AllegroGraph

a graph database

FRANZ INC.
“WEB 3.0's DATABASE”

Gary King
gwking@franz.com

Overview

- What we store
- How we store it – the possibilities
- Using AllegroGraph

Databases

- Put stuff in
- Get stuff out

- quickly
- safely

Stuff

- things with attributes *and connections*
- Reasoning, rules, inference

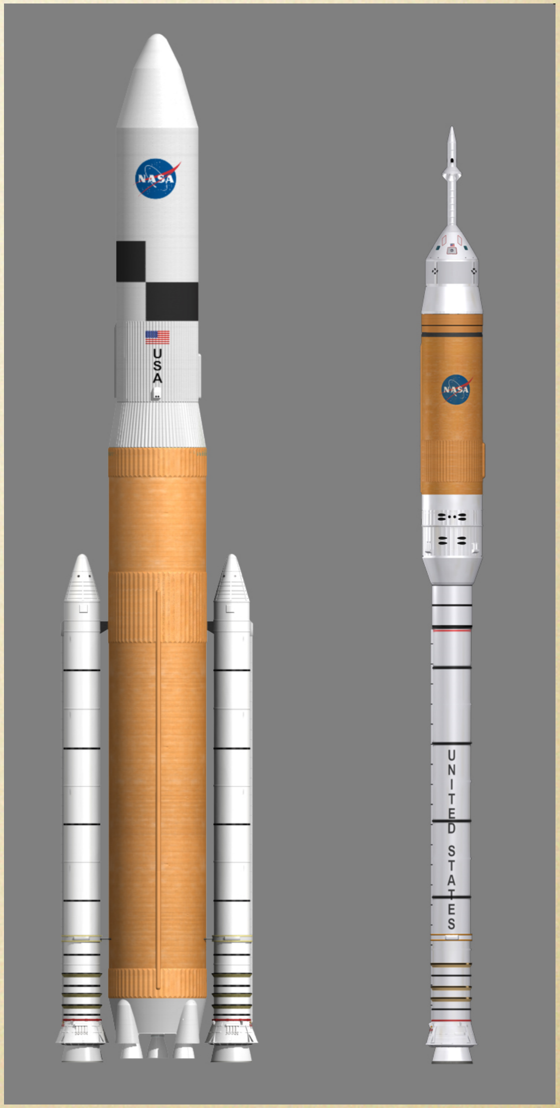
- Lots of things. Really. Lots.
- Lots of change. all the time.

What stuff in?

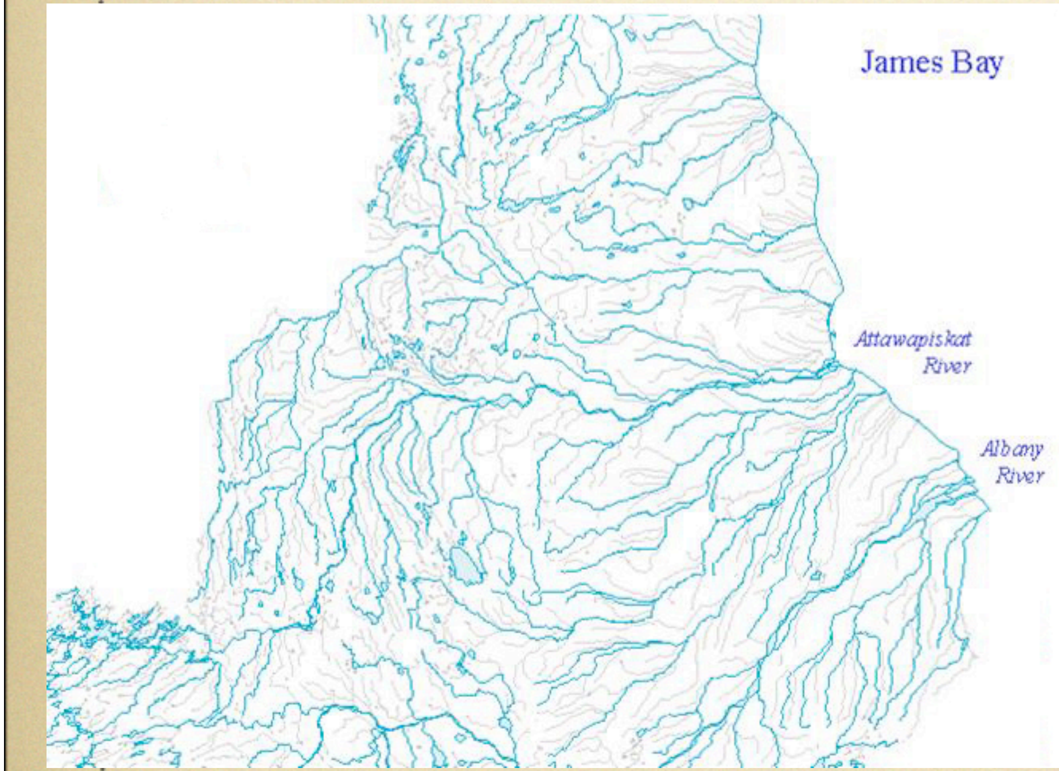
- Modeling knowledge of assets in an Enterprise
- Modeling an extensive river network
- Representing 1000's of different types of objects
- Managing biological knowledge
- Multimedia Metadata
- Bug and version tracking
- Collaborative Workspace for Analyst

