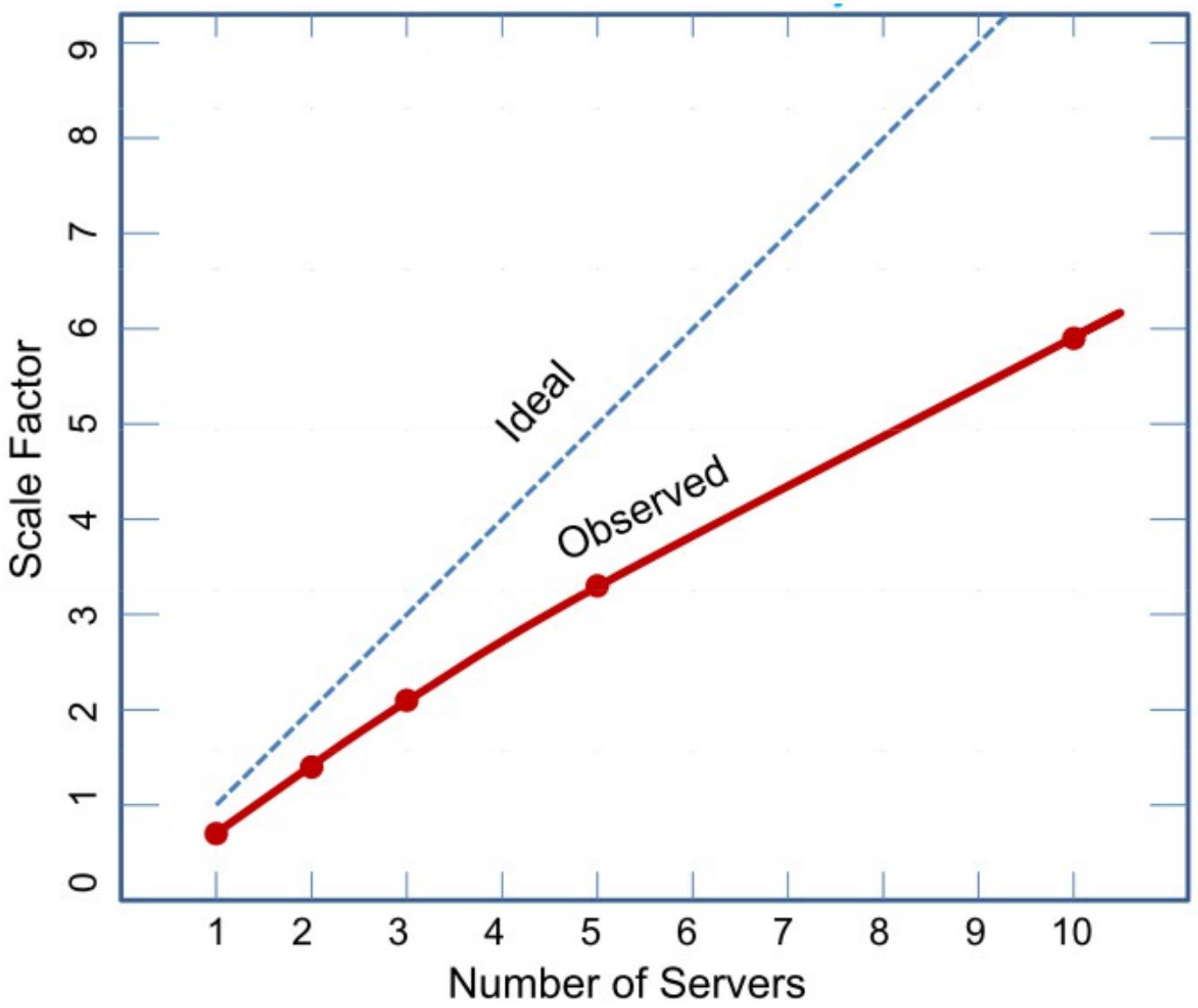
PostgreSQL Master/Slave with Log Shipping



Postgres-XC Symmetric Cluster



Scalability



Somewhat old. Latest measurement will be published elsewhere. So far, no different in the scalability.

DBT-1 (Rev)

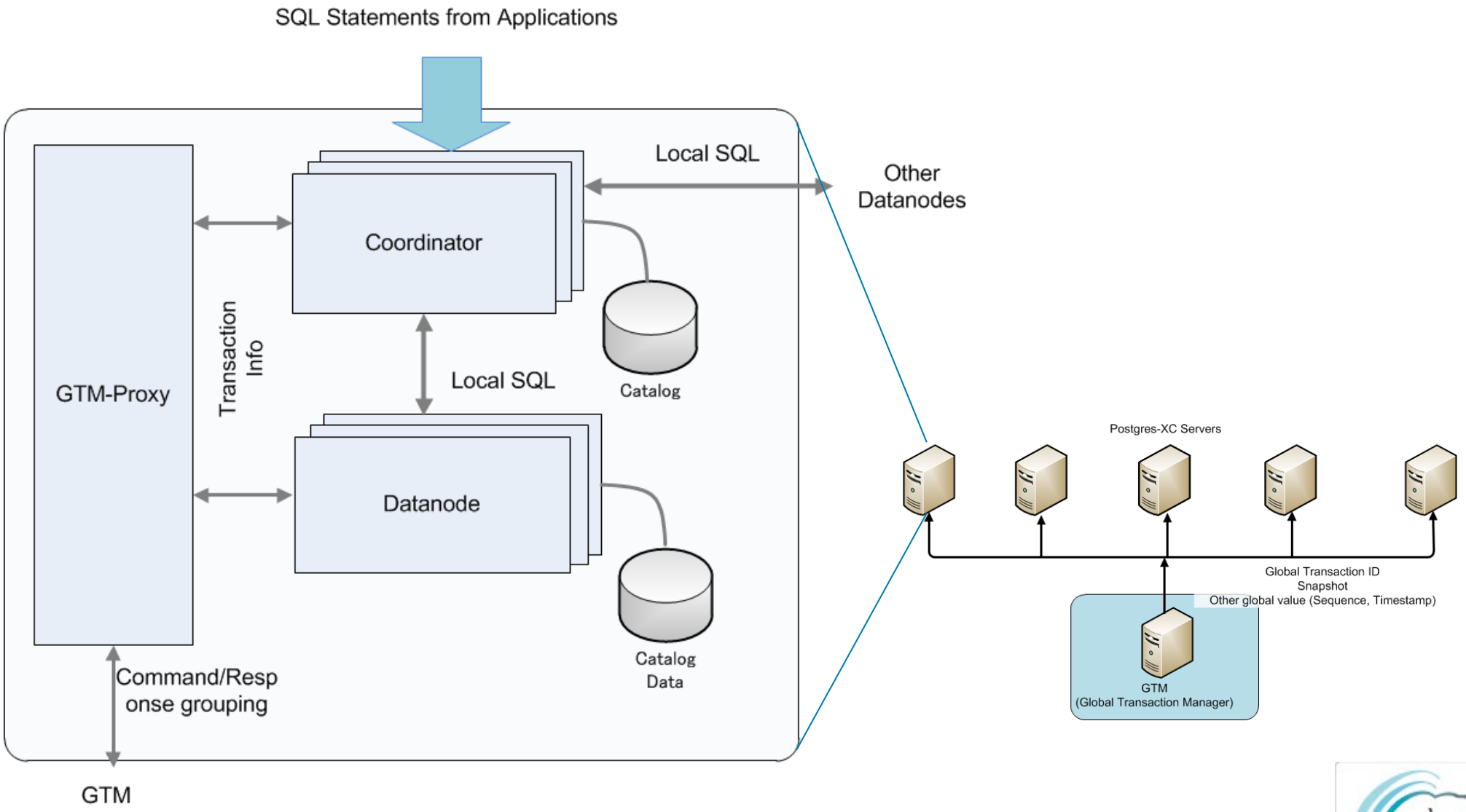
Current Status

NTT DATA

- Version 1.0 was released
  - Supports 64bit Linux
    - Tested under Cent OS 5.3 and ubuntu 10.4
    - Community members are testing in other platform
      - Open BSD
      - MacOS
  - Some restriction though
    - Trigger
    - WHERE CURRENT OF
    - Savepoint …
  - PostgreSQL license

Architecture Overview

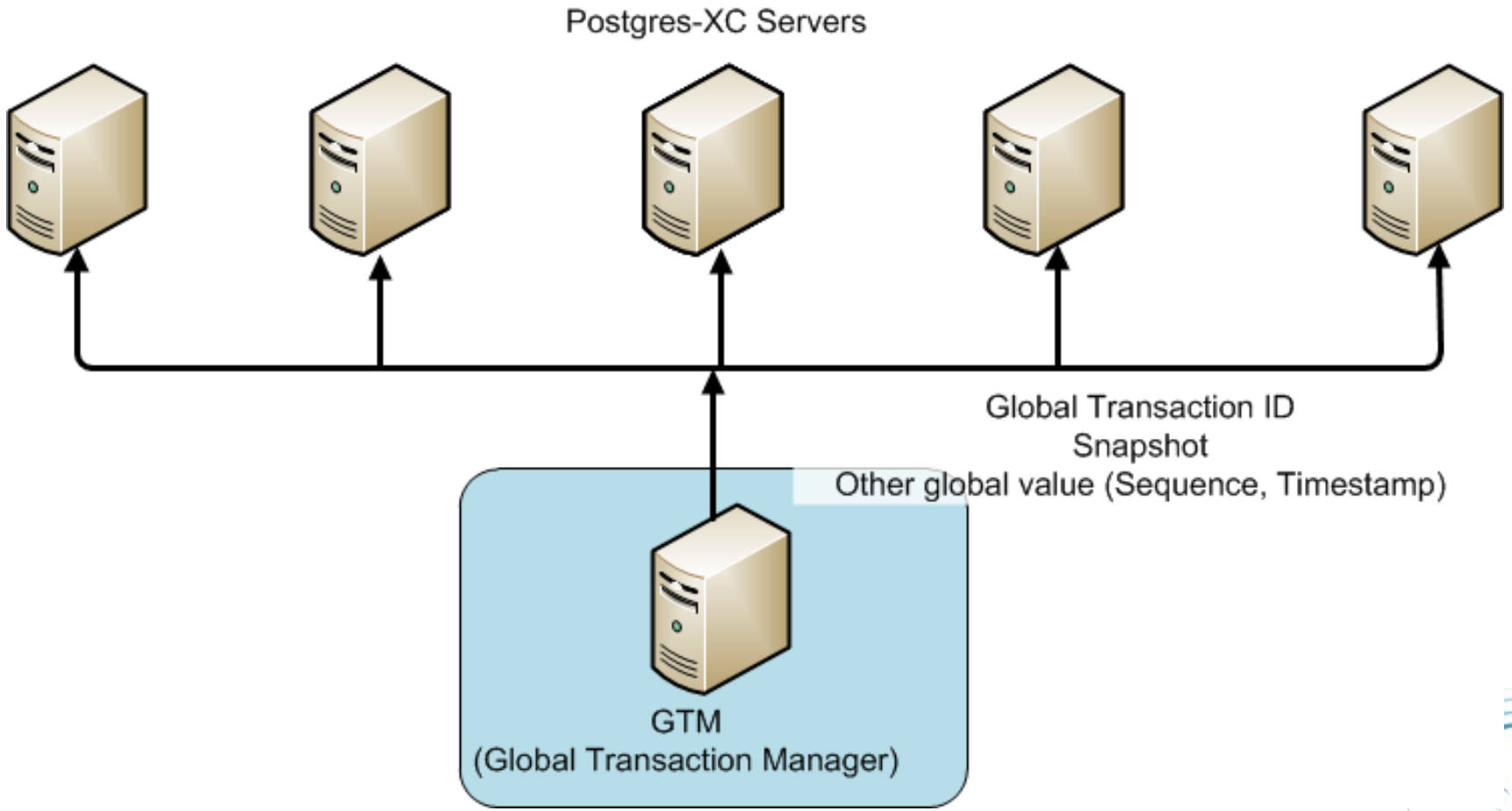
Server Configuration and GTM-Proxy



GTM, coordinator and datanode are key components.

