



The Snowflake Elastic Data Warehouse

marcin.zukowski@snowflake.net

XLDB 2016
May 24, 2016

Who We Are

- Founded 2012
- Mission: Build an enterprise data warehouse as a cloud service
- HQ in downtown San Mateo
- 130+ employees, ~50 engs (and hiring!)



Our Product

- The Snowflake Elastic Data Warehouse
 - Multi-tenant, transactional, secure, *highly scalable, elastic*
 - Designed from scratch for the cloud
 - Built to provide a true service experience
- Runs in the Amazon cloud (AWS)
- Millions of queries per day over petabytes of data
- 100+ active customers, growing fast



Motivation

Some history

- Late 2012...
- SQL-on-Hadoop is all the hype...
- Redshift isn't around yet...
- Let's not look around. Let's look up...

What is that Cloud thing?



What is that Cloud thing?



Cloud: Your Next Computer

- New computing platform
- New operating system
- Elasticity in multiple dimensions
- Infinite* scalability
- SaaS delivery model
- The data hub for the world

Cloud and Databases?

- Can it work?
 - Sure! Let's deploy MySQL on EC2!
- Can it work well?
 - Elasticity?
 - Resilient to hardware failures?
 - Easy to use?

 - Hmmm....

Shared-nothing Architecture



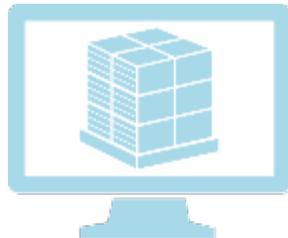
- Tables are horizontally partitioned across nodes
- Scales well for star-schema queries
- Requires a lot of tuning
- Dominant architecture in data warehousing
 - Teradata, Vertica, Netezza...

The Perils of Coupling



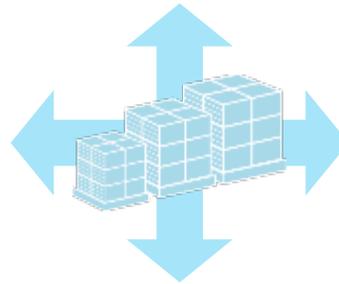
- Shared-nothing *couples* compute and storage resources
- Elastic?
 - Resizing requires redistributing data
 - System often unavailable
 - Cannot disable unused resources → no pay-per-use
 - Impossible to provision correctly
- Homogeneous resources vs. heterogeneous workload
 - Bulk loading, reporting, exploratory analysis