NASA Constellation project...

- Deals with 1000s of different types of objects:
  - Machine parts
  - Processes
  - Software
  - People skills
  - Drawings
  - Documents
- In 100s of distributed databases
- Coordinated through registries
- To provide meaningful search

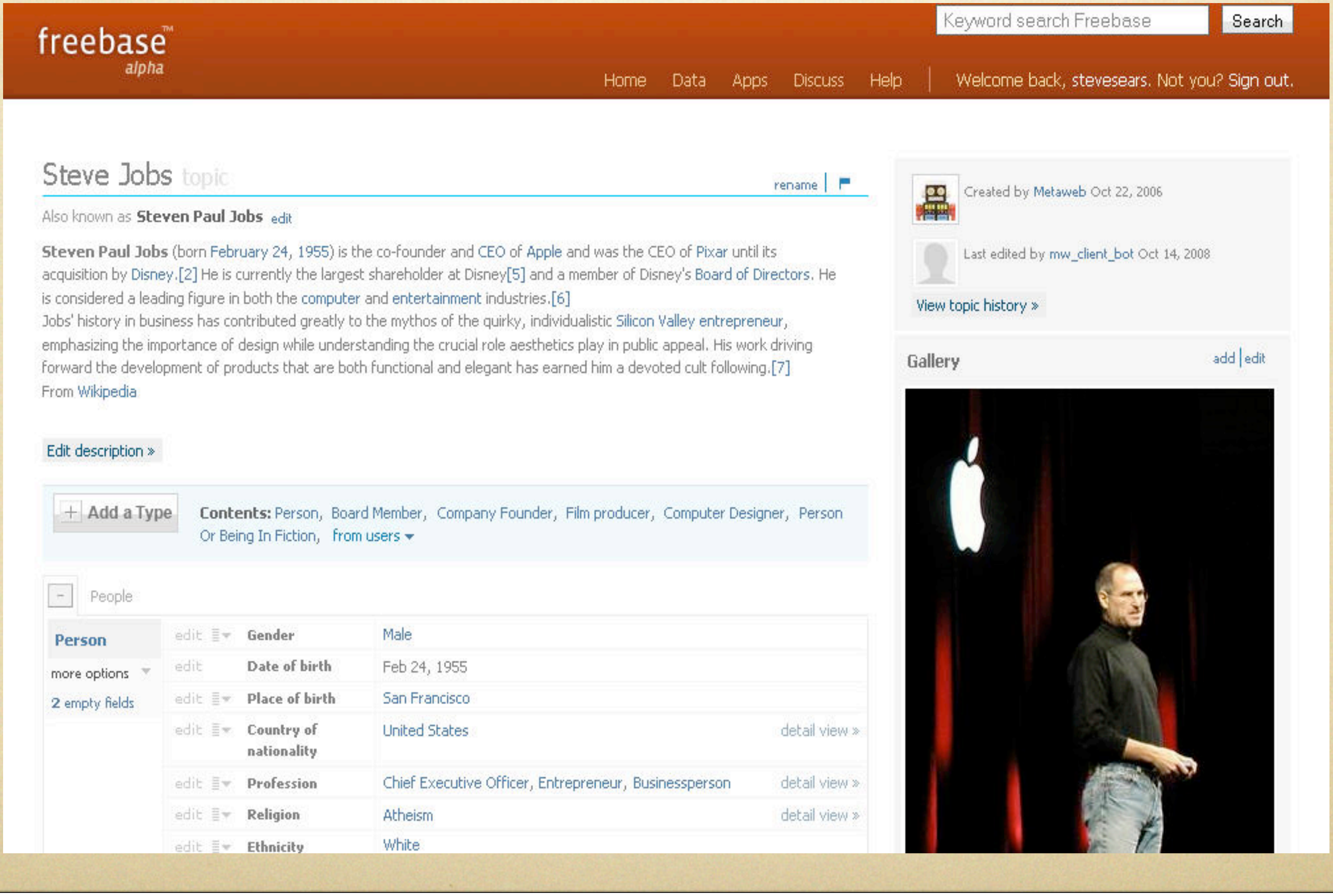


A River Network



- Given the polluted segment S1 find all the upstream segments within 50 miles of City1200
- Given the polluted drainage D1 find all the schools in the rectangle <x1, y1, x2, y2> that might be influenced

Semantic Web…



Welcome back, stevesears. Not you? Sign out.

Steve Jobs topic

Also known as **Steven Paul Jobs** edit

Steven Paul Jobs (born February 24, 1955) is the co-founder and CEO of Apple and was the CEO of Pixar until its acquisition by Disney.[2] He is currently the largest shareholder at Disney[5] and a member of Disney's Board of Directors. He is considered a leading figure in both the computer and entertainment industries.[6]
Jobs' history in business has contributed greatly to the mythos of the quirky, individualistic Silicon Valley entrepreneur, emphasizing the importance of design while understanding the crucial role aesthetics play in public appeal. His work driving forward the development of products that are both functional and elegant has earned him a devoted cult following.[7]
From Wikipedia

Edit description »

+ Add a Type

Contents: Person, Board Member, Company Founder, Film producer, Computer Designer, Person Or Being In Fiction, from users ▾

− People

Person

