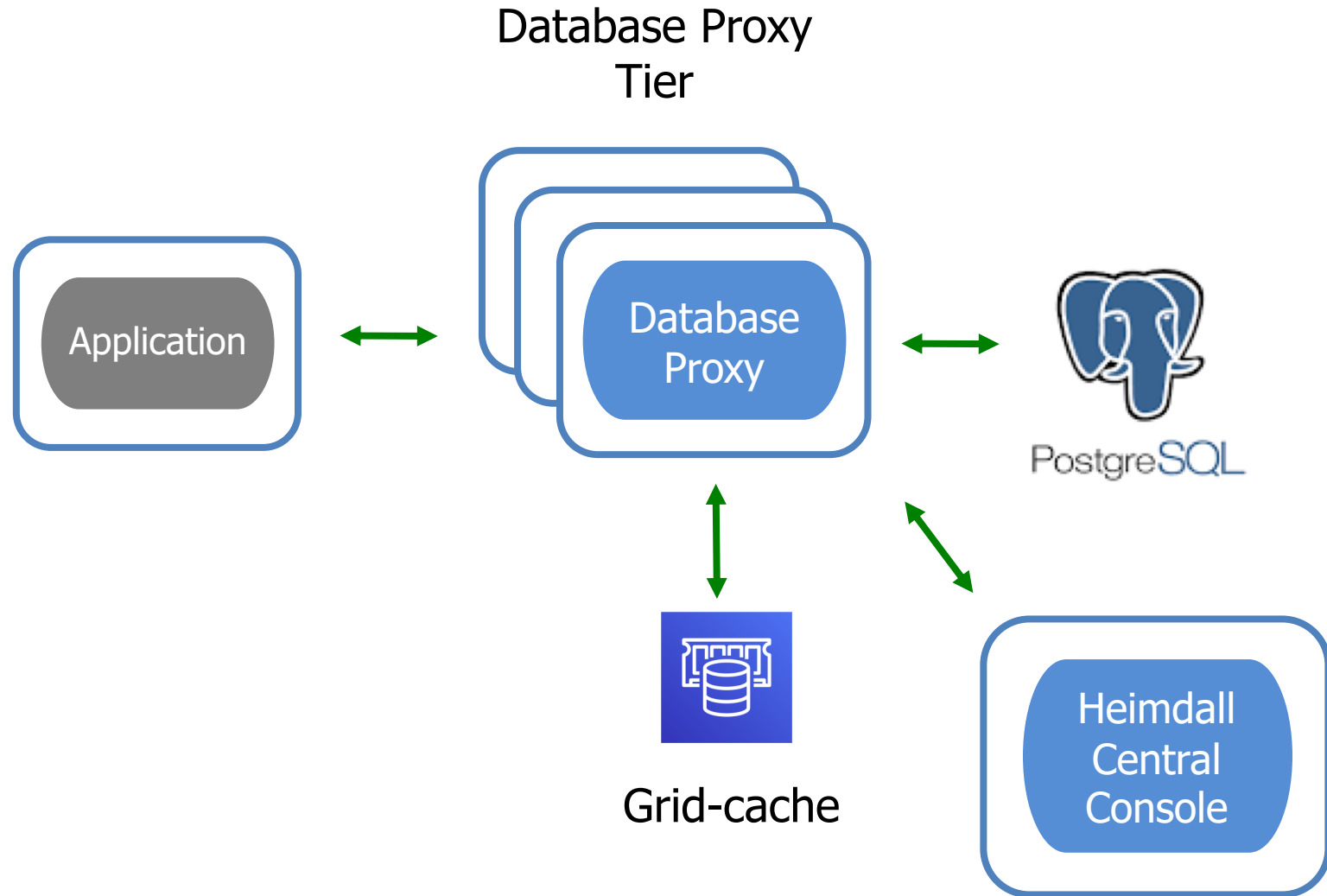


3 Techniques to Improve Postgres Response Times without code changes



Roland Lee



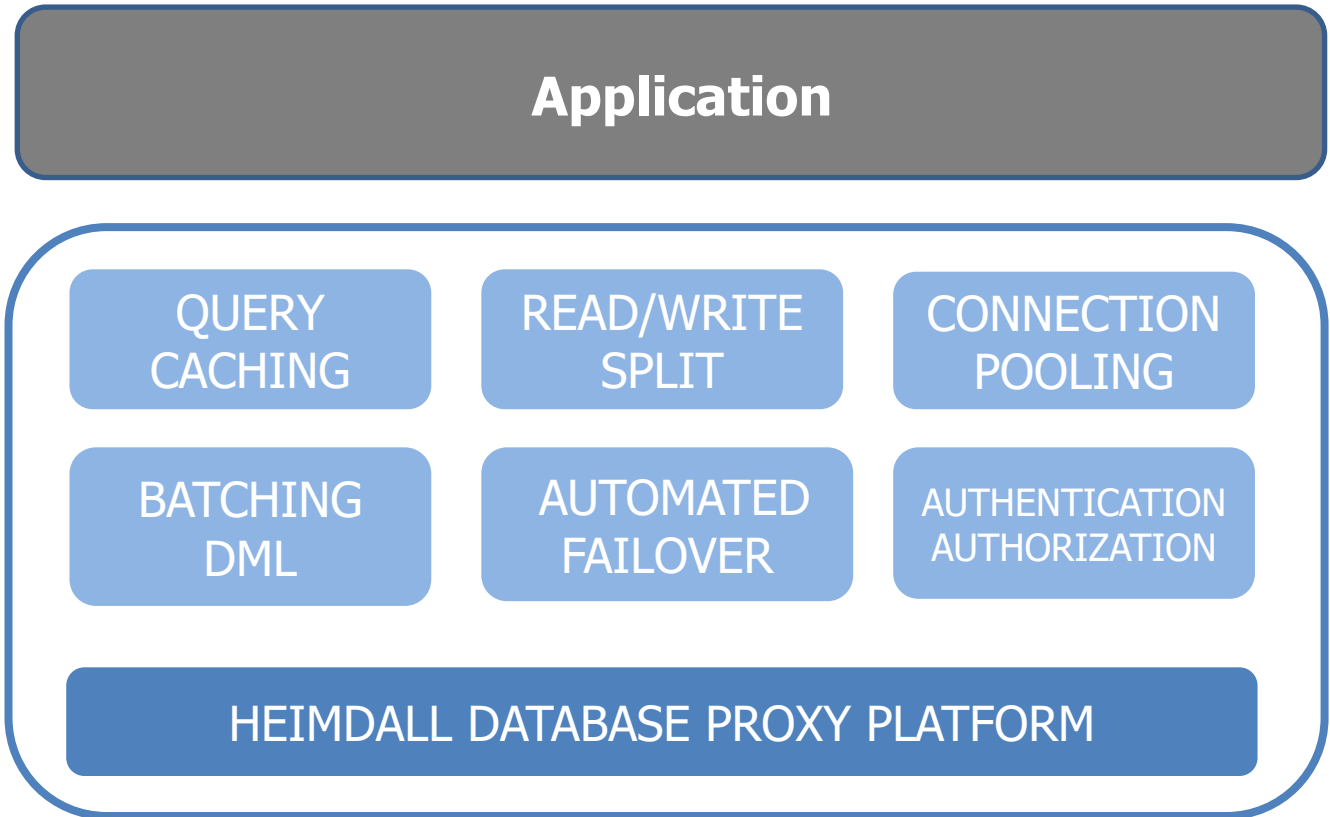
Database Proxy Architecture



Database Proxy Vendors

Feature		PgPool-II	Pg-bouncer	
Databases Supported	All	Postgres	Postgres	Postgres MySQL SQL Server
Connection Pooling	✓	✓	✓	✓
Read/Write Split ACID Compliant	✓	X		
Automated Caching	✓	X		
Active Directory Authentication/Authorization	✓			
Global Database Failover, Low Latency	✓			