GTM: Key for Transaction Transparency

- Consistent Transaction ID (GXID) throughout the system
- Provide global snapshot for consistent visibility from any server



Coordinator: Global SQL processor

NTT DATA

- Point of contact for the application/client
- Management of remote node data
  - Parse and partially plan the statements
  - Determine the data to be fetched from remote nodes at planning or execution
  - Fetch the required data by issuing queries to dedicated Datanodes
  - Combine and process the data to evaluate the results of the query (if needed)
  - Pass the results to the applications
- Manages two-phase commit
- Stores catalog data: cluster-related information
- Needs and manages space for materializing results from remote nodes
- Binary based on the latest PostgreSQL release

Datanode: actual data store

NTT DATA

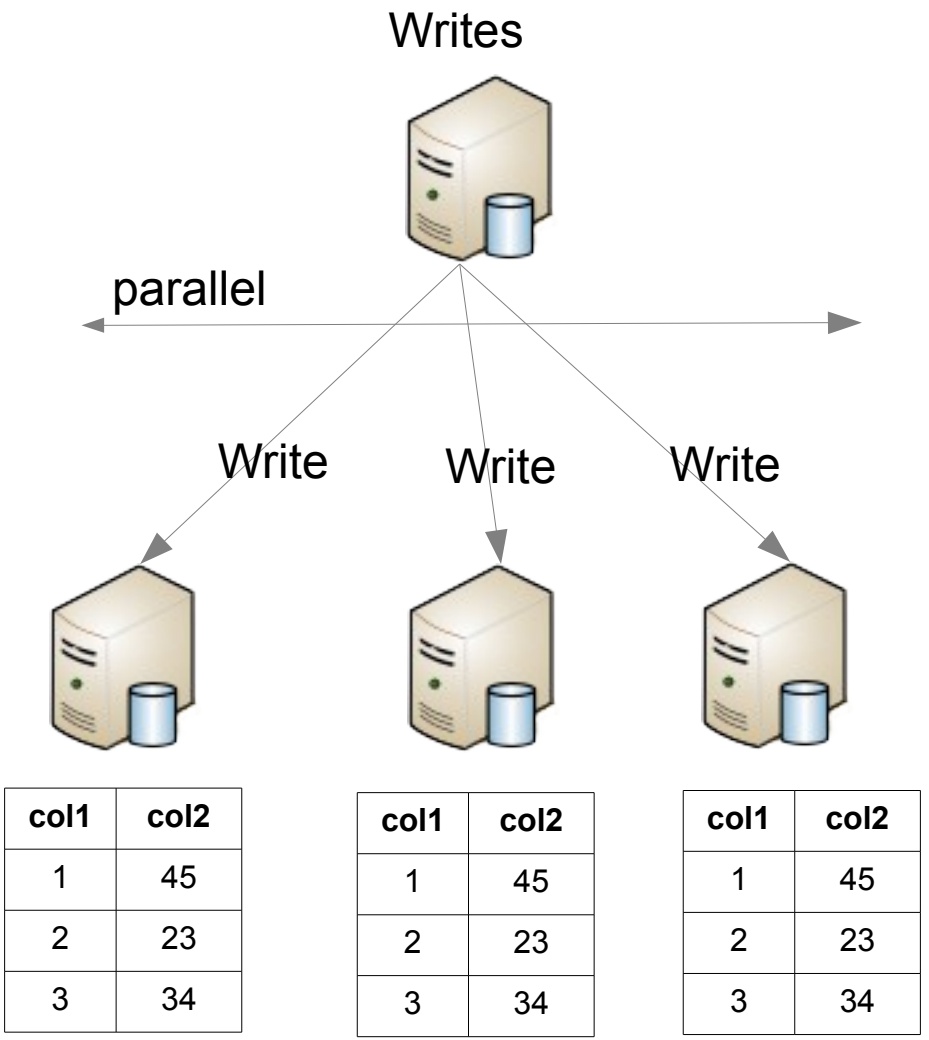
- More or less a PostgreSQL instance (remote node)
- Stores tables and catalogs
- Executes the queries from client Coordinator and returns results to it
- Data nodes can be made fault tolerant by Hot-Standby and Synchronous Replication technologies available of standard PostgreSQL
- Binary same as Coordinator, based on latest PostgreSQL release

Data Management

NTT DATA

- Table types
  - Replicated table
    - Each row replicated to Datanodes
    - Statement based replication
  - Distributed table
    - Each row of the table is stored on one datanode, decided by one of following strategies
      - Hash
      - Round Robin
      - Modulo
- Managed by SQL extensions (CREATE TABLE)
- Possible to define subset of nodes