more options ▾

2 empty fields

Gender	Male	
Date of birth	Feb 24, 1955	
Place of birth	San Francisco	
Country of nationality	United States	detail view »
Profession	Chief Executive Officer, Entrepreneur, Businessperson	detail view »
Religion	Atheism	detail view »
Ethnicity	White	

Created by Metaweb Oct 22, 2006

Last edited by mw_client_bot Oct 14, 2008

View topic history »

Gallery

What stuff out?

- Things like *this*
- Things like *this* only with *that*
- Things like *this* only with *that* and the other thing sorted by *that*
- Things like *this linked* to that *linked* to that *linked* to *that* and *that* and back to things like *this*
- Things like *this* where *that* can be *inferred* from *this* other stuff

In particular

- We want to ask for
  - What – Attributes
  - Where – geospatial
  - When – events and temporal logic
  - Whom – Social networks

Find the people I know that share my taste and have traveled to Hawaii during the last year?

Data – Dissected

- Documents (unstructured – mostly)
- Key/value
- subject/predicate/object
- Tuples (by row, by column)

System of Analysis

- The main data is stored safely away somewhere else
- Batch & Bulk oriented loads
- Materialize types and other inferences
- Do queries & analysis
- Few simultaneous users

System of Record

- Data changes on a second to second basis
- You care about the long time persistence of the data
- You care about transactions and recoverability
- You care about concurrent access
- You care about continuous querying and instant reasoning