Our Vision for a Cloud Data Warehouse



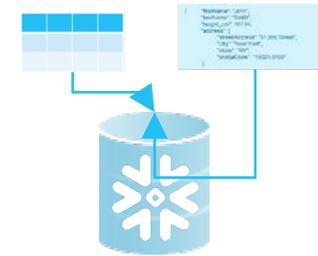
Data warehouse as a service

No infrastructure to manage, no knobs to tune



Multidimensional elasticity

On-demand scalability data, queries, users



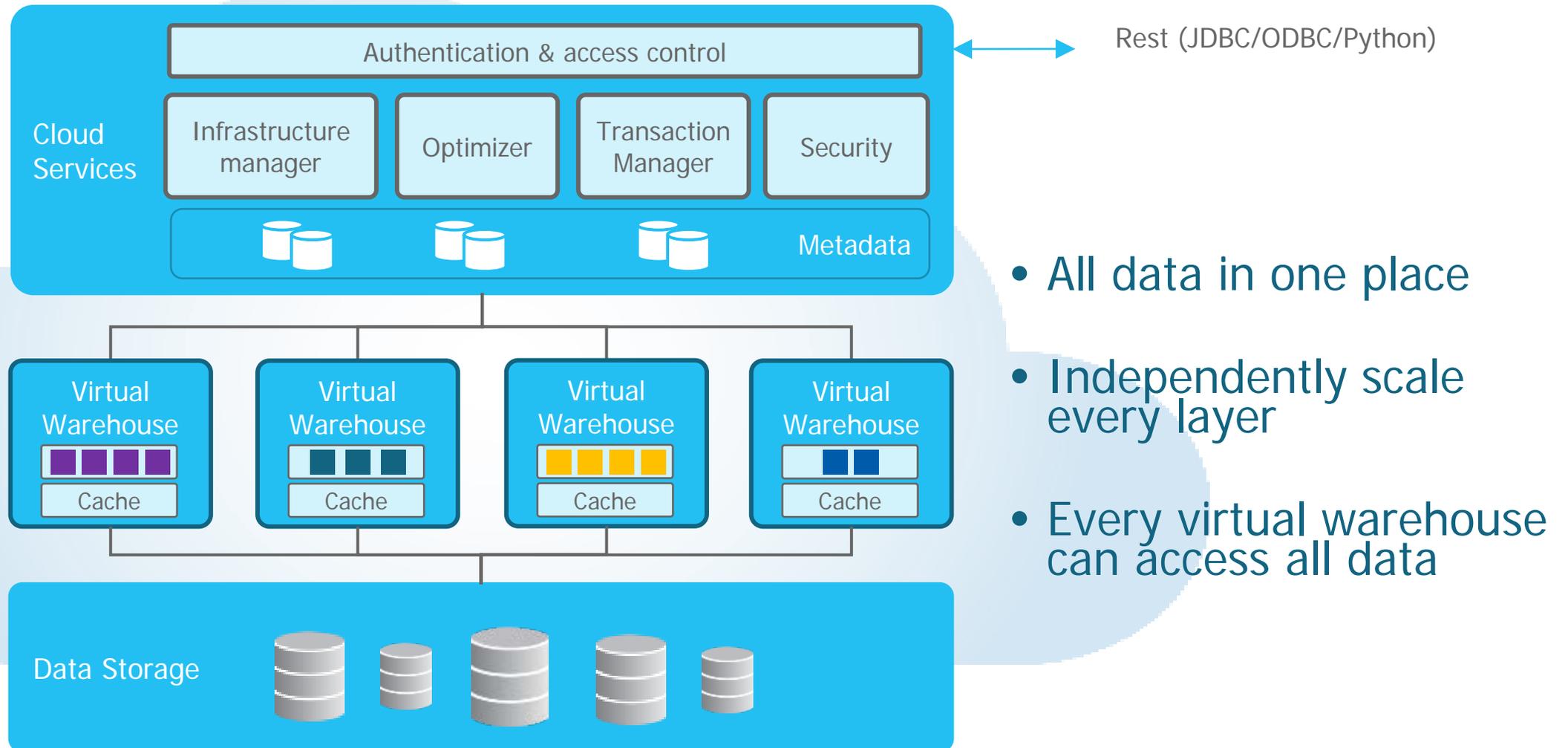
All business data

Native support for relational + semi-structured data



Architecture

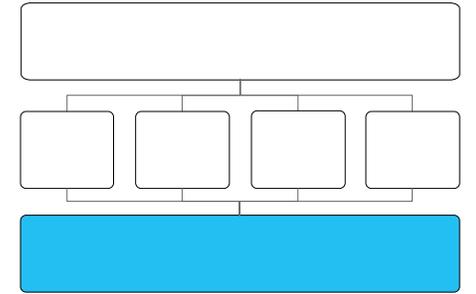
Multi-cluster Shared-data Architecture



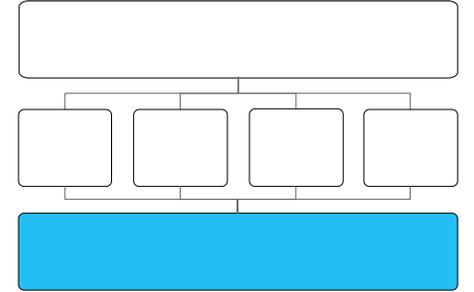
- All data in one place
- Independently scale every layer
- Every virtual warehouse can access all data

Data Storage Layer

- Stores table data and query results
- Uses Amazon S3
 - Object store (key-value) with HTTP(S) interface
 - High availability, extreme durability (11-9)
- Some important differences w.r.t. local disks
 - Performance (sure...)
 - No update-in-place, objects must be written in full
- S3-optimized file format and concurrency control

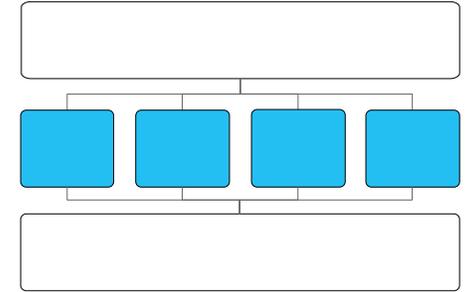


Other Data



- S3 also used for temp data and query results
 - Arbitrarily large queries, *never run out of disk space*
 - Retrieve and reuse previous query results
- Metadata stored in a transactional key-value store (not S3)
 - Mapping of S3 objects to tables
 - Optimizer statistics, lock tables, transaction logs etc.
 - Part of Cloud Services layer (see later)

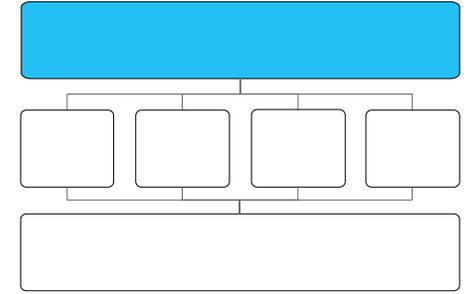
Virtual Warehouse



- Cluster of EC2 instances
- Pure compute resources
 - Created, destroyed, resized on demand
 - Users may run multiple VW at same time
 - *Shared data access* with *isolated performance*
 - Users may shut down *all* VWs when they have nothing to run
- Worker nodes are ephemeral
- Each worker node maintains local table cache