Database Proxy Platform



Amazon Aurora



Amazon RDS



Amazon Redshift

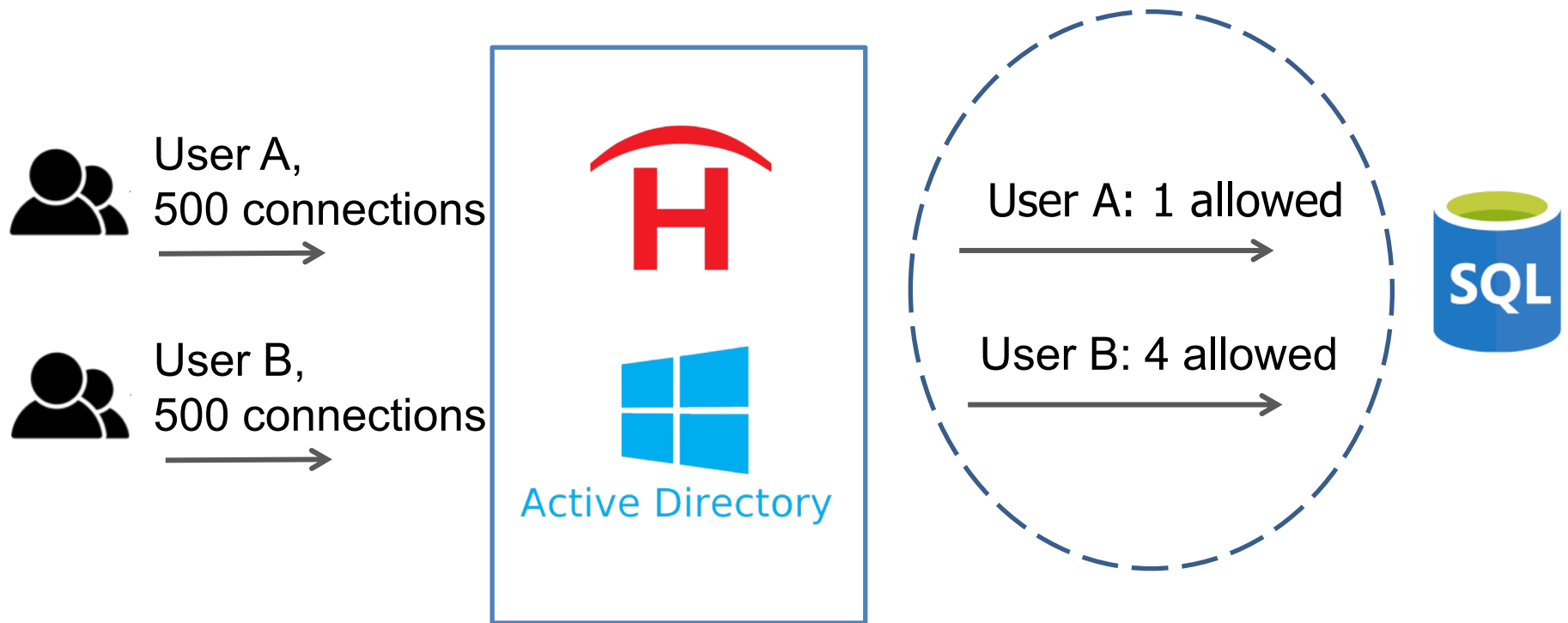


Pivotal Greenplum®

Use Cases

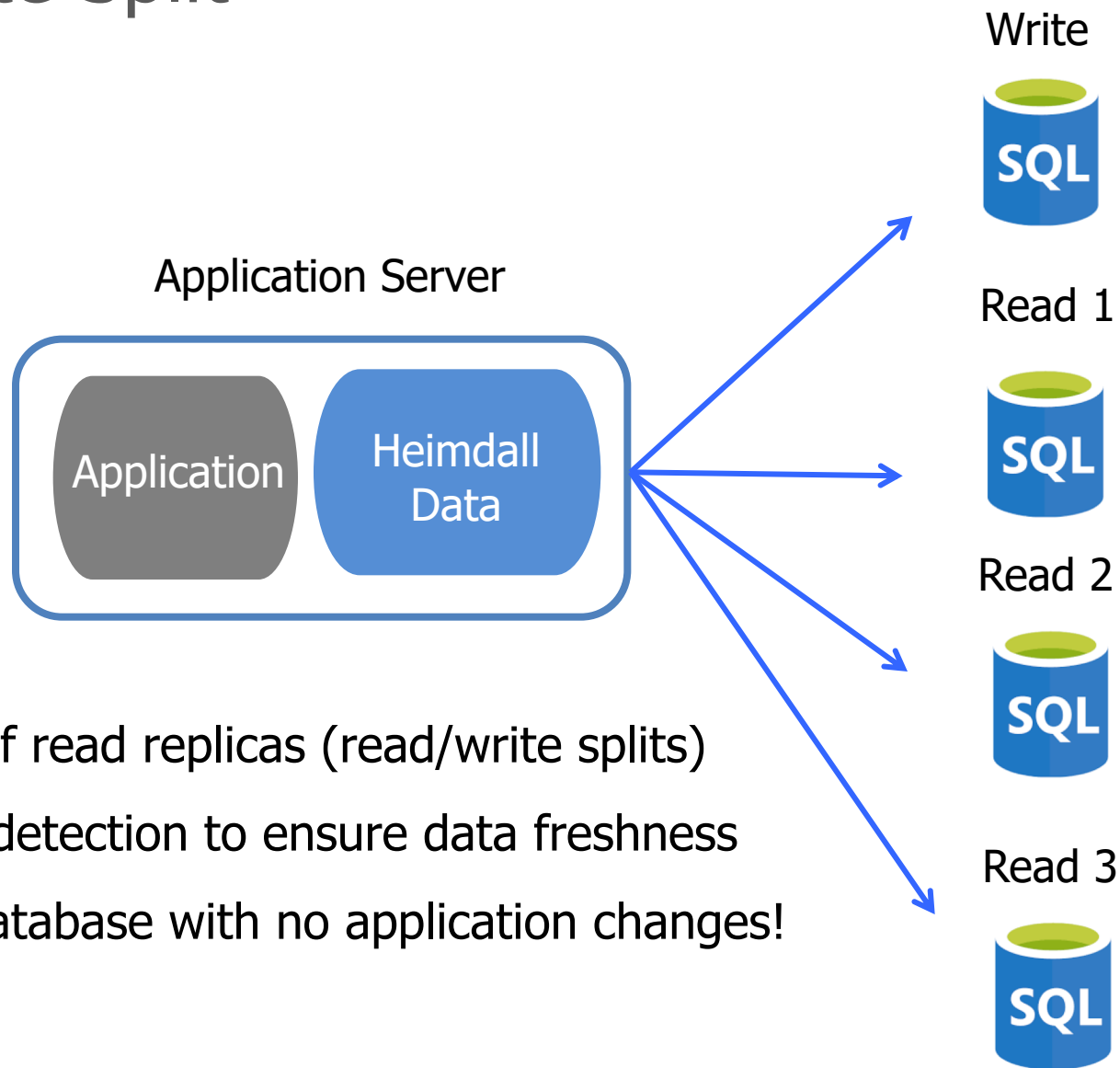
Connection Pooling / Multiplexing

1000:5 Application to Database connection ratio



- Two-tier pooling limits: 1) Per user, 2) Per database instance
