Goldman Sachs

GS Big Data Platform

Data Philosophy

1 Instrument everything

2 Put all data in one place

3 Data first, questions later

4 Store first, structure later

5 Let everyone party on the data (with controls)

6 Keep raw data forever

7 Produce tools to support the whole research cycle

8 Modular and composable infrastructure

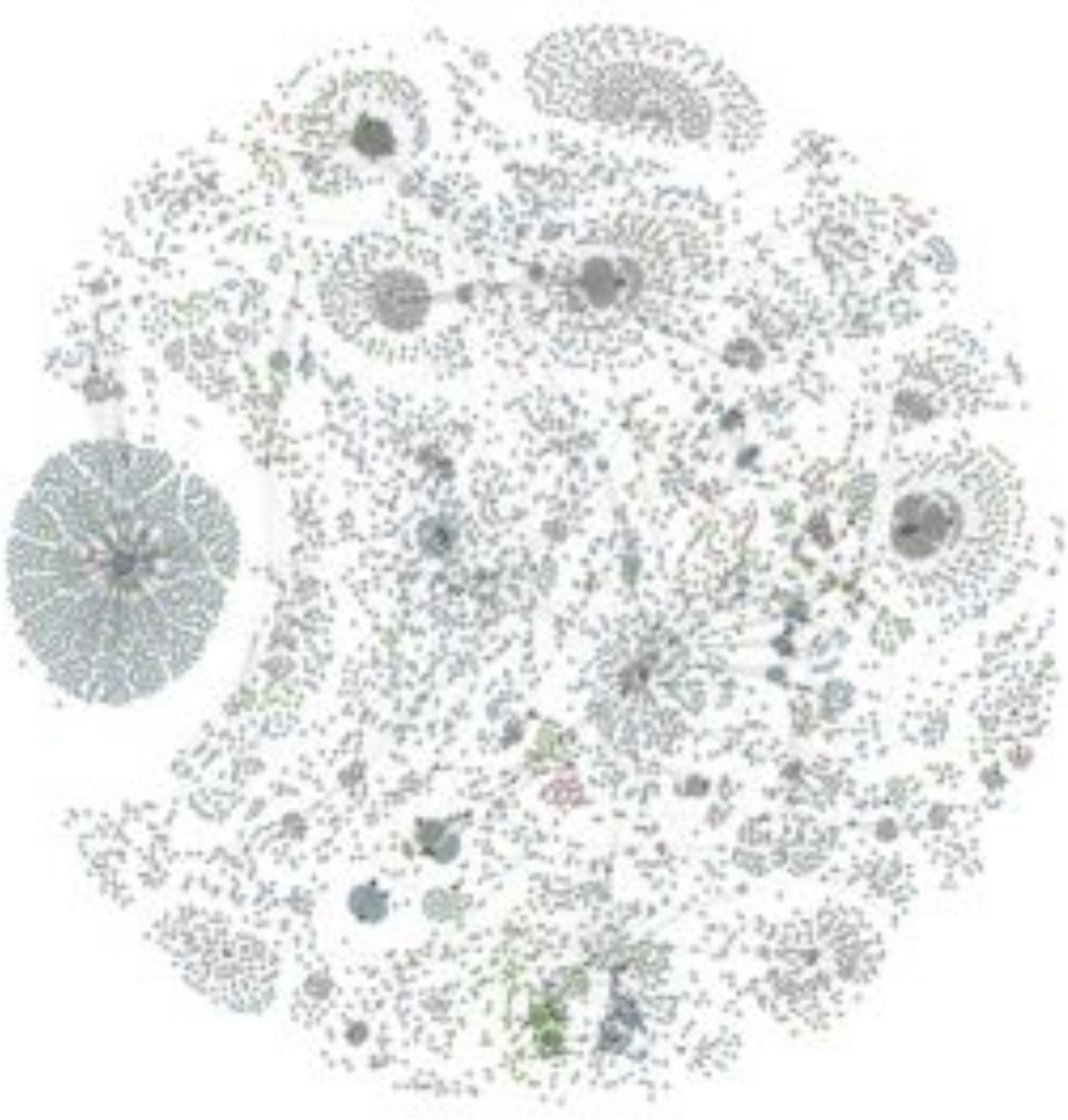
Distributed Systems; Distributed Data

Our ‘Big’ small data problem

- Highly functionally aligned systems
  - Excellent Data Segregation
  - Local Data Autonomy
  - Local Governance & retention
  - Locally negotiated data evolution

- Extensive ~~ab~~use of data movement technologies
  - ‘Shared Data’ (reference data) is broadly disseminated, but mostly from central locations
  - ‘Event’ data (Transactions) flow across systems and persisted at each stage
  - We rarely used centralized shared services like the reference data farm

- Our Data is an ‘*Asset’* and should be treated as such



Big Data Platform Goals

Create a 'GS Data Lake' to allow for many datasets to coexist and be available which is external to any specific GS application.

Creating a data registry to store the dataset metadata and allow for datasets to be discovered and used.

Create a facility to properly entitle access to the datasets (that code is typically custom logic embedded directly within each application)

Create the facilities to ingest the data and provide resiliency.

Build an integrated software stack using multiple data management software products to provide the full suite of function required for the Data Lake.

