

# High-Performance Transaction Processing in SAP HANA

Presentation by Young-Rae Kim

# What is SAP HANA?

- An in-memory, column-oriented, RDBMS marketed by SAP SE.<sup>[1]</sup>
- 'HANA' is not an acronym.



# What is SAP HANA?

- An in-memory/main memory DB system:
  - Provides high performance without slow disk interactions.
  - Eliminates seek time when querying data.

# What is SAP HANA?

- Column-oriented:
  - Not strictly column-stored (i.e. also has row store).
  - Great for OLAP due to its advantage in aggregate calculations.
    - compare to row-oriented storage which is better for transactional workloads (think: single datasets and highly insert/update-intensive)
  - High potential for compression (great for storing in main memory)

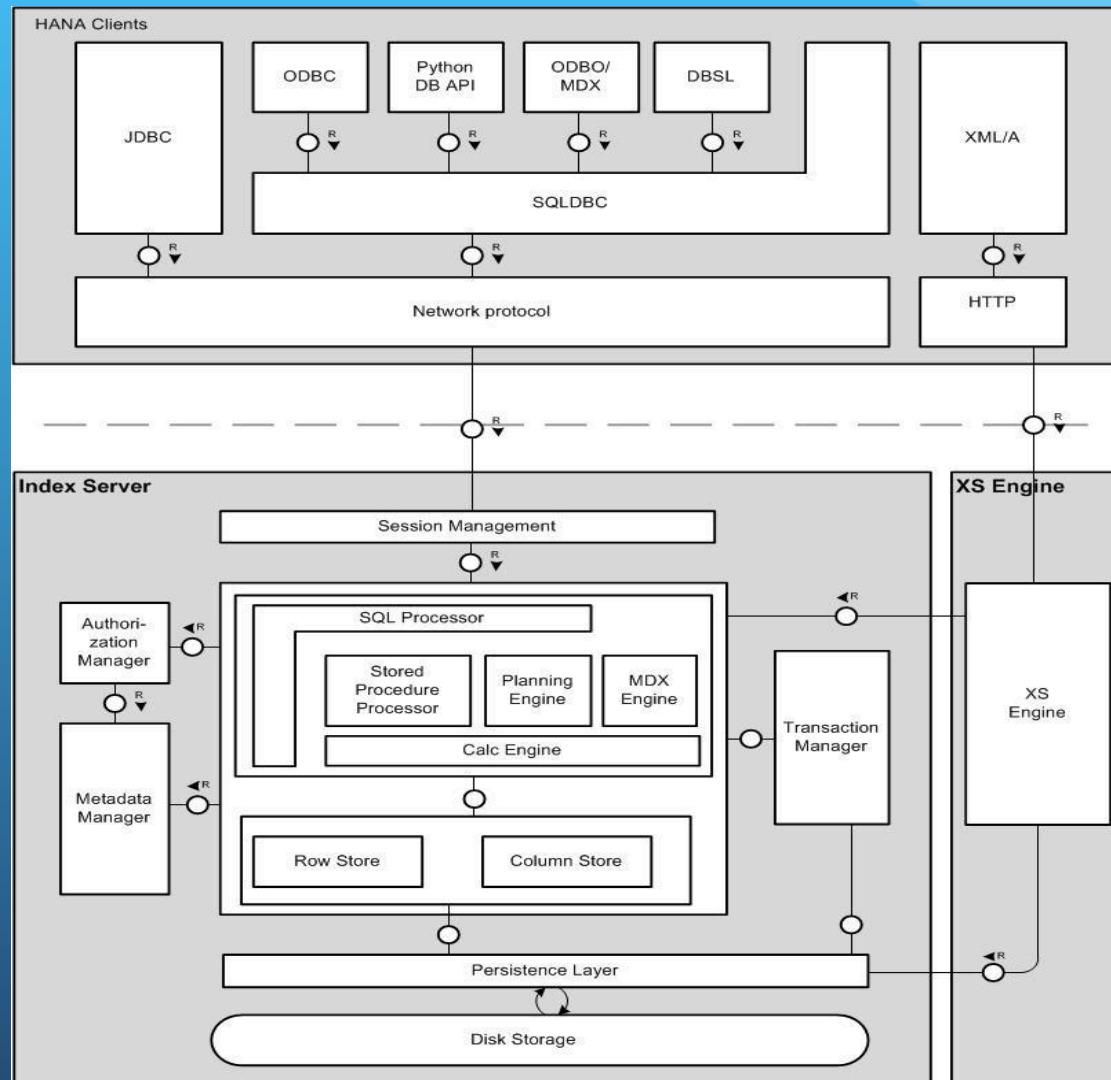
# What is SAP HANA?

## Row Store v. Column Store

Record #	Name	Address	City	State
0003623	ABC	125 N Way	Cityville	PA
0003626	Newburg	1300 Forest Dr.	Troy	VT
0003647	Flotsam	5 Industrial Pkwy	Springfield	MT
0003705	Jolly	529 S 5th St.	Anywhere	NY

Record #	Name	Address	City	State
0003623	ABC	125 N Way	Cityville	PA
0003626	Newburg	1300 Forest Dr.	Troy	VT
0003647	Flotsam	Industrial Pkwy	Springfield	MT
0003705	Jolly	529 S 5th St.	Anywhere	NY

# What is SAP HANA?



# Concurrency Control in SAP HANA

- SAP HANA relies on Multi-Version-Concurrency-Control (MVCC).
  - Snapshot isolation is used to guarantee that all reads made in a transaction will see a consistent 'snapshot' of the database.
  - A central transaction manager generates *transaction tokens* which contain all information needed to construct the consistent view for a transaction.
  - The transaction manager also keeps track of the following for write transactions:
    - Unique transaction IDs
    - Transactional state
    - Commit ID (once committed)

# Optimizations to Achieve High Throughput in SAP HANA

- Distributed Snapshot Isolation Optimization
- Optimized Two-Phase Commit Protocol



# Distributed Snapshot Isolation Optimizations

- “In a distributed environment, ... a worker node should access the transaction coordinator to retrieve its snapshot transaction token.”<sup>[2]</sup> This could lead to:
  1. A throughput bottleneck at the transaction coordinator
  2. Network delay to worker-side local transactions

# Distributed Snapshot Isolation Optimizations

## Solutions:

1. Local (single-node) read-only transactions may run without accessing the global coordinator
2. Local read or write transactions may run without accessing the global coordinator
3. Multi-node write transactions may access the global coordinator only once using Write-TID-Buffering

# Optimized Two-Phase Commit Protocol

## Solutions:

1. The commit log is written to disk following the first commit phase. Second commit phase logging is done asynchronously.
2. Log I/Os is eliminated by skipping prepare-commit log entries. Tradeoff between transactional throughput and recovery time.
3. Group together commit and prepare-commit requests as much as possible.

# Bibliography

- [1]: [http://en.wikipedia.org/wiki/SAP\\_HANA](http://en.wikipedia.org/wiki/SAP_HANA)
- [2]: High-Performance Transaction Processing in SAP HANA. Lee et al. (pg. 4)

# Images (in order)

- <http://forums.bsdinsight.com/attachments/sap-hana-jpg.6725/>
- <http://cdn-s4.tarikmoon.com/wp-content/uploads/2014/05/row-store-v-column-store.gif>
- <http://upload.wikimedia.org/wikipedia/commons/9/9f/Hana.jpg>