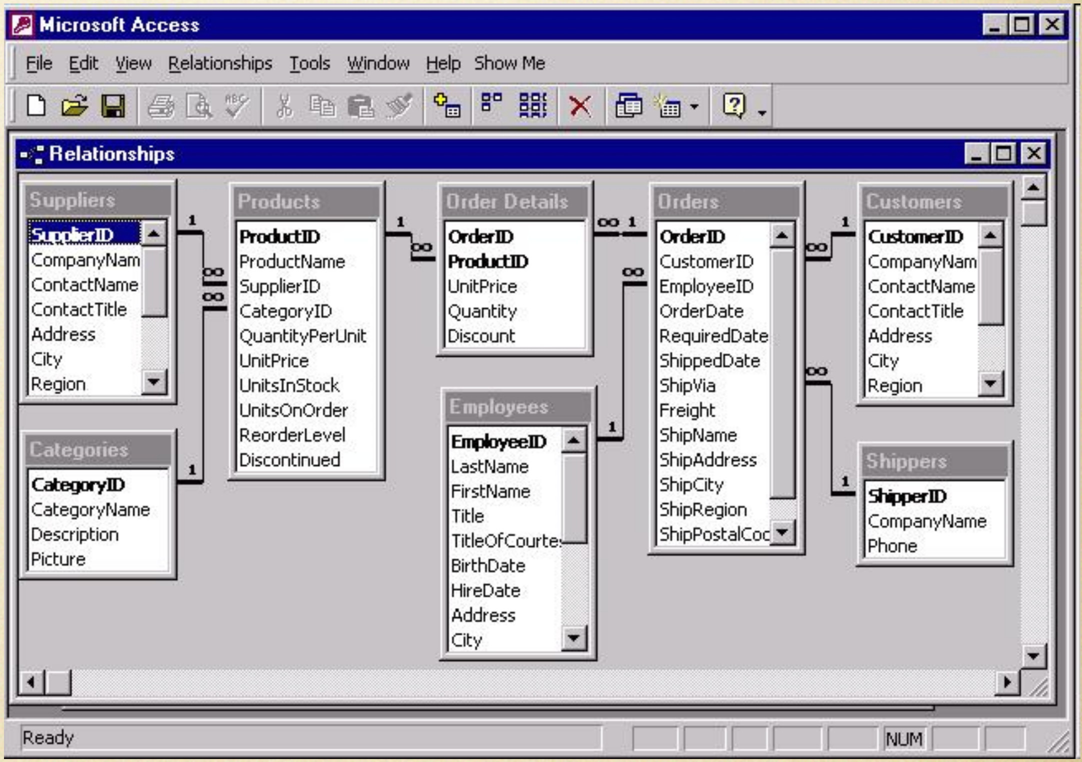
How?

- Relational Database Systems
- Object Oriented Databases
- Key-value Databases
- Graph databases (Triple Stores)

- Essentially equivalent; the devil is in the details.