The platform will be composed of multiple products integrated and made consistent by GS developed infrastructure.

HDFS/Hive for deep petabyte scale online archive

MPP Column-store ParAccel for high performance aggregation/pivoting

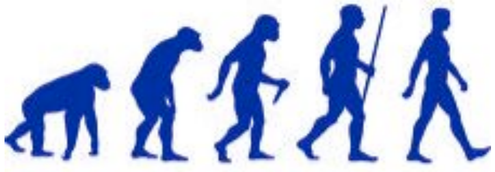
Graph database for non-relational queries and semantic search

Text search function for unstructured or semi-structured datasets

Metadata registry and entitlement model

The evolution of scale-out data management platforms

Perhaps the single biggest factor in enabling Big Data is the rapid innovation occurring for the "scale-out" of data management platforms.



- DBMS runs on single host only (traditional RDBMS)
- DBMS runs on multiple hosts, single copy of data, statically partitioned (DB2 DPF)
- DBMS runs on multiple hosts, two copies of data, statically partitioned (ParAccel)
- DBMS runs on multiple hosts, many copies of data, statically partitioned (MongoDB)
- DBMS runs on multiple hosts, many copies of data, dynamic partioning (H-Base)

Big Data Platform Desired Properties

Scalable

- No fixed upper bound to ultimate dataset size.
- Storage and CPU capacity must be able to be increased in an incremental and linear fashion.
- Platform technology stack should already be in use at larger scale than GS use-case.

Affordable

- Technology hardware stack should be based on a scale-out of commodity components.
- Technology software stack should be based on open source projects.
- Platform should be designed to run on GS Dynamic Compute nodes
- Vendor lock-in for any unique portion of the platform should be avoided when possible.
- Operating cost of platform should be kept to a minimum via low touch infra-structure that self manages.

Trusted

- Entire be resilient to individual component failure not requiring any manual intervention
- Must be easy to both self heal from failure and to scale-out additional capacity
- Must have facilities to allow for authentication, security and access entitlement

Big Data Platform Ongoing Research (1)

Entitlement for Big Data platform datasets

Two different entitlement problems to be solved for:

- How to model the entitlement rules on who should be able to see what data.
- How to implement those rules within the platform.

Products such as sqrrl and Accumulo are being looked at to provide the fine grained access control. Alternatively the rules could be implemented in a GS access layer software

Big Graph

- Graph databases can be powerful, allowing for queries that are difficult to express in SQL.
- Graph databases do not easily lend themselves to data shard'ing and scale-out.
- YarcData Urika product is being looked at for high performance Big Graph solution.
- Aurelius Titan graph database also being tested.

Text Search Data Store / Semantic Search Data Store

- Entitling data stored in an unstructured or semi-structured manner poses new challenges
- Elasticsearch product is used in several different applications within GS
- Attivio product is also in use at GS

Big Data Platform Ongoing Research (2)

Big Data platform - data movement

- Information loaded to the Big Data platform should be considered immutable.
- Data fed into the Big Data platform will need to be stored identically on multiple clusters.
- Gigabus will be instrumental in creating serialized streams of data across the Big Data platform
- Any new data created on the Big Data platform will need to be streamed back into the platform

Big Data platform – data retention

- Traditional concepts such as a ‘database backup’ or ‘transaction log’ need to be completely rethought for the Big Data platform.
- Forcing all data through a product with data retention such as Gigabus should be enforced.
- All products that feed data unto the Big Data platform should have a method of replaying datasets unto the platform on request.

Big Data platform – workload management

- All things being equal a fewer number of clusters is preferable to a greater number of clusters.
- YARN and other technologies are being tested to understand workload management functions
- Hadoop data federation technologies are being tested to bridge multiple products.

Big Data Product Status Q2 2014

GS Big Data Catalog

- Runs on standard Dynamic Compute nodes.
- Utilizes CKAN open source metadata repository application and UI.
- Metadata is externalized in Google DSPL format.
- Entire registry stack can be extended for GS specific requirements.

Hortonworks Hadoop 2.06

- Runs on Dynamic Compute large storage nodes.
- Major engineering effort underway to have full Kerberos integration.
- Standard monitoring to Fabric with 24x7 support team.
- HDFS file based technologies such as M-R, PIG and Hive currently used in production.
- H-Base key value database currently used in production.
- Site resiliency and data retention will not be provided via the Hadoop stack

ParAccel Relational DBMS

- Runs on Dynamic Compute large storage nodes.
- Mature high performance MPP columnar RDBMS.
- Cluster is inelastic and does not keep multiple copies of data.

Appendix

Using R: Statistical Data Analysis

You may have heard of….

Predictive Analytics, Data Mining, Data Analysis, Statistical Analysis, Data Visualization, Business Intelligence, Big Data

Who uses it? (who doesn't?)

Google, Facebook, Double-Click, LinkedIn

Credit Card Companies, Insurance Companies, Finance (of course)

Anywhere you want to extract value from your data

Development Patterns

Data Visualization

Graphing statistical summaries of data to gain insights

Modeling and Prediction

Model the system using statistical models, then use those models to check new data

Data visualization used to understand how the model performs

R is a great way for programmers to do statistics