- Heimdall delivers the connections for only active concurrent queries

Read / Write Split



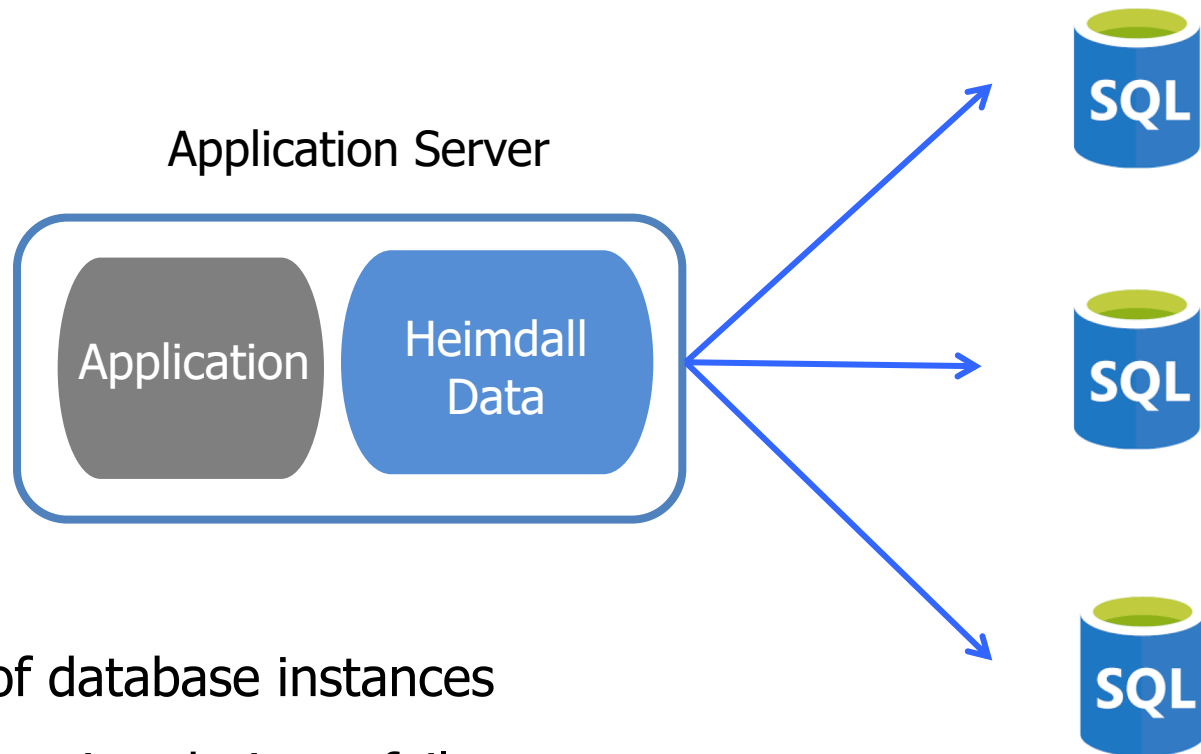
1. Automate use of read replicas (read/write splits)
2. Replication lag detection to ensure data freshness
3. Scale out the database with no application changes!