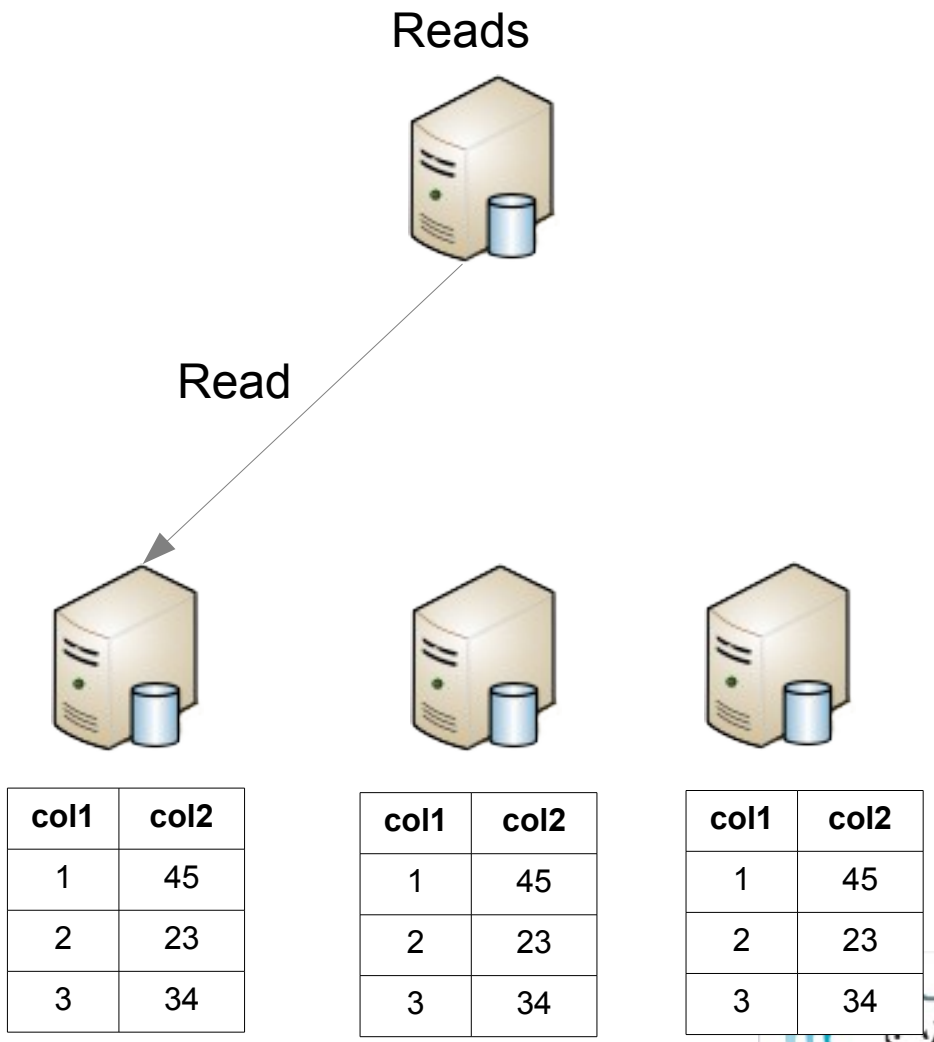
Replicated Tables



col1	col2
1	45
2	23
3	34

col1	col2
1	45
2	23
3	34

col1	col2
1	45
2	23
3	34

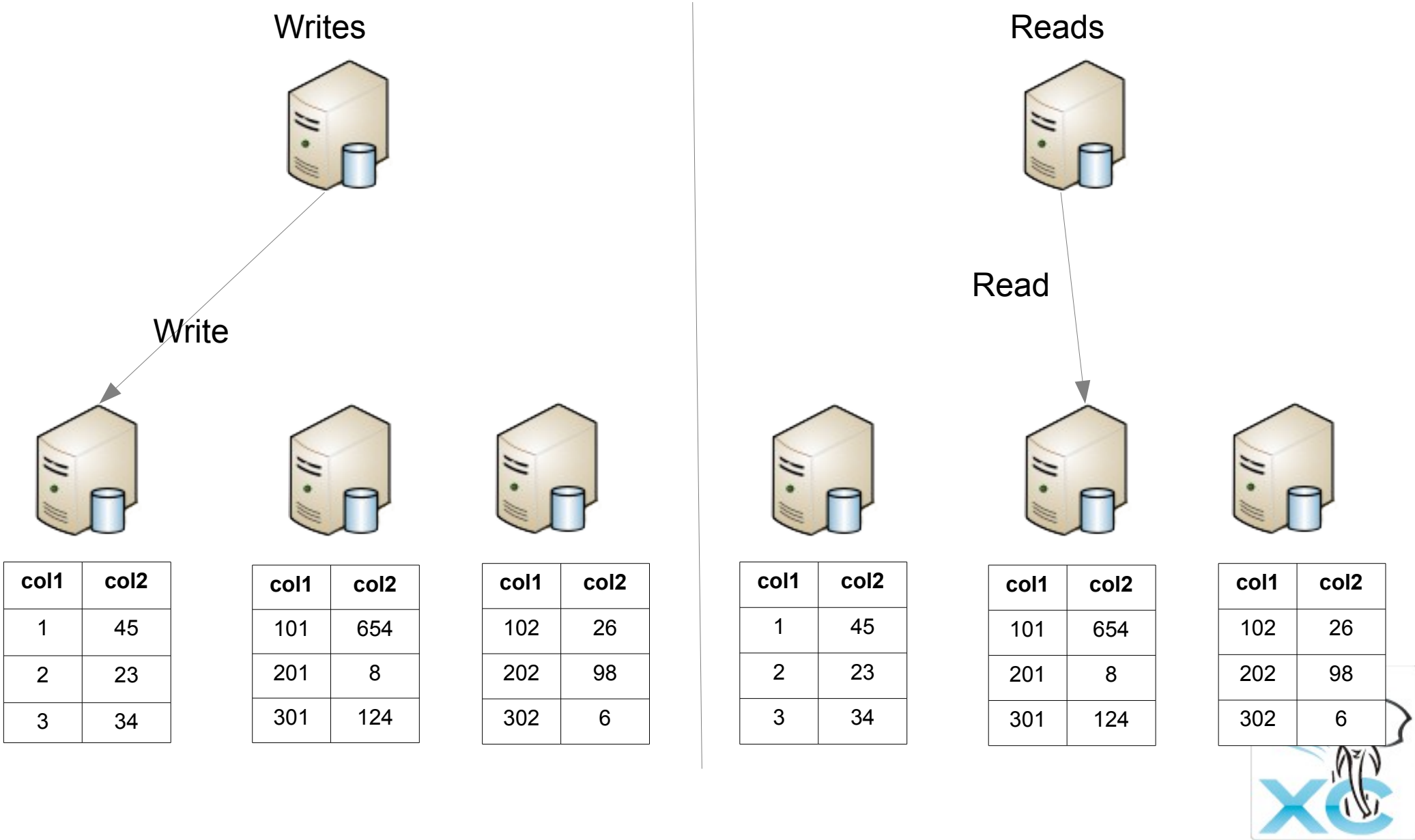


col1	col2
1	45
2	23
3	34

col1	col2
1	45
2	23
3	34

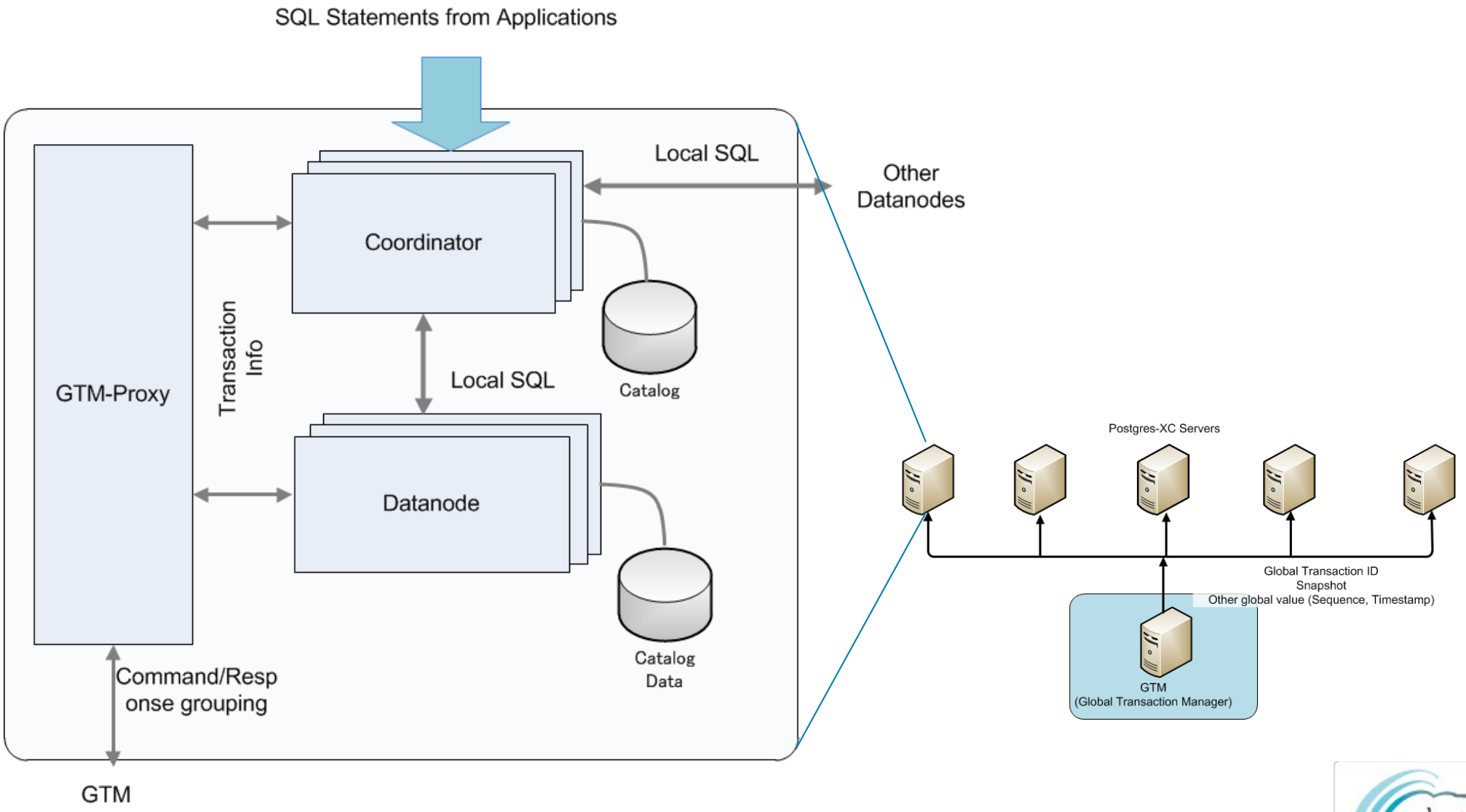
col1	col2
1	45
2	23
3	34

Distributed tables

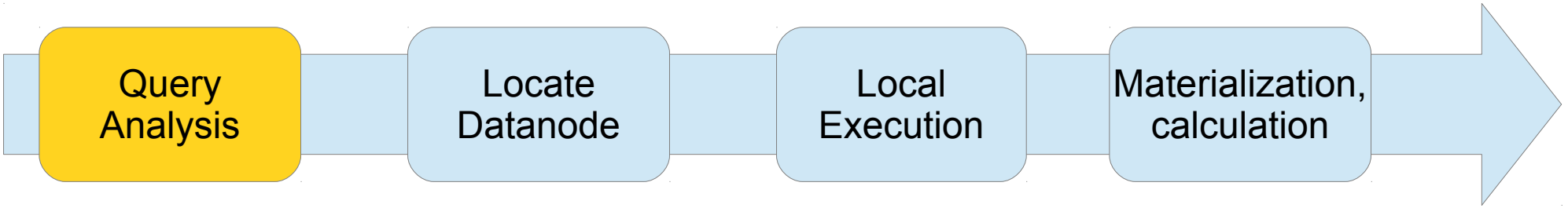


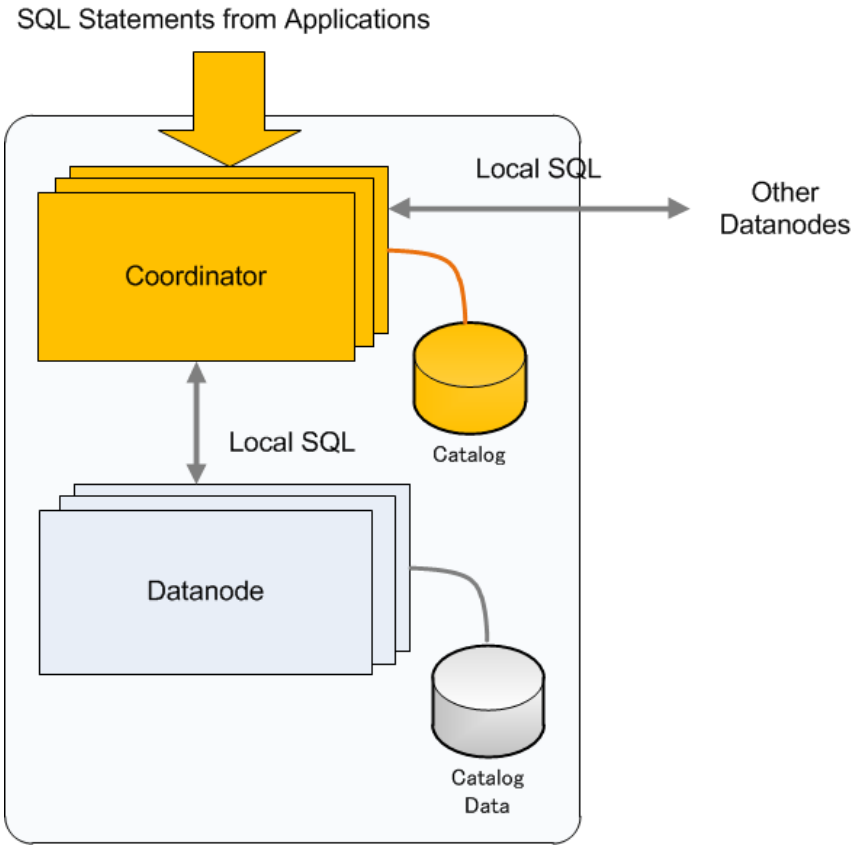
Statement Handling

Node configuration, again



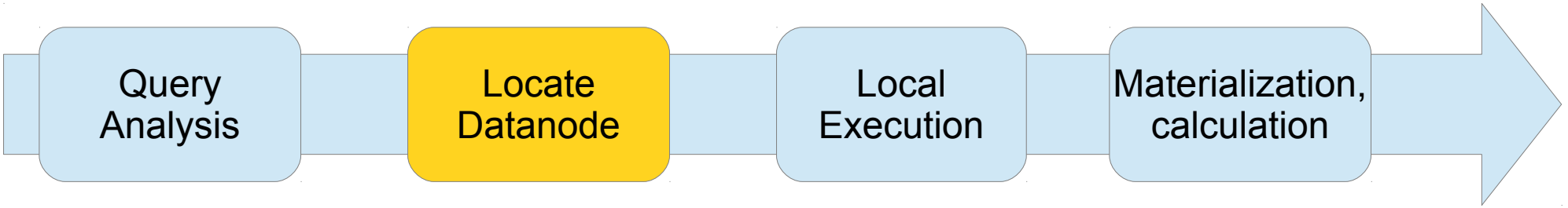
Query processing (1)

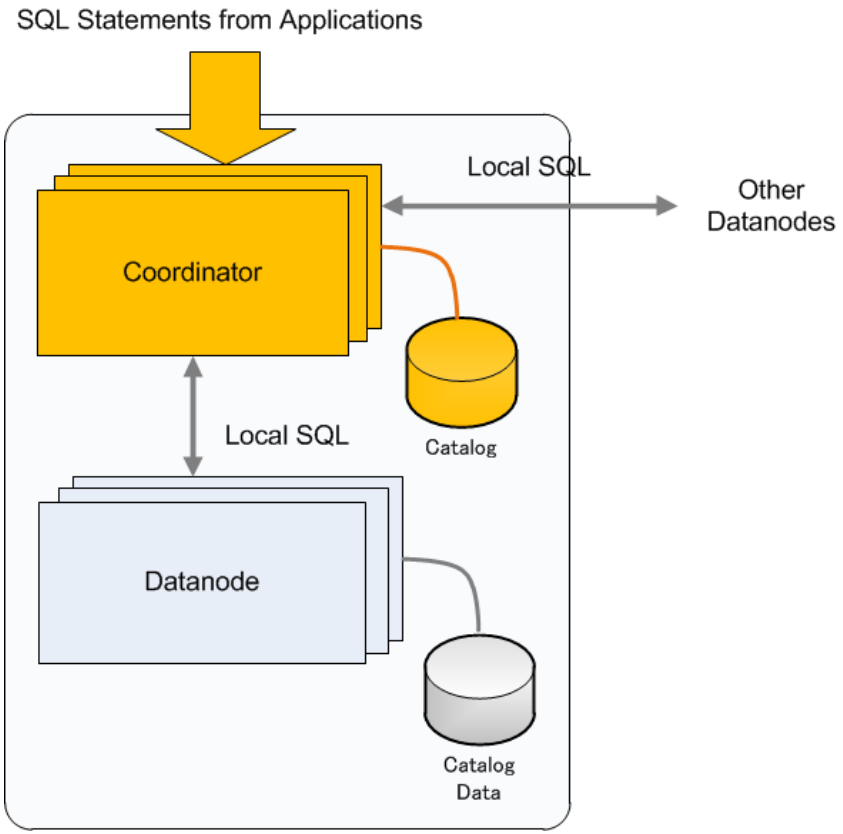




Query processing (1)

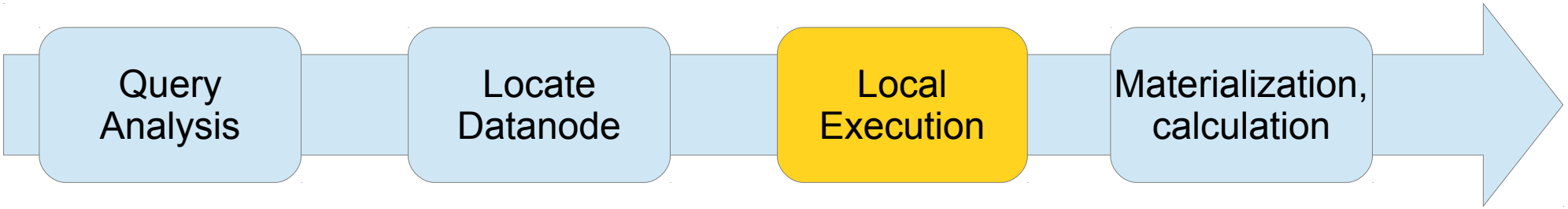
NTT DATA

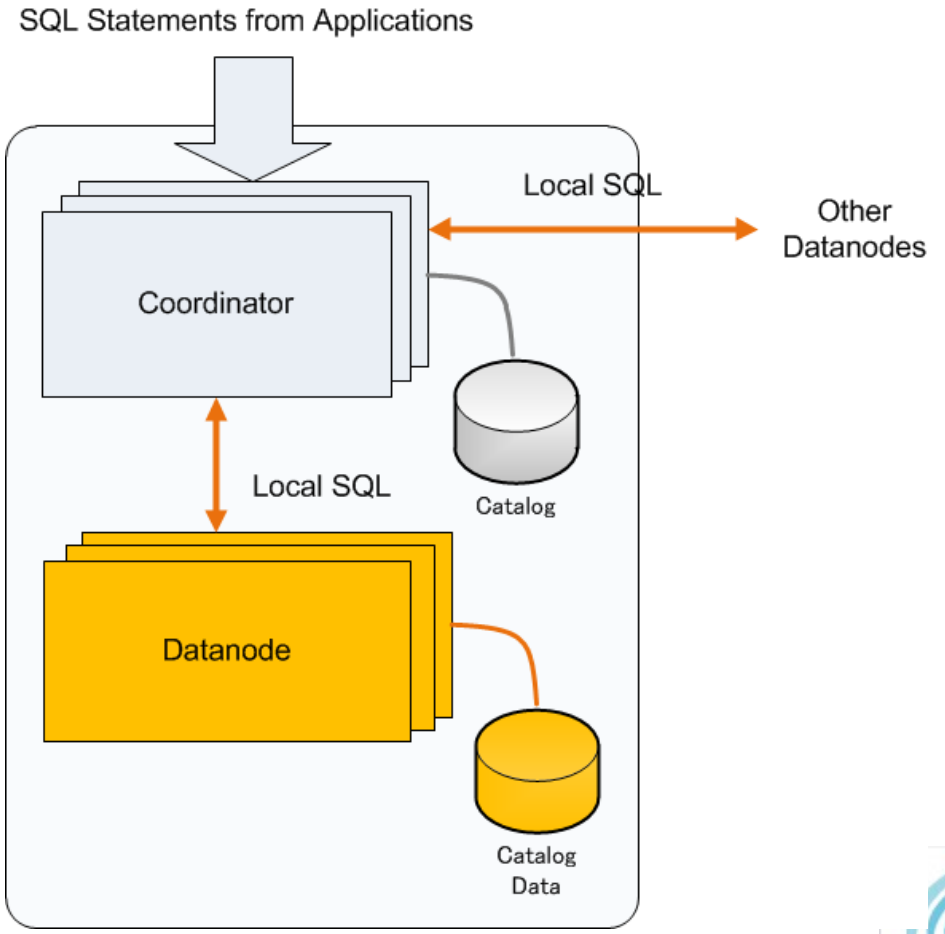




Query processing (1)

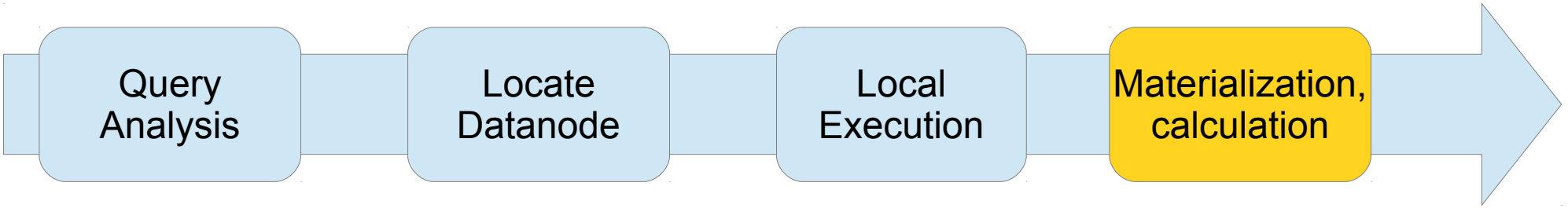
NTT DATA

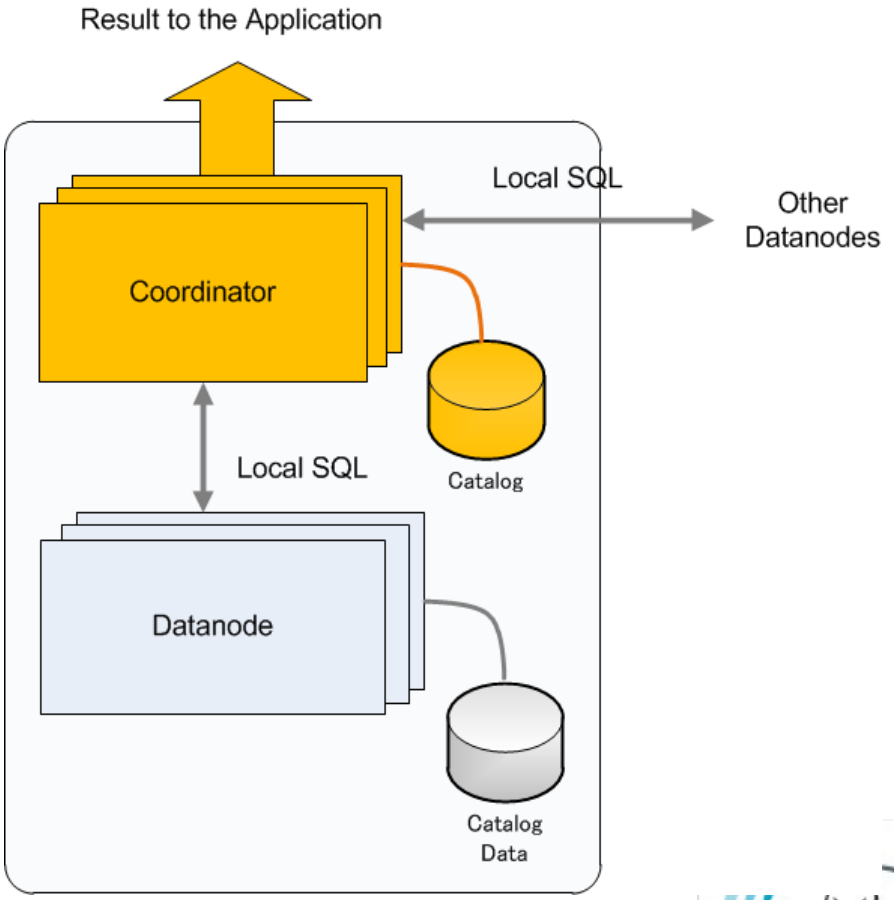




Query processing (1)