Cloud Services

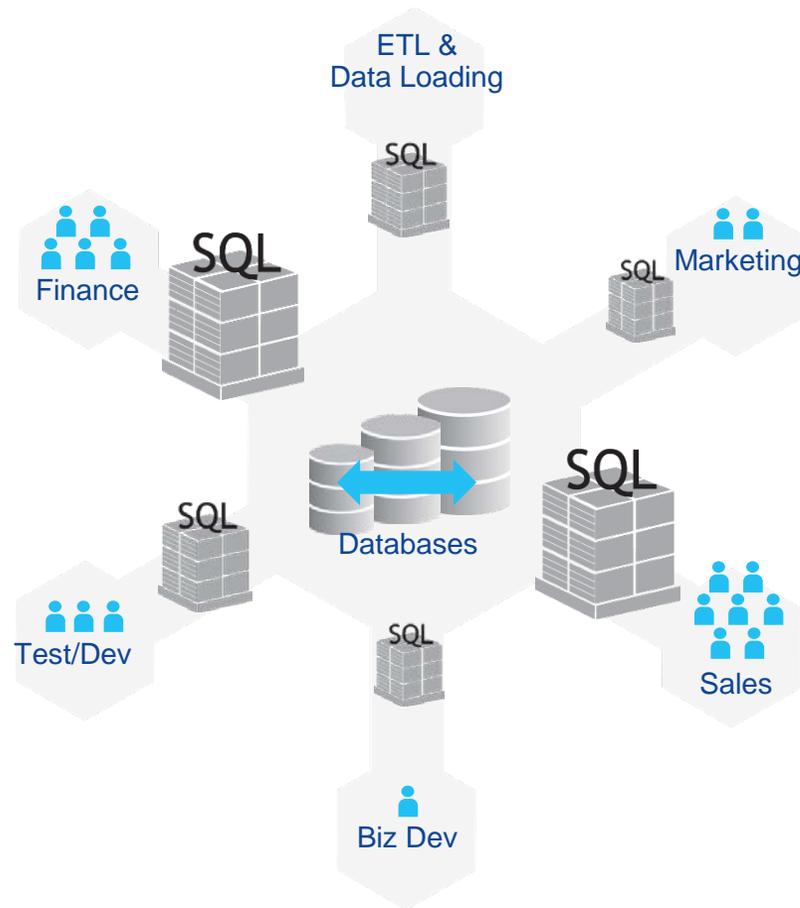
- Collection of services
 - Access control, query optimizer, transaction manager etc.
- Multi-tenant and always on
- Replicated for availability and scalability
- Hard state stored in transactional key-value store
- Standard interfaces and feature-rich web UI
- Focus on ease-of-use and service experience





Feature highlights

Multi-dimensional elasticity



- Elastic scaling for

- Storage
- Compute
- Concurrency

- All thanks to decoupling of storage and compute!

Elastic Storage

- S3: Low-cost, fully replicated, secure and resilient
- Infinite* capacity
- Pay for space/time you use
- All data available to everyone
 - Full transactional consistency
- Requires elastic processing engine

Elastic compute and concurrency

- Optimize Virtual Warehouses for workloads
 - Small VW for continuous loading
 - X-Large VW for once-a-week report
- Optimize for concurrent use
 - Different VWs for different users
 - Access to the same data, no performance interference
 - Automatic scaling for high-concurrency scenarios
- Pay for what you use

New usage scenarios

- “Cheaper than walking to the DBA”
 - Asking DBA for permission takes 10 minutes.
 - Time => Money => Compute (if elastic!)
- “It’s like a Porsche for the weekend”
 - “I use a 64-node machine for my weekly report!”
- No more: “Don’t run queries! We’re loading new data!”
 - No resource/performance interference. No data marts!
- “No tuning, it just works”
 - “I lost 20 pounds and reduced smoking”

Other features

- Multi-AZ deployment
- Continuous availability
- Always up-to-date
- Security (SOC-2, HIPAA)
 - Federated authentication & MFA
 - Access control
- Automated backup
- Automated scalability
- Time travel
- Instant cloning
- Optimized semi-structured storage and processing
 - Matching relational performance
- JavaScript UDFs
- ODBC, JDBC, NodeJS, Python, R, Spark, ...
- Tableau, Informatica, Looker...



Lessons learned

Lessons Learned

- Decoupling storage and compute a game changer for users
 - Maps onto cloud very well
 - Allows a novel multi-cluster, shared-data architecture
 - Fewer data silos and easier data access
 - More flexible use scenarios
 - Scale costs for different layers independently
- Semi-structured extensions were a bigger hit than expected
- SaaS model helps both users and us
- Users love “no tuning” aspect

Ongoing Challenges

- SaaS and multi-tenancy remain biggest challenges
 - Hundreds of concurrent users, some of which do *weird* things
 - Metadata layer is becoming huge
 - Failure handling
- Security
 - There is more to running a secure service than “encrypt everything”
- Lots of work left to do
 - SQL functionality and performance improvements
 - Self-service model

P.S. See you at SIGMOD!