Global Database Load Balancing



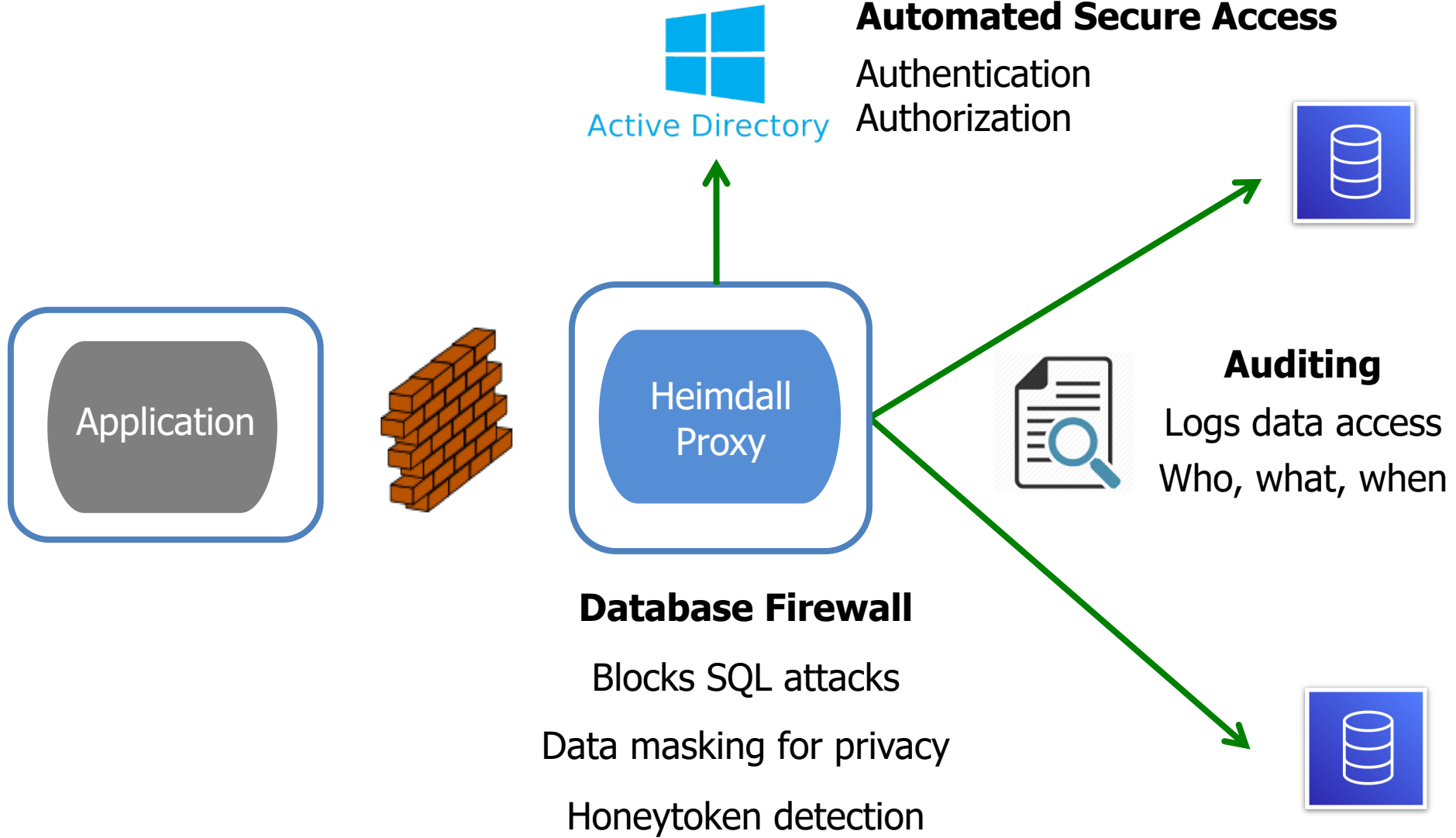
Aurora Global Database: Proxy routes to the closest read replica based on latency