NTT DATA





Query optimization

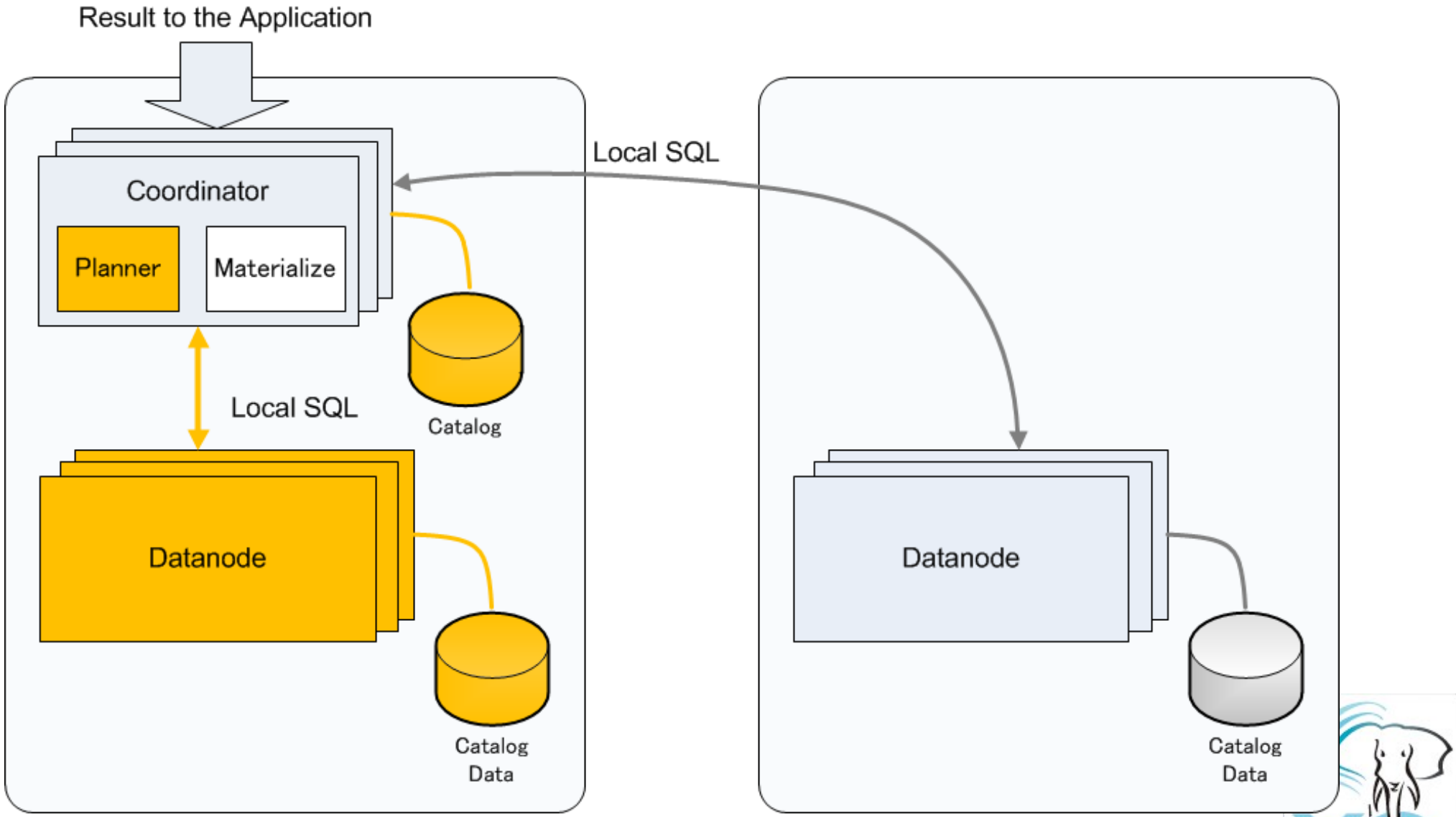
NTT DATA

- Clause push down
  - Join
  - WHERE
  - Aggregate
  - Functions (immutable only)
  - Column Projection
- Utilize info
  - Table distribution
  - Distribution key
  - Shading algorithm
- First Query Shipping (FQS)
  - Determine simple query
  - Ship to datanodes quickly

Optimization example (Join-1)

NTT DATA

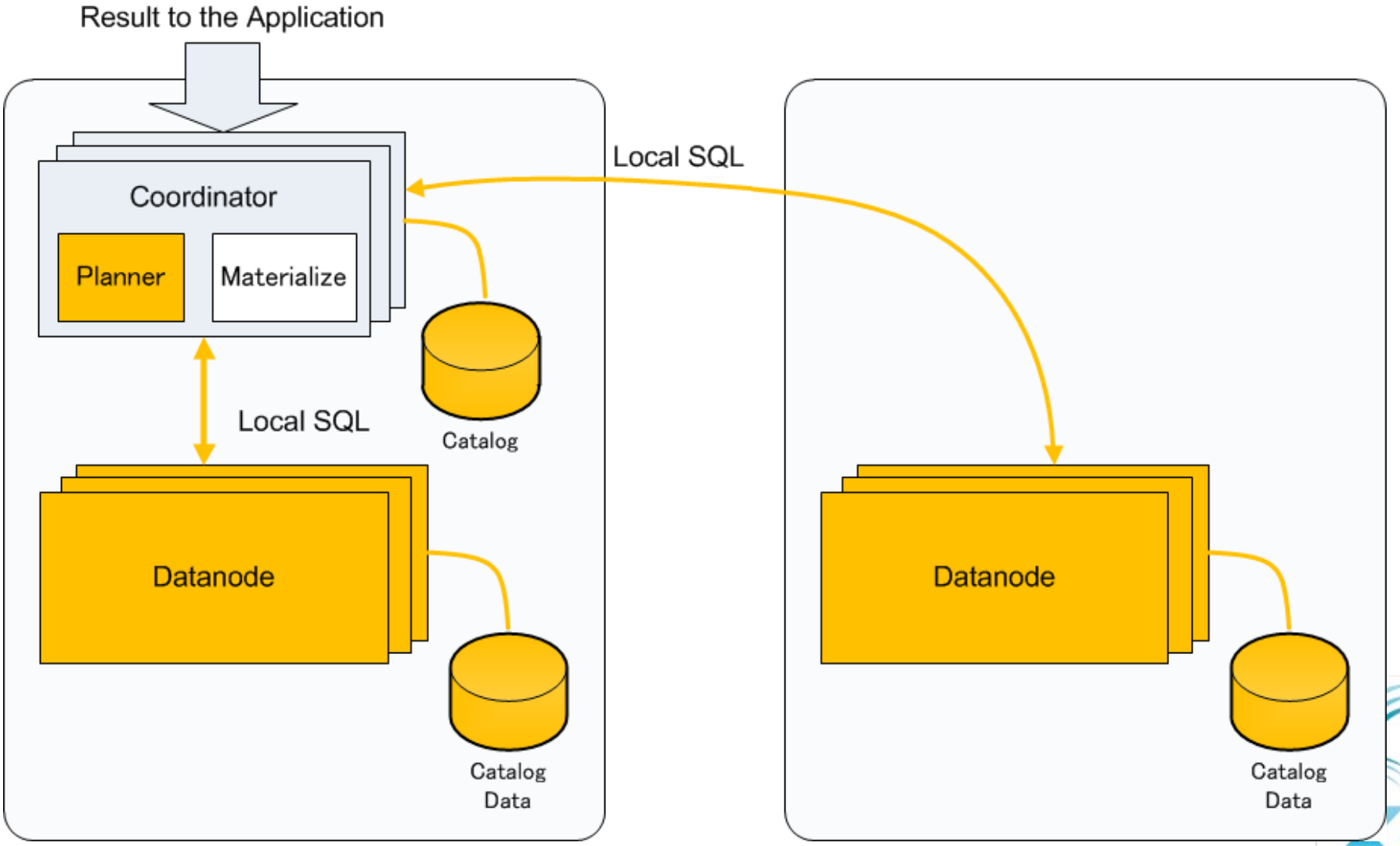
- Both tables are replicated



Optimization Example (Join-2)

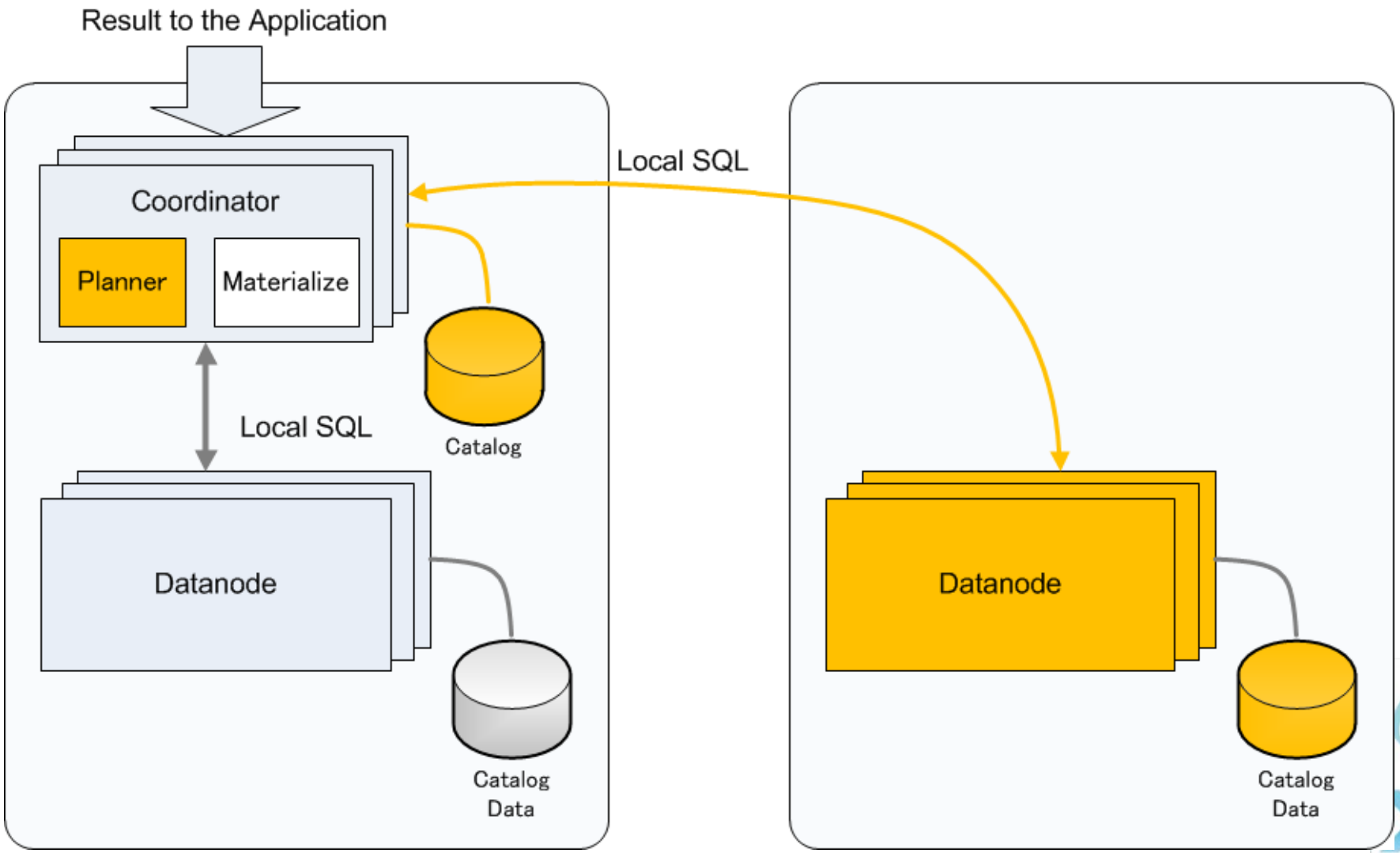
NTT DATA

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



Optimization Example (Join-3)