RDBMS

- Tables
- Columns
- Indices
- Joins

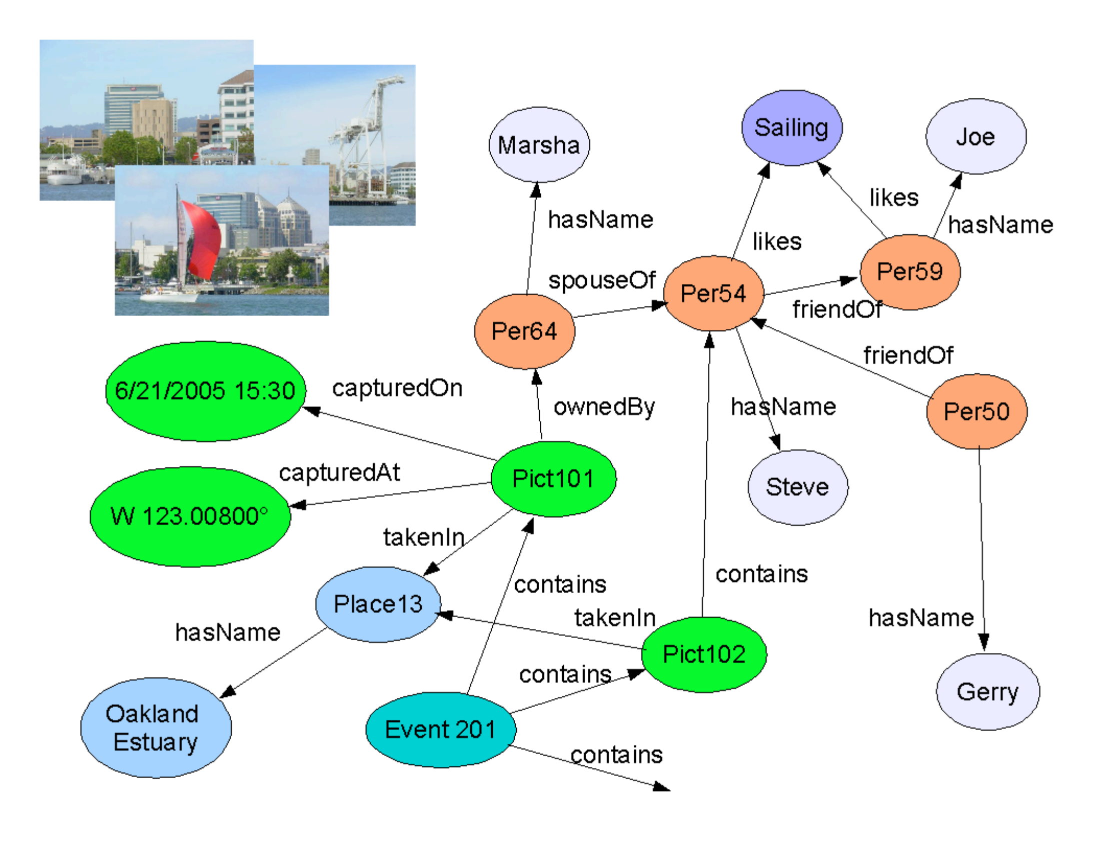


RDBMS

- Mature and Standardized (SQL)
- Robust, safe, scalable
  - Great for simple queries that touch only a few tables once
- But...
  - Modeling the world in tables is hard
  - Table schema is inflexible; early design lock-in
  - One-to-many and many-to-many relationships add extra tables
  - Lousy for queries that follow transitive relationships across many tables (or the same table many times)

Graph DBMS

- Subject – Predicate – Object



GDBMS

- Easy to put stuff in
- No Schema, everything indexed
- But...
  - Young technology
  - Less robust, less standardized

Our problem

- Continually accrue massive interconnected information with an evolving schema (or no schema) including text, events, relationships, locations
- Query this data using description logics, custom rule sets and ask for information on moving objects, events, and social networks, in real-time

In particular

- We want to ask for
  - What – Attributes
  - Where – geospatial
  - When – events and temporal logic
  - Whom – Social networks

Find another truck that can pick up package *X* at location *Y* so that I can pick up package *A* at location *B* so that we both will arrive at *P* before time *T*.

RDBMS is not the answer

- A graph database looks like an relational database with only one table so start with an RDBMS and add triple-store features
- The relational model is *too* complex for triple-stores
- The relational model is *too* simplistic for rapidly evolving schemas and massive transitive relations