Persisted Connection Automated Failover

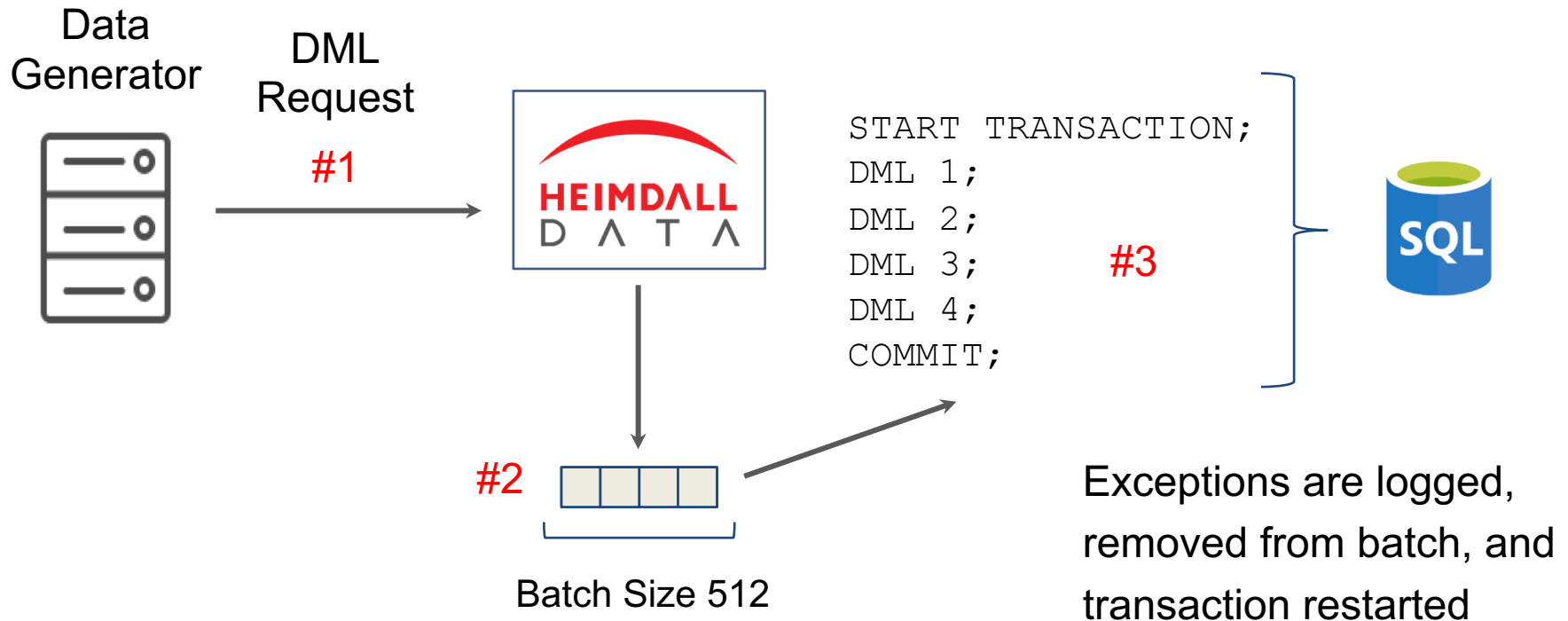


1. Detects health of database instances
2. Queues up connection during a failover
3. No connection drops
4. Reducing application errors and exceptions

Heimdall Proxy SQL Security



Transparently Batching INSERTs



Benefits:

- Lower CPU overhead due to fewer commits
- Improved application response time
- Improved DML scale

Thank You

The logo features a red curved line above the text. The text "HEIMDALL" is in red and "DATA" is in grey.
HEIMDALL DATA