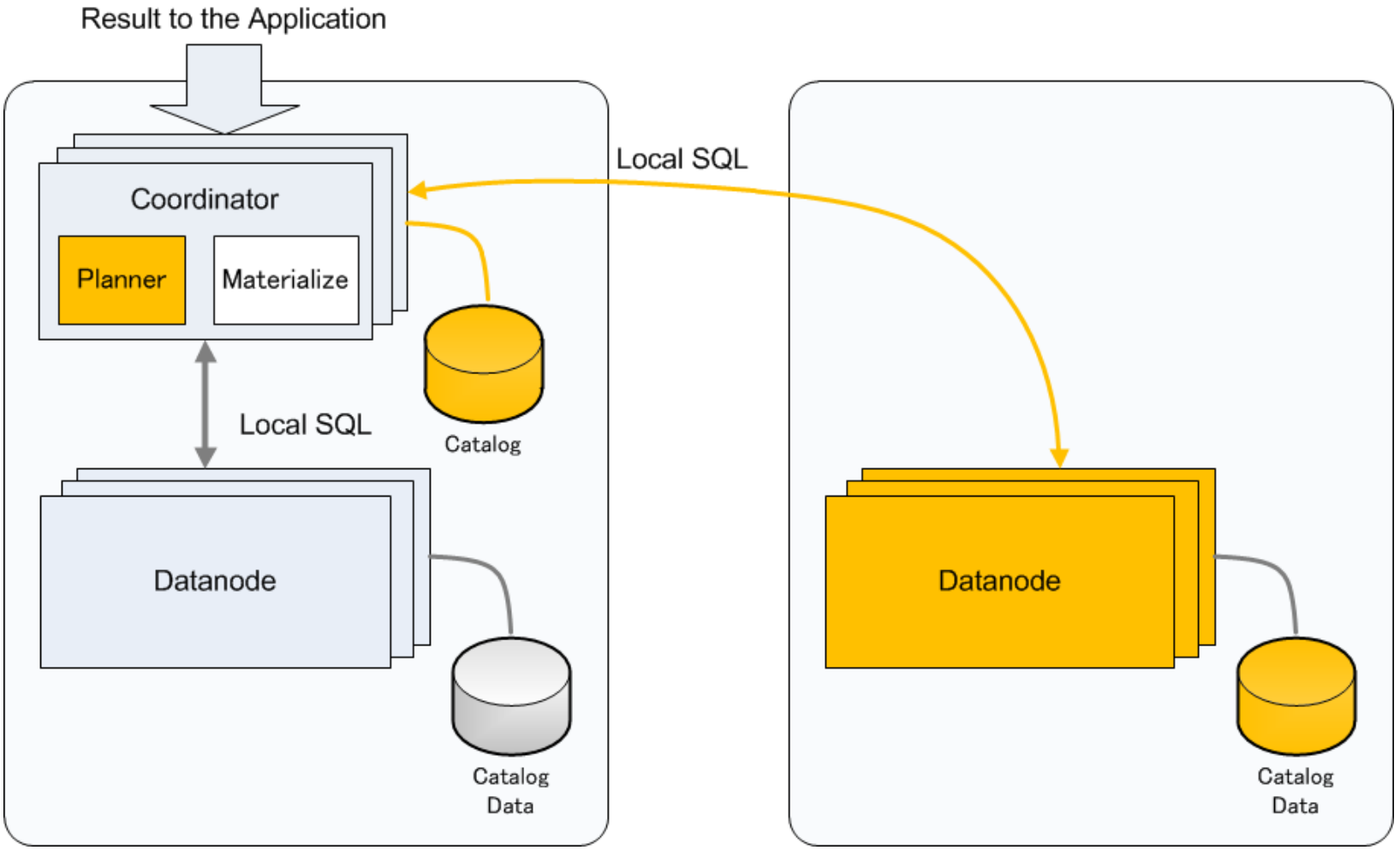
- Replicated Table and Partitioned Table
  - Can determine which datanode to go from WHERE clause



Optimization Example (Join-4)

NTT DATA

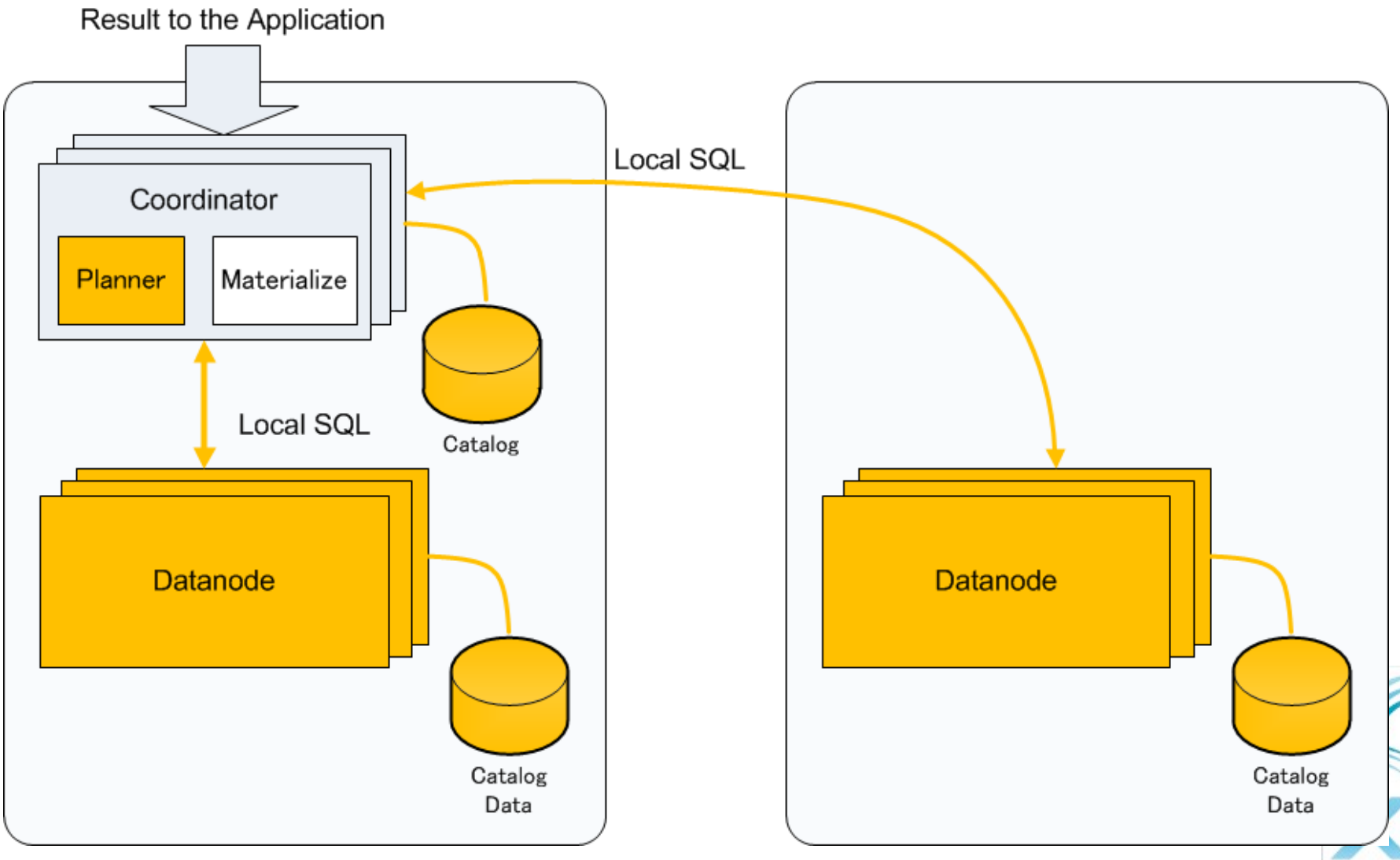
- Partitioned Table and Partitioned Table
  - Both Join columns are distribution (partitioning) column
  - Where clause can determine which datanode to go



Optimization Example (Join-5)

NTT DATA

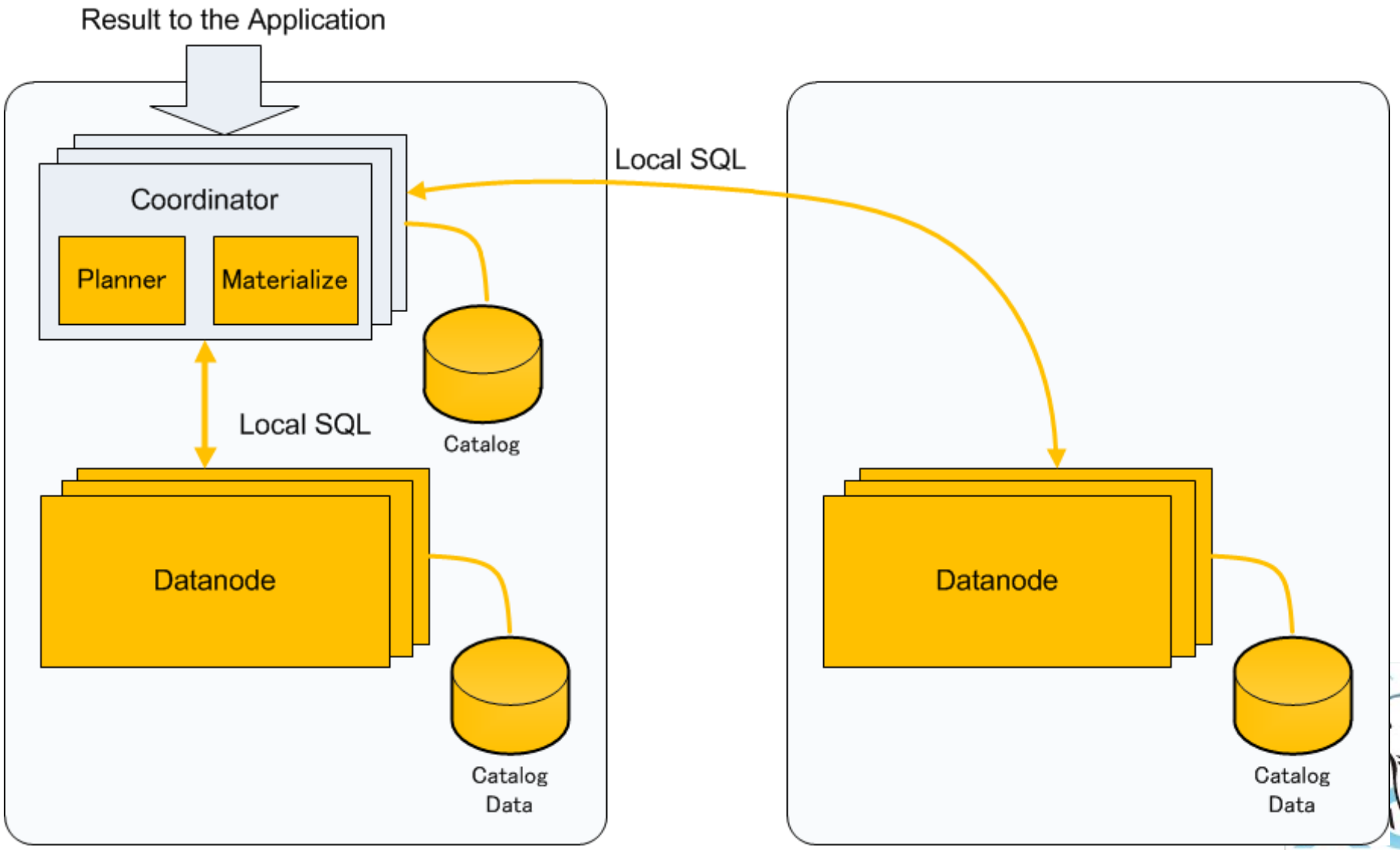
- Partitioned Table and Partitioned Table
  - Both Join columns are distribution (partitioning) column



Optimization Example (Join-5)

NTT DATA

- Partitioned Table and Partitioned Table
  - One of Join columns are not distribution (partitioning) column

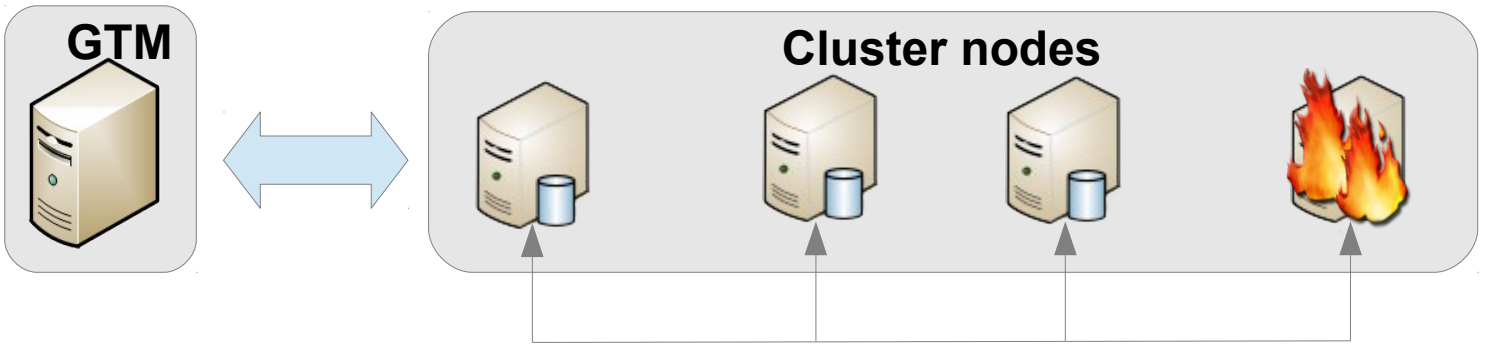


High Availability

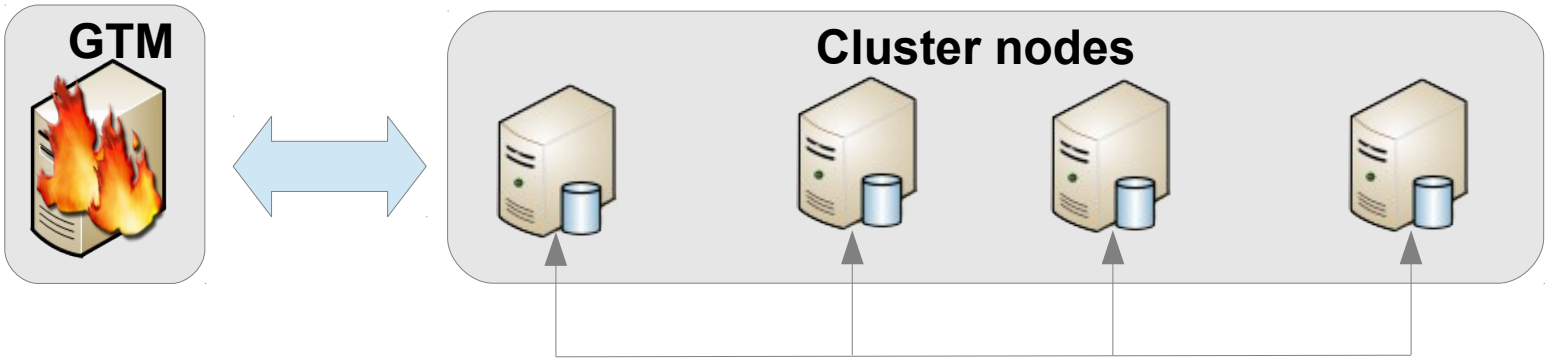
Cluster SPOF problem

NTT DATA

Datanode is a SPOF if it has a portion of distributed table.