Hadoop is not the answer

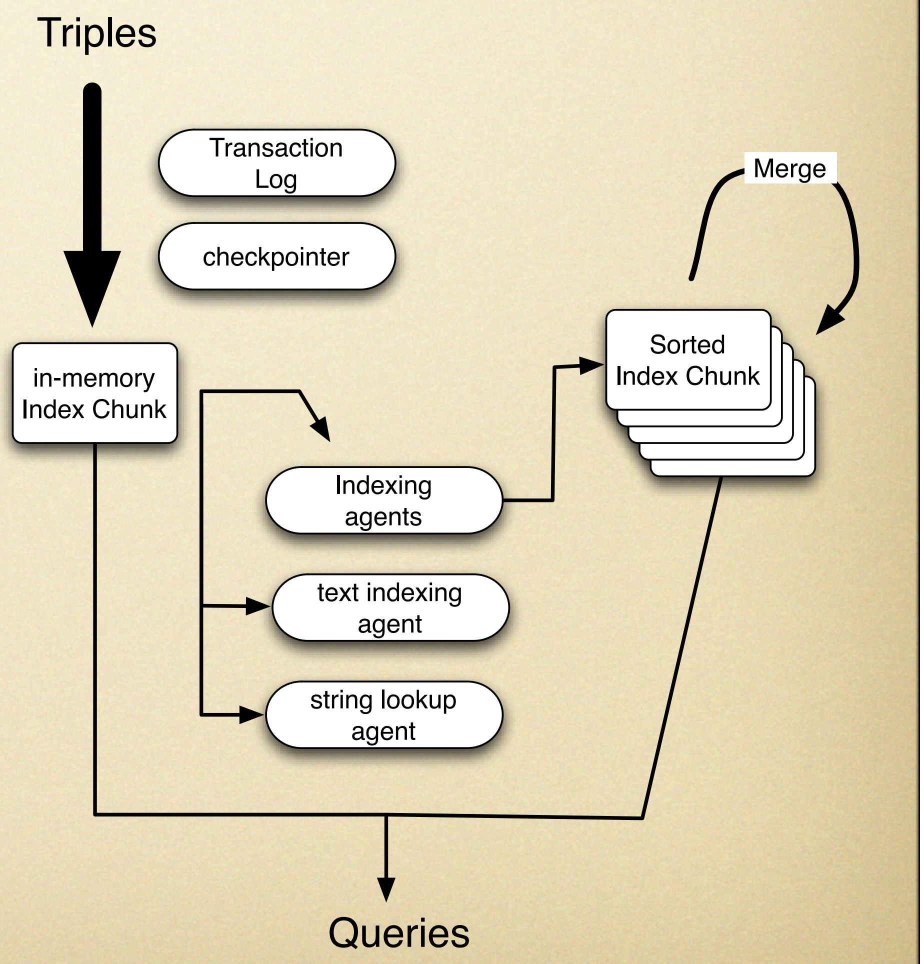
- Yes, it is a great way to store billions of triples
    - Hadoop can be used for work that is batch-oriented rather than real-time, very data-intensive, and parallelizable.
- But what about
    - Deeply nested SPARQL or rule based queries (e.g., Prolog)
    - Graph & Social network analysis.
    - Reasoning and inference

Building a triple-store

- Start fresh and add *enterprise* features
- adding *triples* (with five parts)
- Emphasis is on addition (not updates, not deletion)

Really Simple Diagram

- Triples In
  - processes to
    - index
    - merge
    - text index
    - process strings



