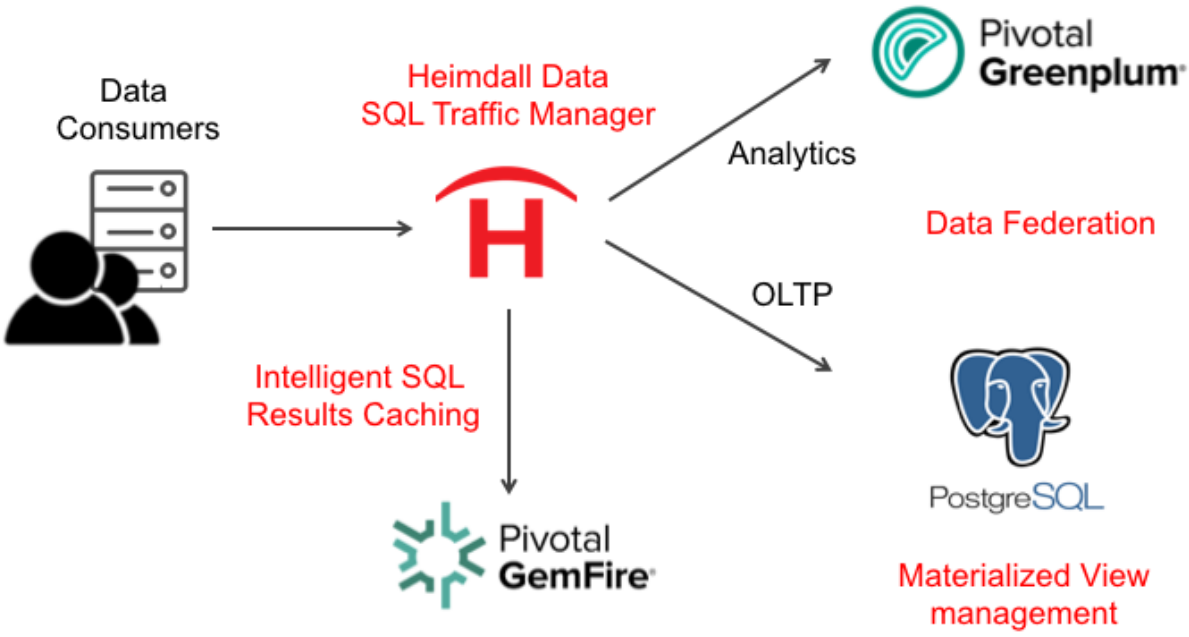
Heimdall Data for Pivotal Greenplum



Companies deploy Pivotal Greenplum for large-scale MPP (Massively Parallel Processing) data analytics. These systems apply parallel compute resources to answer queries quickly. However, there are still performance challenges:

- Transactional overhead: Singleton DML operations result in slow application response times
- Higher latency: Distributed queries means that many nodes have to coordinate to generate an answer
- No materialized views: If a query result is used for multiple following queries, additional work is required to preserve the result or repeated calls to the same base query will be required

These challenges can be solved by modifying the application. However, the ideal solution requires no code changes. Heimdall Data is a SQL Traffic Manager providing Pivotal Greenplum users the ability to:

- Batch singleton, DML operations
- Fast materialized views for results for Pivotal Greenplum
- Automated SQL caching for Pivotal GemFire

Asynchronous Batching Processing

Heimdall Data improves database write performance by batching DML operations against a table and putting them onto a separate connection under a single transaction. Batching DML operations result in:

- Improved application response times due to fewer commits
- Improved DML scale

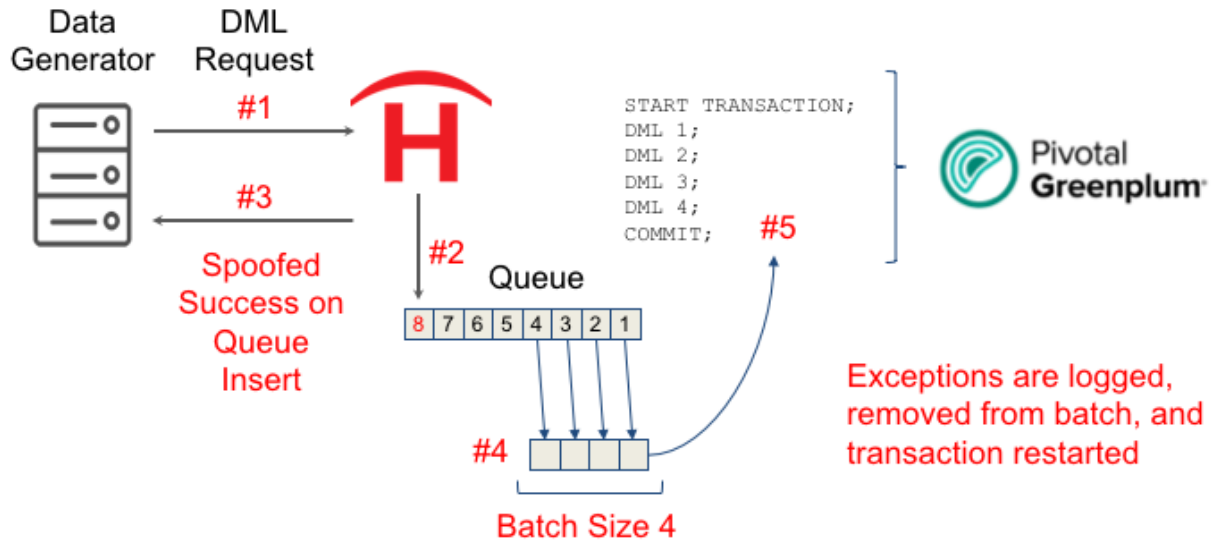
Ideal use case for batch processing:

Insert/update/delete a large amount of data at once on a thread. Heimdall can process it all at once much faster than if individual queries outside of a transaction were completed.

Not so ideal use case for batch processing:

If there are concurrent writes and reads against the same table, on the same thread, as everything will just block until the DML operation is completed anyway.

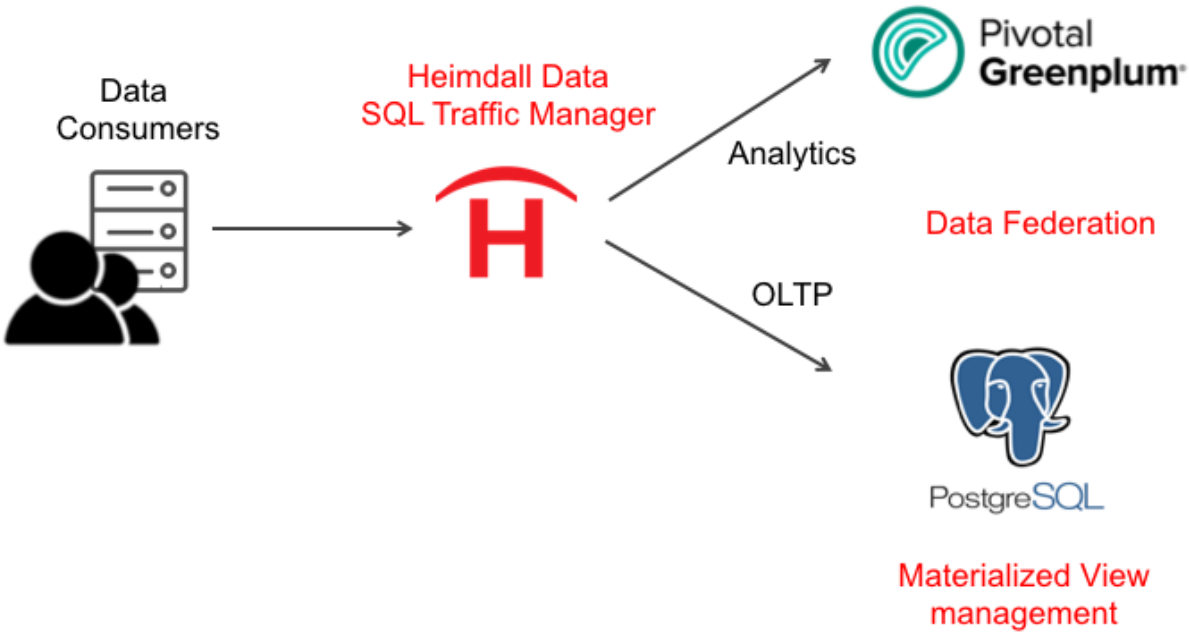
If an exception occurs in the transaction, Heimdall remove the query that caused the exception, report the exception in the logs, and reprocess the batch without the query in place. See the dataflow diagram below:



Materialized views for Pivotal Greenplum

Fast materialized views are very important in analytics environments. When reports are generated, a subset of data is pulled from the back-end data store, then various operations are performed on that data. Heimdall provides the following functionality:

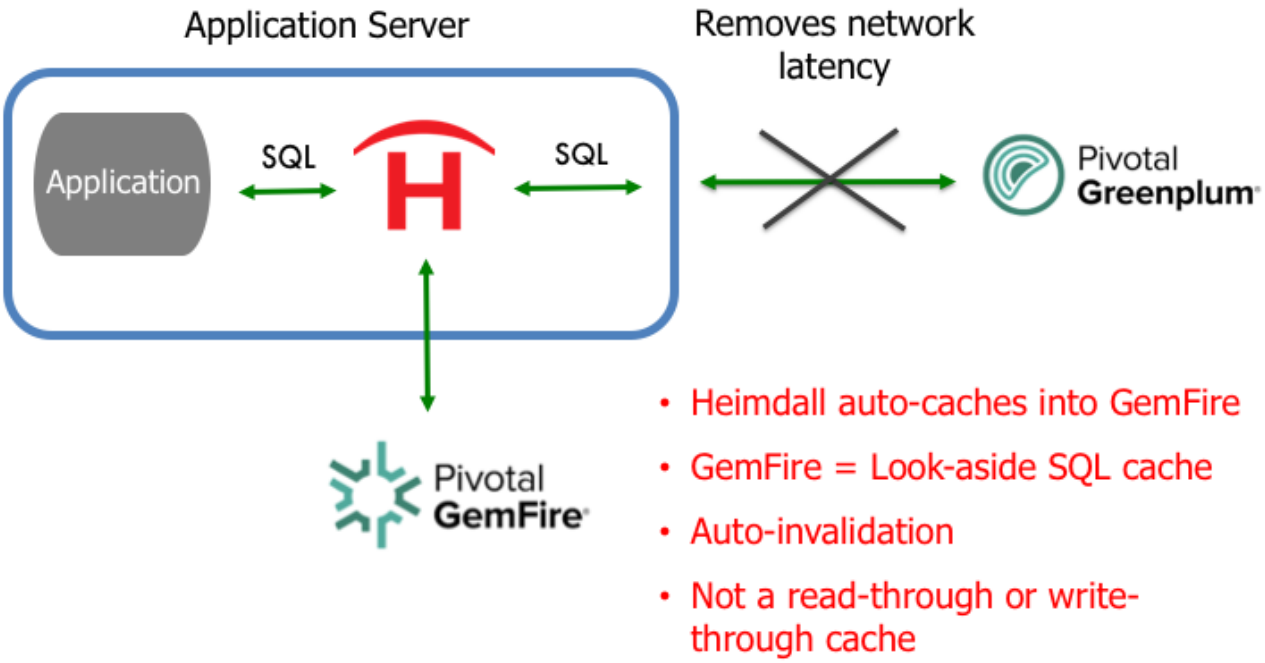
- Queries against a materialized view can be routed to an alternate database (i.e. Postgres), which acts on behalf of Pivotal Greenplum. Postgres answers queries offloading Pivotal Greenplum.
- Heimdall triggers a refresh of the view automatically. Heimdall is aware updated views from Pivotal Greenplum and when data was loaded that may impact the view. The net result is faster reports and a lighter load on Greenplum, allowing the processing of other queries to be faster and more scalable.



Heimdall Data for Pivotal Greenplum

Automated SQL Caching for Pivotal GemFire

The fastest query is the one that does not have to be executed. Heimdall's intelligent auto-caching and auto-invalidation work together with Pivotal GemFire and Greenplum. Our unique architecture consists of a two-tier SQL caching solution: 1) Caching at the application tier for hot objects; and 2) Caching into Pivotal GemFire. Result sets are cached in tandem and are invalidated upon writes to the table. Best of all, Heimdall deployment requires zero code changes.



Heimdall Data is a Pivotal Advanced Technology partner. Our SQL Traffic Manager provides improved SQL performance and reliability Pivotal Greenplum.

The Heimdall architecture was designed for ease of deployment without the need to modify the application or database. Configuration changes are updated at runtime without restarting the application. The net result is a platform that can be updated with 100% application uptime.

For more information about how Heimdall Data can help improve your AWS environment, contact us at info@heimdalldata.com.