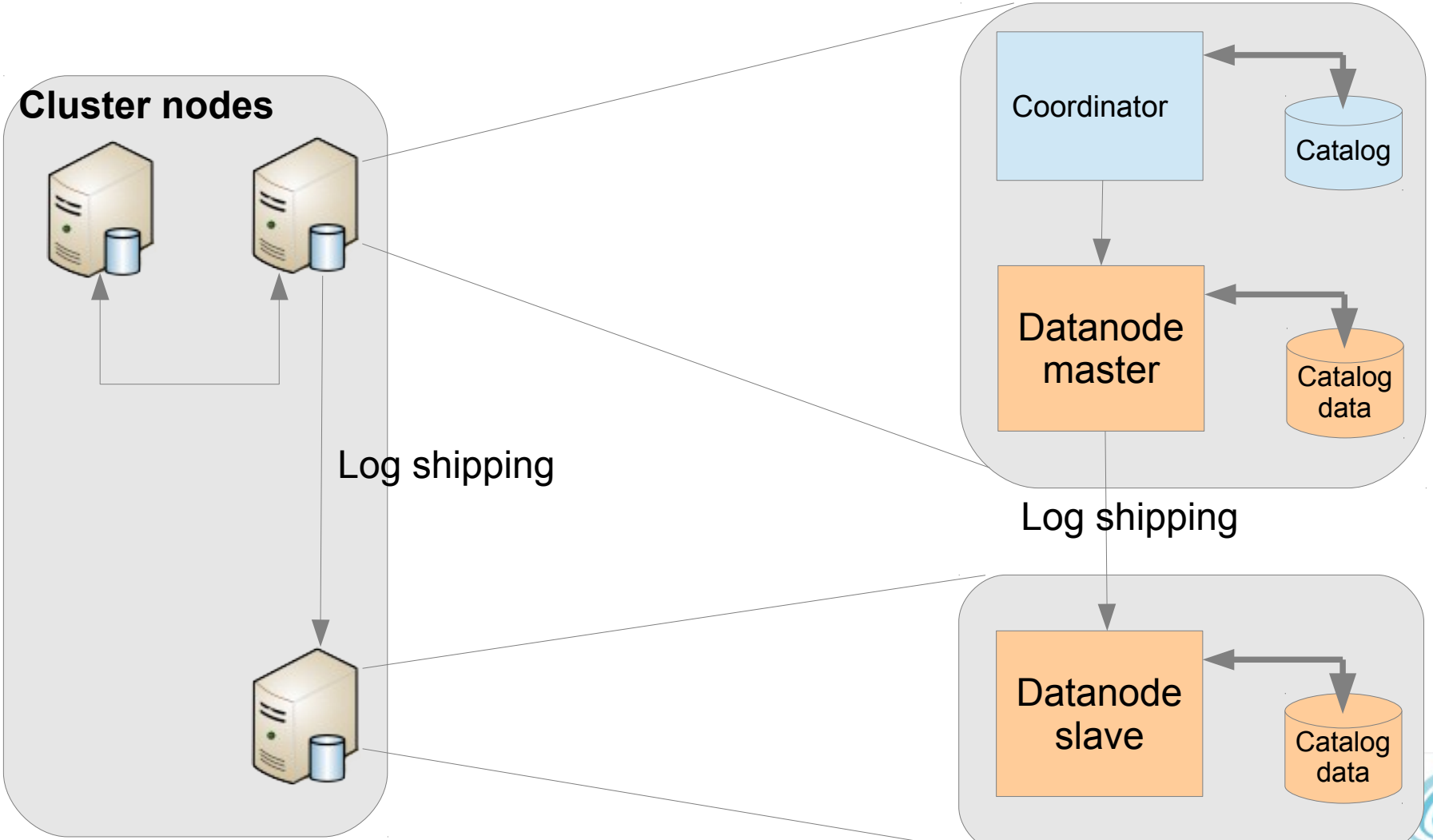


GTM case



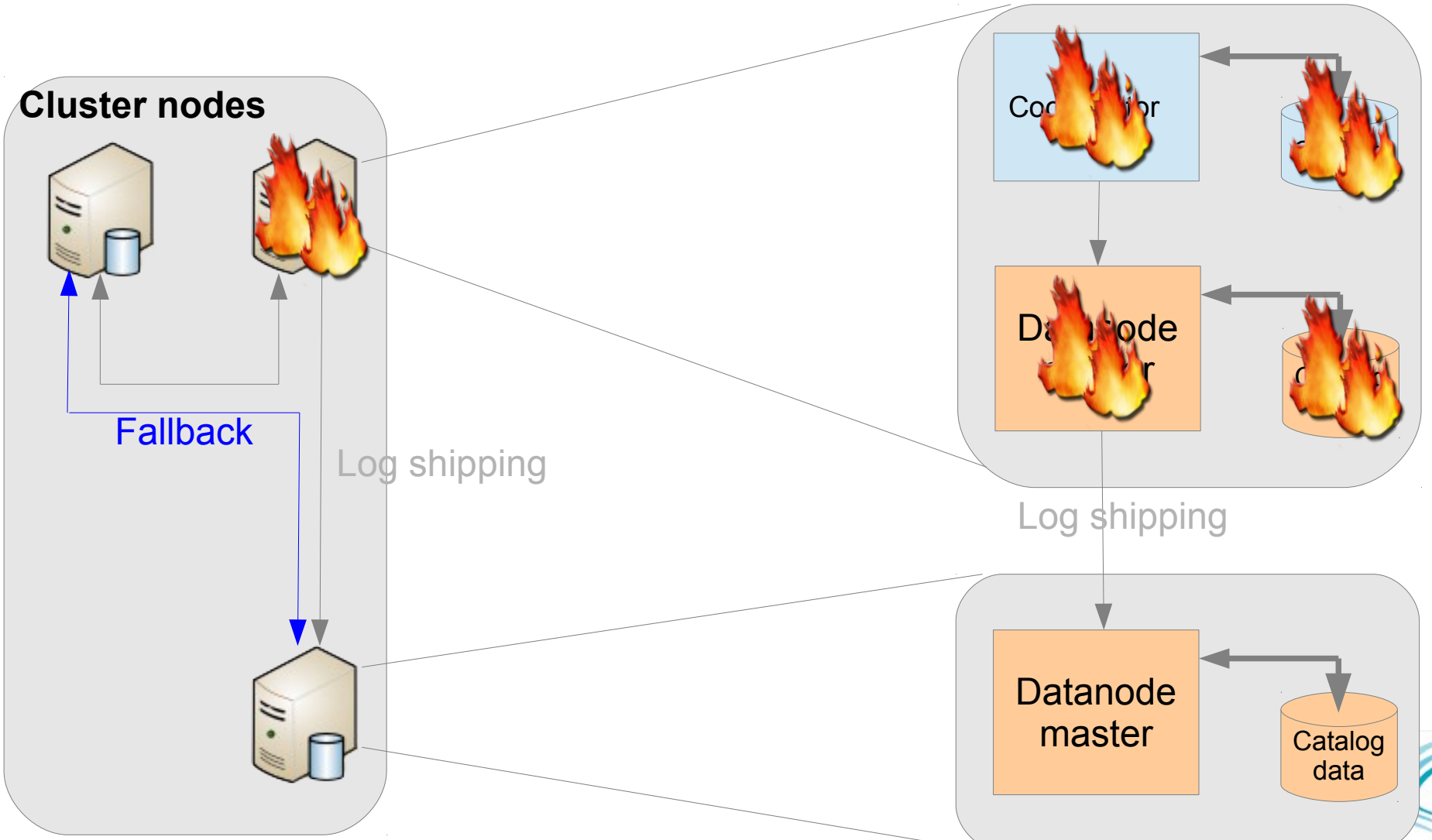
Datanode SPOF resolution (1)

- PostgreSQL 9.1 synchronous stream-rep



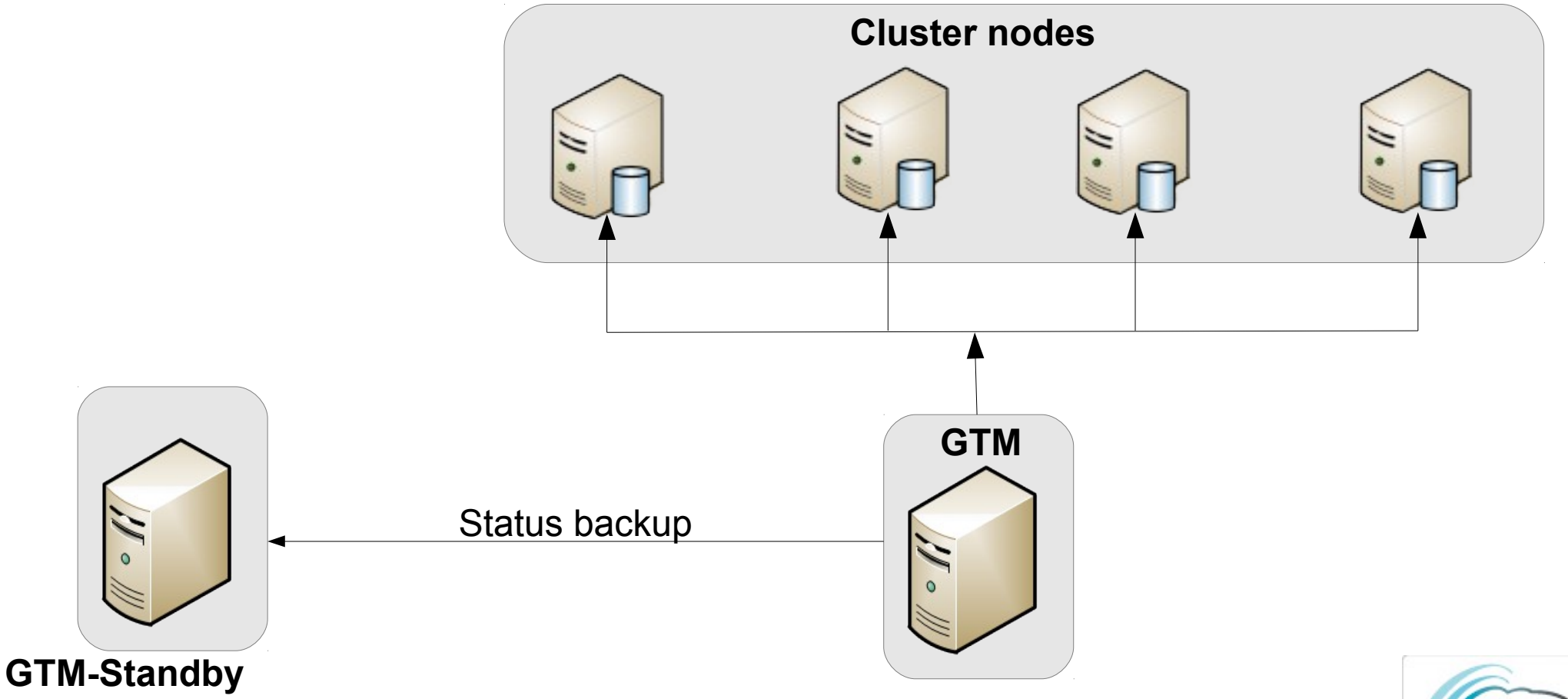
Datanode SPOF resolution (2)