AllegroGraph 4.0

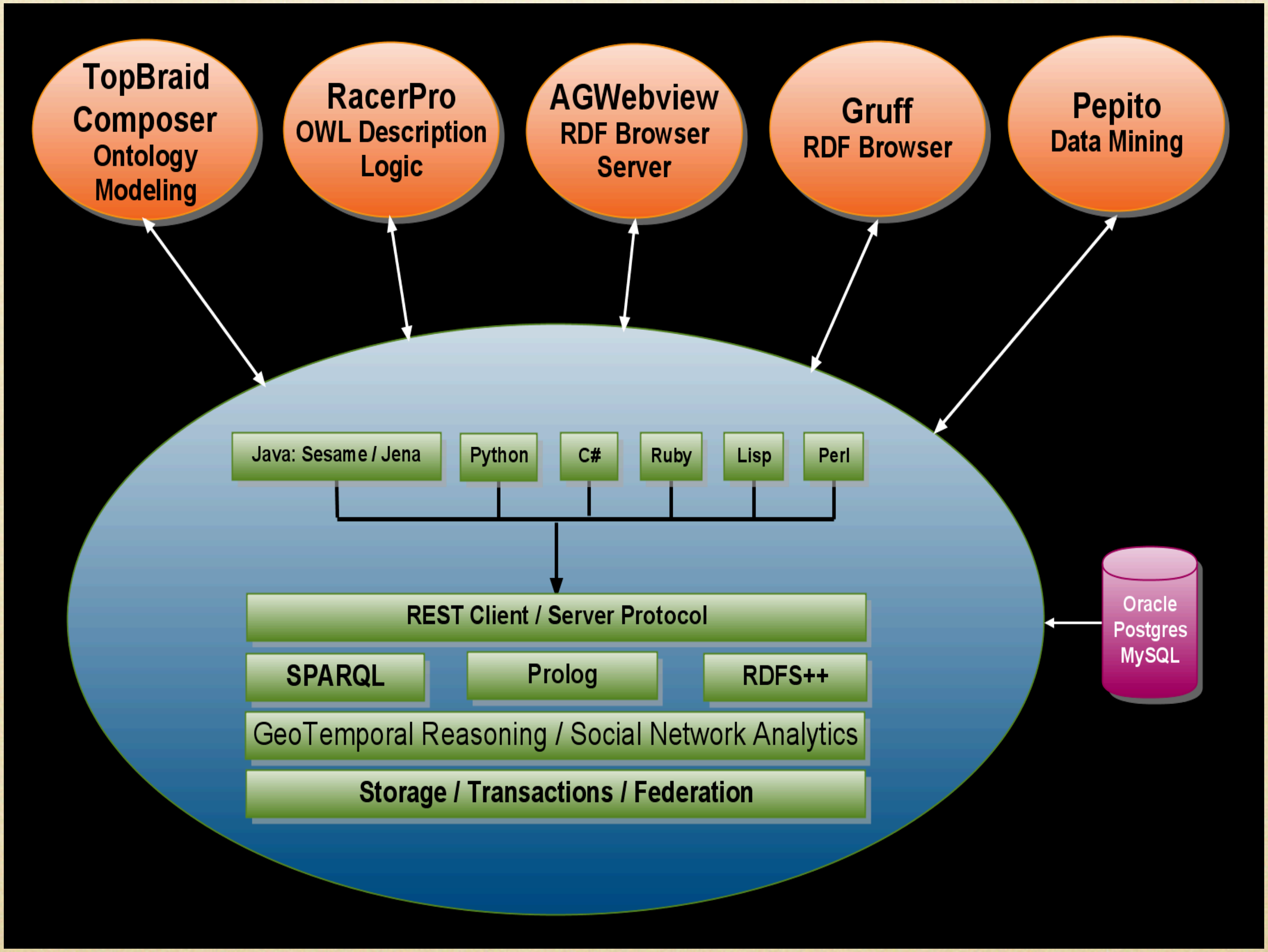
- ACID Transactions and Recoverability
  - page management
  - checkpointing every *x*-minutes or *y*-triples
- Read / write concurrency
  - 100 % read concurrency at all times
- Dynamic and automatic indexing
  - with column based compression
- Resource management
  - Use all disks, all memory and all processors (one box)
  - Automatic, or user configurable

AllegroGraph 4.0

- Per-predicate *Lucene* style text indexing
- 2D and 3D geo-temporal indexing for moving objects
- Social networking toolkit with path finding, importance measures, etc.
- REST protocol for all client interaction
  - Franz supported: Sesame, Jena, Python,
  - Community supported: Ruby, Perl, C#

2D and 3D details

AllegroGraph


TopBraid Composer Ontology Modeling
RacerPro OWL Description Logic
AGWebview RDF Browser Server
Gruff RDF Browser
Pepito Data Mining
Java: Sesame / Jena
Python
C#
Ruby
Lisp
Perl
REST Client / Server Protocol
SPARQL
Prolog
RDFS++
GeoTemporal Reasoning / Social Network Analytics
Storage / Transactions / Federation
Oracle
Postgres
MySQL

Performance: Input

- $5000 quad-core machine with 32 Gigabytes RAM

dataset	Size (Billions)	Time
LUBM 8000	1.1	3:48
Billion Triples Challenge	1.15	5:13
2000 Census data	0.99	2:00
	3.2	11:01

- *with* full-text indexing on all strings

Thanks

gwking@franz.com

http://www.franz.com

FRANZ INC.

"WEB 3.0's DATABASE"