- Fallback slave node



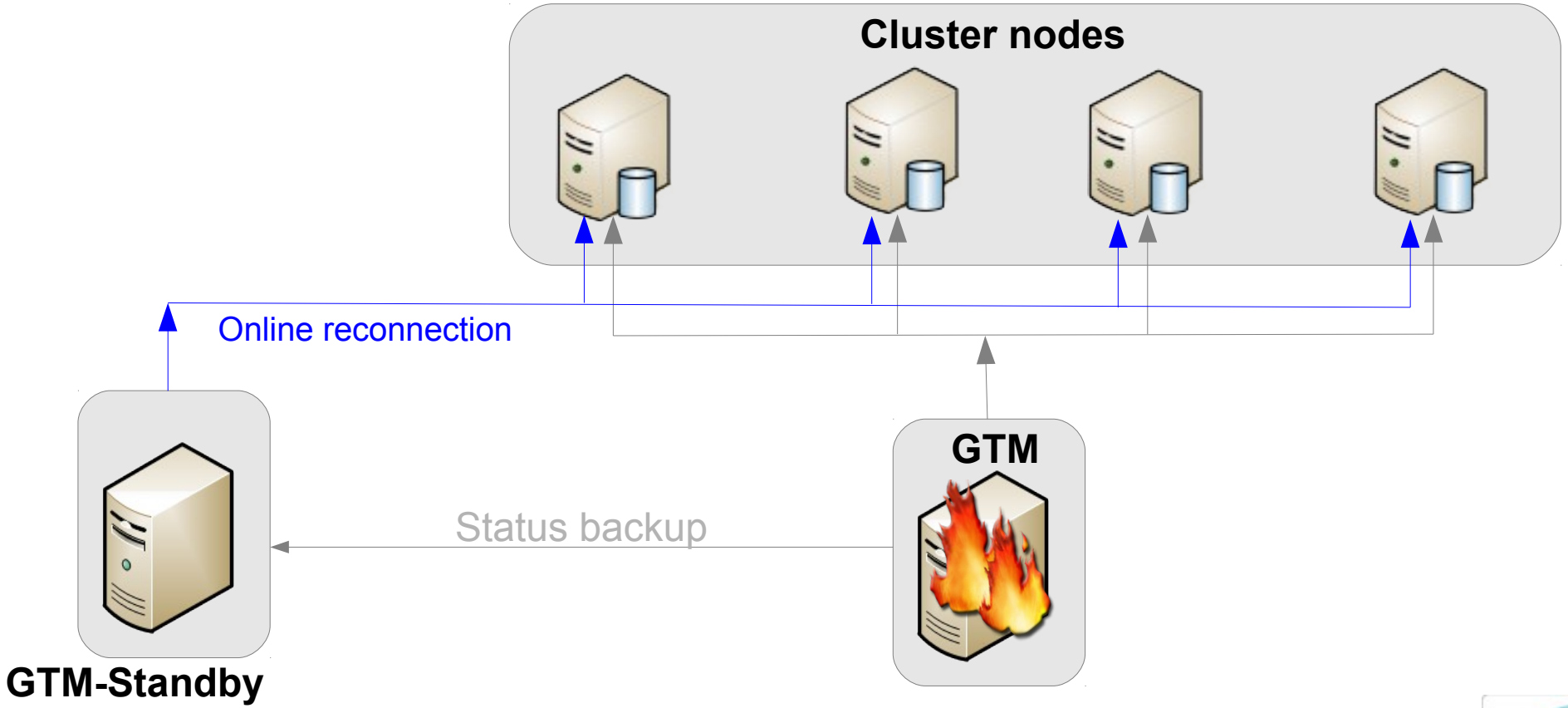
GTM SPOF resolution (1)

- Use of a standby for GTM



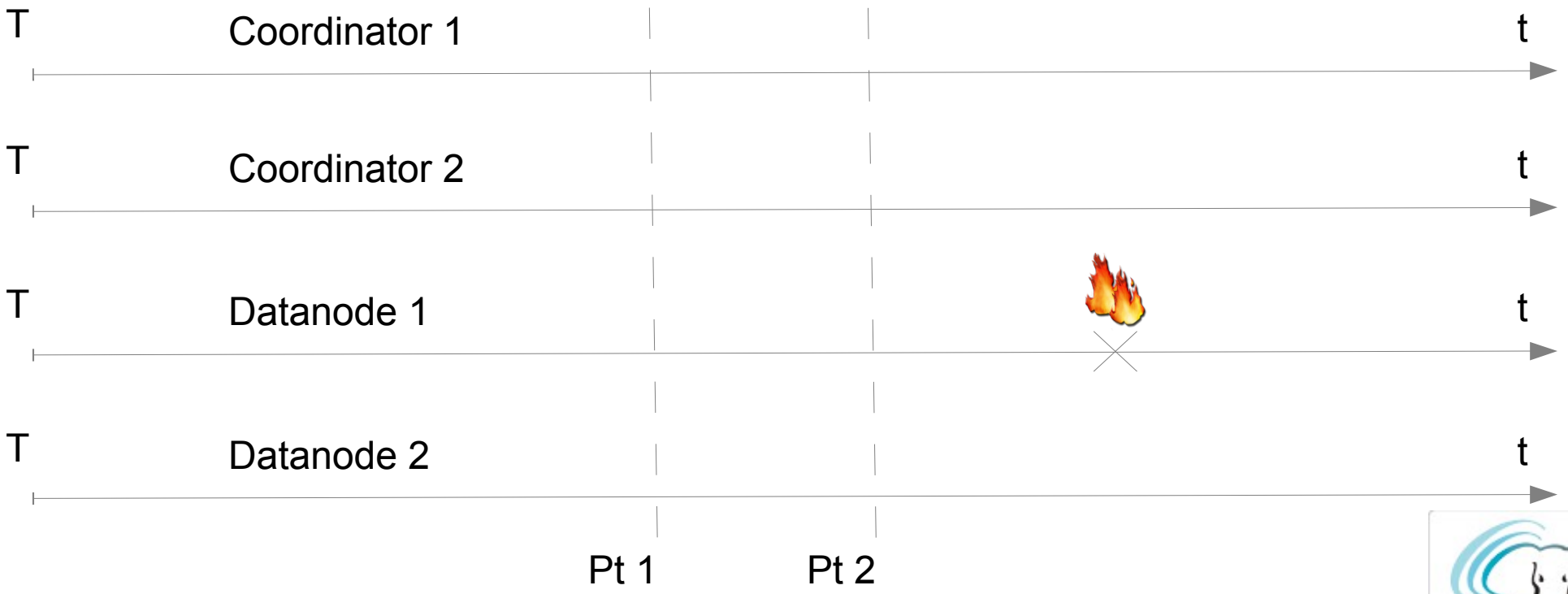
GTM SPOF resolution (2)

- Fallback to standby and reconnect nodes



PITR – requirements (1)

- PITR, Point in-time recovery
  - Rollback the database to a given past state
  - Need consistent points to restore shared-nothing nodes



PITR – requirements (2)

NTT DATA

- Transaction status has to be consistent in the cluster
- Each transaction must be either:
  - Committed/Prepared/Aborted/Running on all the involved nodes
  - We must avoid cases where transaction is prepared and committed partially, or prepared and rollbacked partially
- Write record in WALs of all the coordinators and datanodes at a moment when all the transaction statuses are consistents.
- External Application can provide such timing as with BARRIER
  - `CREATE BARRIER` *barrier_id*
- BARRIER:
  - Waits that partially committed or aborted transactions commit (2PC)
  - Blocks all transaction commit when running a barrier
  - Still needs a timeout functionality
- When running PITR, specify recovery_target_barrier in recovery.conf

Recovery pgxc_clean

- Datanode crash may result in inconsistent 2PC status
- `pgxc_clean` finds such inconsistencies and cleans them up
- Super-user maintenance tool.

Release Status

Version 1.0 released

NTT DATA

- Released at June 8th, 2012
  - Source tarball
  - GIT download
- Comes with most of SQL feature
- Basic HA feature
- Still some restriction such as
  - Trigger
  - WHERE CURRENT OF
  - Savepoint
- Community members are working to provide binary packages

Version 2.0 expected

- Node addition/removal
- Table redistribution
- Trigger
- WHERE CURRENT OF
- Planner improvement
- PostgreSQL 9.2 merge
- More HA-related feature
  - Component watchdog

Project Resources and Contacts

- Project home
  - http://postgres-xc.sourceforge.net
  - https://sourceforge.net/projects/postgres-xc/
- Developer mailing list
  - postgres-xc-developers@lists.sourceforge.net
  - postgres-xc-general@lists.sourceforge.net
- Contacts
  - koichi.clarinet@gmail.com
  - michael.paquier@gmail.com
- Twitter: @koichiclarinet

PGConfChina 2012

PostgreSQL usage in NTT group

NTT DATA

NTT Overview

NTT DATA

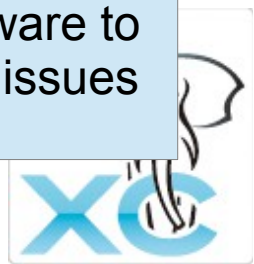
- Japanese Giant Carrier
  - Revenue: 10.2 trillion yen ($120 billion)
    - World's second largest telecommunication company
  - Employees: 200,000
  - Business
    - Consolidated Subsidiaries: 536 (may be more now …)
    - Telecommunication subscribers: 93 million
      - Regional, long distance, mobile
    - System Integration
      - Customers including large companies and government organizations
    - Others
      - Construction, medical, publishing, florists, etc.

NTT System Characteristics

NTT DATA

- Telecommunication operation system (OpS)
  - Large-scale
    - Each DB is large (e.g., 100GB or more) and some communicate each other.
  - High availability and reliability
    - More than 99.999% availability
  - Long-life
    - Expected lifetime is 7 years.
- Issues
  - Proprietary DBMS are widely used.
    - High-cost, short support duration
    - Vendor lock-in

We expect Open Source Software to solve these issues



Open Source Software Center

NTT DATA

- Mission
  - Reduce TCO with OSS, by replacing proprietary software
    - Support NTT Group companies' OSS usage
      - Q and A
      - Consultation
    - Develop/Improve OSS
  - Center of OSS competence in NTT Group
- Established in Apr.2006
- Location: Shinagawa, Tokyo.

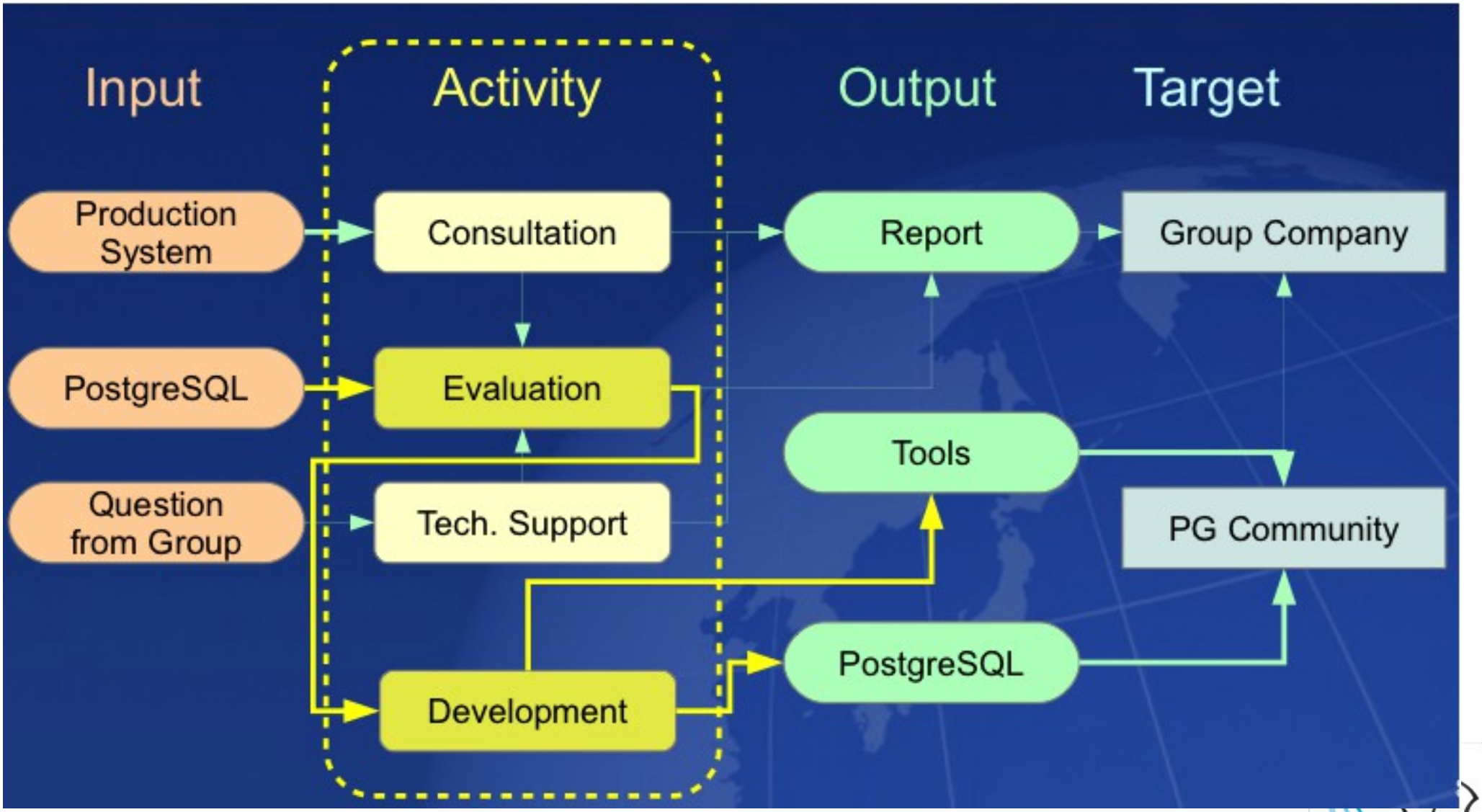
Understanding User Needs

NTT DATA

How to encourage PostgreSQL use?

- Information on performance
  - Show good and stable performance
  - Availability/reliability
    - Downtime to recovery (e.g. annually 5min, for five-9s)
  - Hardware (HDDs, CPUs, etc.)
- Operation Capability
  - Compatibility with other operation tools
  - Usability
- Improve performance and usability
- Technical support

Activities



Needs evaluations

NTT DATA

- System characteristics
  - Most sytems are OLTP, not OLAP
  - Types of transactions: read/write intensive
- TPC-C and TPC-W models are used.
  - DBT-2 (TPC-C): write, I/O itensive
  - DBT-1 (TPC-W): read, CPU intensive
  - Other: pgbench, DBT-3
- Throughput and stability
  - Peak performance test (3hr., workload > 90%)
    - CPU scalability evaluated
  - Long-run test (72hr., 70% workload)
    - Observe stability during vacuum and checkpoint

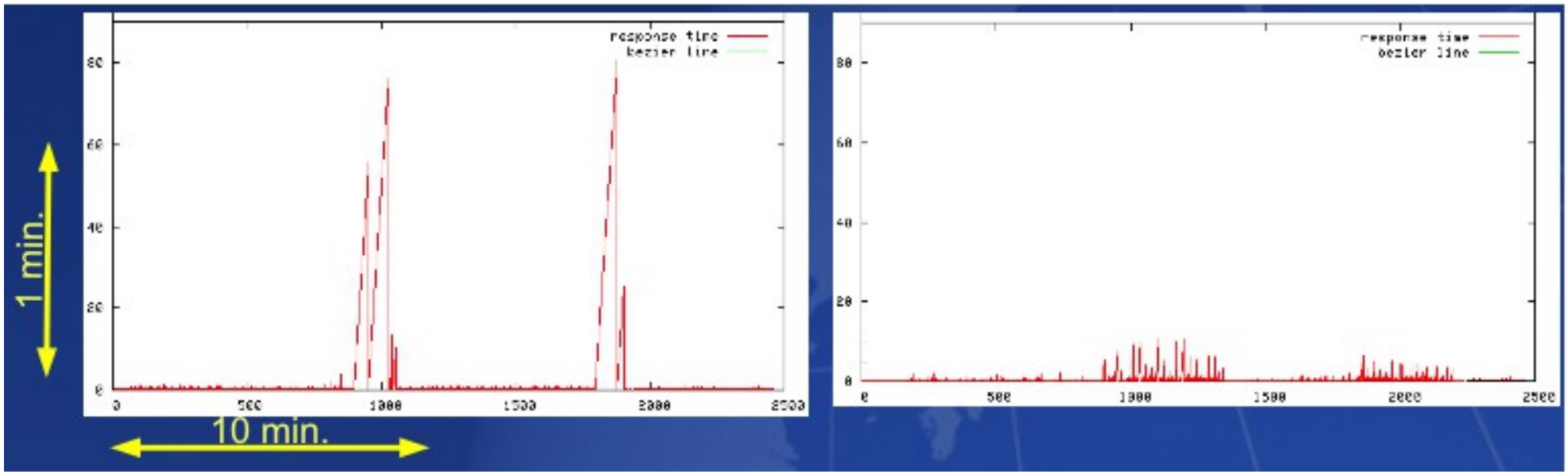
NTT's effort for PostgreSQL improvement

NTT DATA

- Fix context switch storm (8.2)
- Smoothing performance during checkpoint (8.3)
- High-speed data loader (8.1)
- Archive WAL compression (8.3)
- Log-shipping replication
  - 9.0: Asynchronous
  - 9.1: Synchronous
  - 9.2: Cascade
- Background cluster/vacuum full – pg_reorg
- Automated backup/restoration – pg_rman
- PostgreSQL status monitoring – pg_statsinfo
- And many others...

Performance during checkpoint

- Eliminated most serious performance problems during the checkpoint



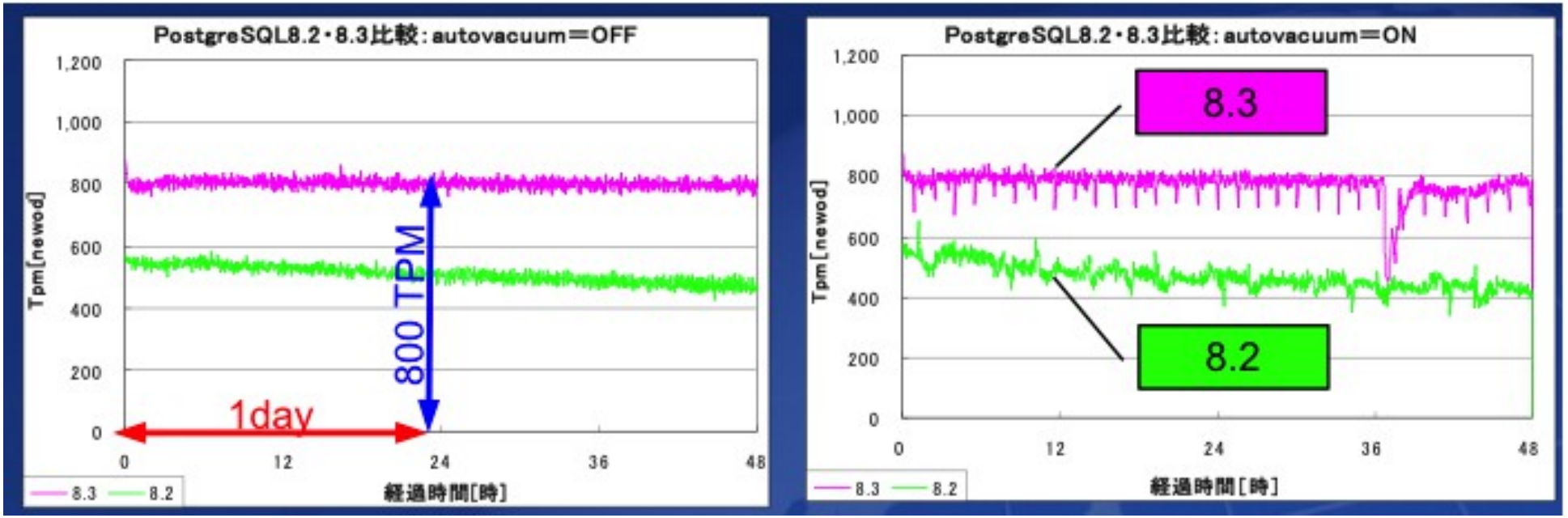
8.2

8.3

Stability Evaluation

NTT DATA

- Effect of HOT and Autovacuum



http://lets.postgres.jp/documents/case/ntt_comware/2

Support Activities

NTT DATA

- Technical Q and A
  - A couple of hundreds of question answered in a year. Each answered within three business days
  - Various question
    - Usage, trouble issues
- Consultation
  - Migration from proprietary DBMS
    - About 50 or more know-hows are accumulated.
  - Performance tuning aids
    - Evaluate particular workloads and suggest tuning
  - Recovery
    - Recover corrupted database

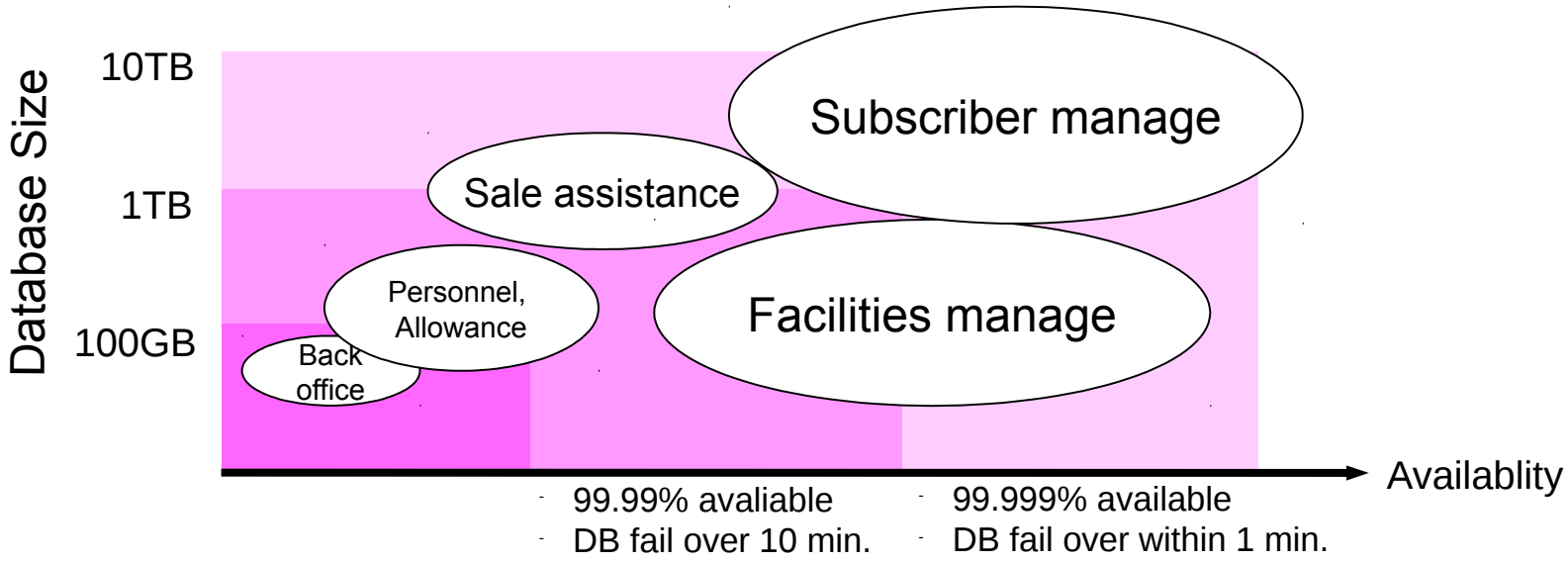
NTT's experience

NTT DATA

- Introduced more than PostgreSQL to more than 200 systems. Highlight of database specs are as follows:
  - Size: Largest 3TB
  - Throughputs: 1000 TPS (or more)
  - HA: failover in less than 1min (15sec., measured)

View of NTT Production Systems

- Target of OSS in NTT in-house systems
  - NTT runs several hundreds of systems
  - Survey shows PostgreSQL can be applied to 80% of them.
- Trend of PostgreSQL deployment
  - Beggining from small-scall, less available system to large-scale, high availabile system

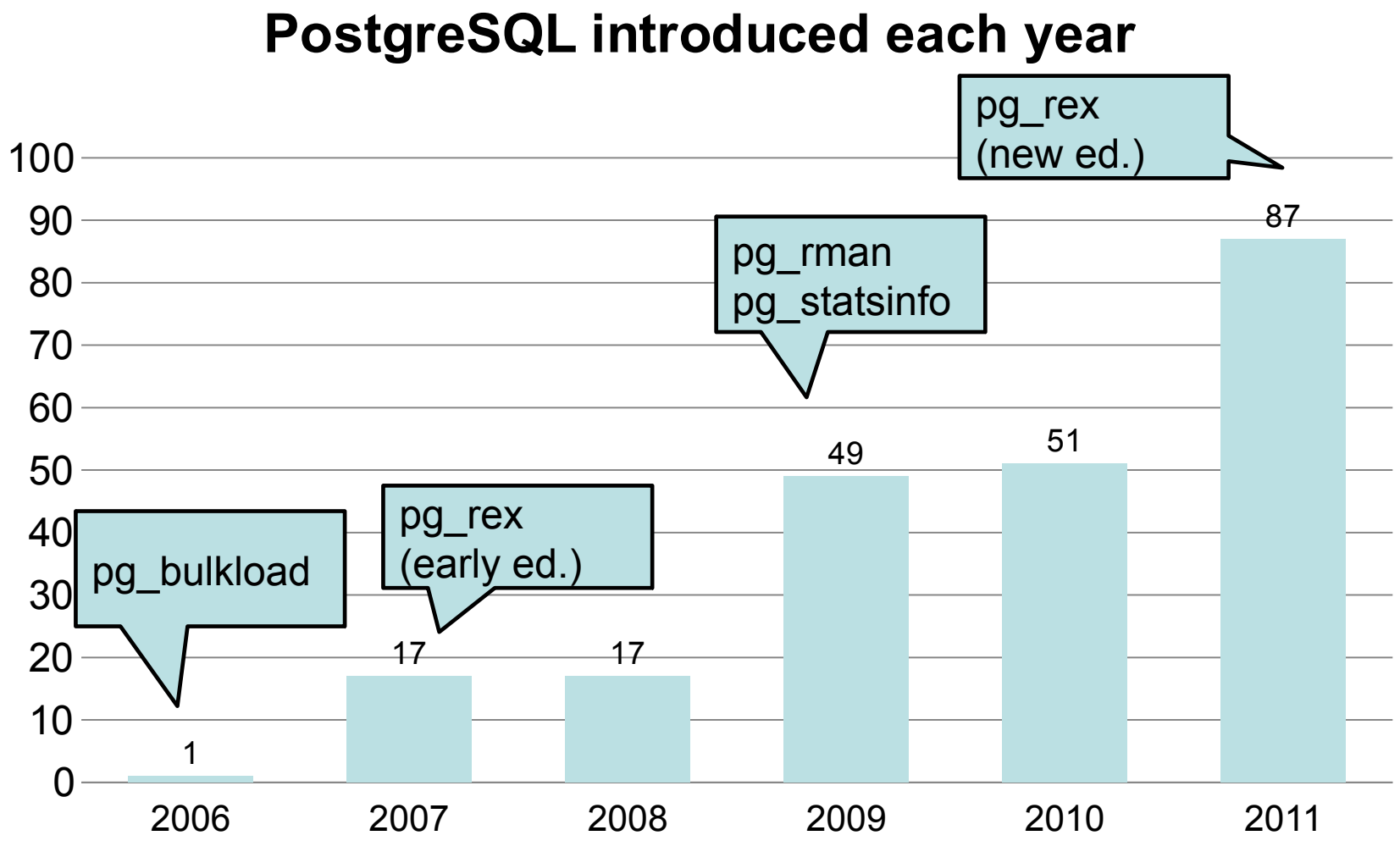


PostgreSQL tools from NTT

NTT DATA

- All open source
  - pg_bulkload (fast data loader)
  - pg_rman (backup and restore tool)
  - pg_statsinfo (monitors PostgreSQL database activities
  - pg_reorg (concurrent vacuum full and cluster)
  - pg_rex (automatic failover with log-shipping replication)
  - pg_lesslog (archive WAL compression)

PostgreSQL deployment



PostgreSQL Application Map

NTT DATA

Eyes Only

Thank you very much!!

koichi.clarinet@gmail.com
koichi@intellilink.co.jp