bigdata ™

Flexible
Reliable
Affordable
Web-scale computing.

SYSTAP, LLC

Background

- Requirement
  - Services for fast analytic access to massive, heterogeneous data
- Traditional approaches
  - Relational
  - Super computer
  - Business intelligence tools
  - Semantic web or graph based
- New approach
  - Evolution lead by internet companies
  - Linear scale out on commodity hardware
  - Higher concurrency, up time; lower costs.

The Information Factories

http://www.wired.com/wired/archive/14.10/cloudware.html

- Google
  - Massively concurrent reads and writes with atomic row updates and petabytes of data.
  - Basic data model is
    - { application key, column name, timestamp } : { value }
- Yahoo/Amazon
  - Cloud computing and functional programming to distribute processing over clusters (Apache Hadoop, Amazon elastic computing cloud).
- e-bay
  - Partitioned data horizontally and into transactional and non-transactional regimes.

Use Cases

- Online services
  - Search (Google), Advertising (Google, AOL), Recommendation Systems (Amazon), Auctions (e-bay), Logistics (Wal-Mart, FedEx)
- Federal Government
  - OSINT
  - Predict and interdict
  - Information fusion
  - …
- Bond trade analysis
- Bioinformatics

OSINT

- Harvest OSINT to support IC mission
  - Many source languages
  - Directed and broad harvest
  - Long standing and ad hoc analyses
  - Entities of interest automatically tagged
  - Knowledge captured and republished using a semantic web database
  - Aligned to multiple ontologies.

- Challenges
  - Scale deployment from department to enterprise.
  - Continued data growth and increasing growth rate.
  - Federated policy-based information sharing at scale.

Connect the actors

- Multi-agency database of names, events, relationships.
  - Data quality issues (alternative spellings, timeline, provenance)

- Key computation
  - Preempt actors by linking entities and events to predict and interdict enemy operations.

Bond Trade Analysis

- Post-trade analysis for friction
  - Data are regular (well-structured)
  - Analysis only on historical data
  - Data quality requires real-time updates
  - Data volume grows with market consolidation

- Key computation
  - Dynamically compute face of a hypercube
    - Running in 15-20 seconds on $1M specialty hardware
    - Need to get that down to under 1 second.

Desired solution characteristics

- Cost effective, scalable access to and analysis of massive data.
  - Petabyte scale
  - Linear scale out
  - Support rapid innovation
  - Low management costs
- Very high:
  - parallelism
  - concurrency
  - resource utilization
  - aggregate IO bandwidth

bigdata

architecture

Solution Overview

- Plug-and-play
- Commodity hardware
- Choice of data models
- Service Oriented Architecture
- Self-healing with automatic fail-over
- Directly manage latency, risk and costs

Expect Failure

- Disk, service, machine and network failures are common in a large data center.
- Data must be redundantly available.
- Machines must be commodity items.
  - Add and removing hardware must be easy.
- System must be robust to multiple failures.
- System must support partial outages.

Architecture layer cake

services

- Search& Indexing
- Events
- Data Analysis
- Pub/Sub
- SLA Reports

data layer

- GOM (OODBMS)
- RDFS
- Sparse Row Store

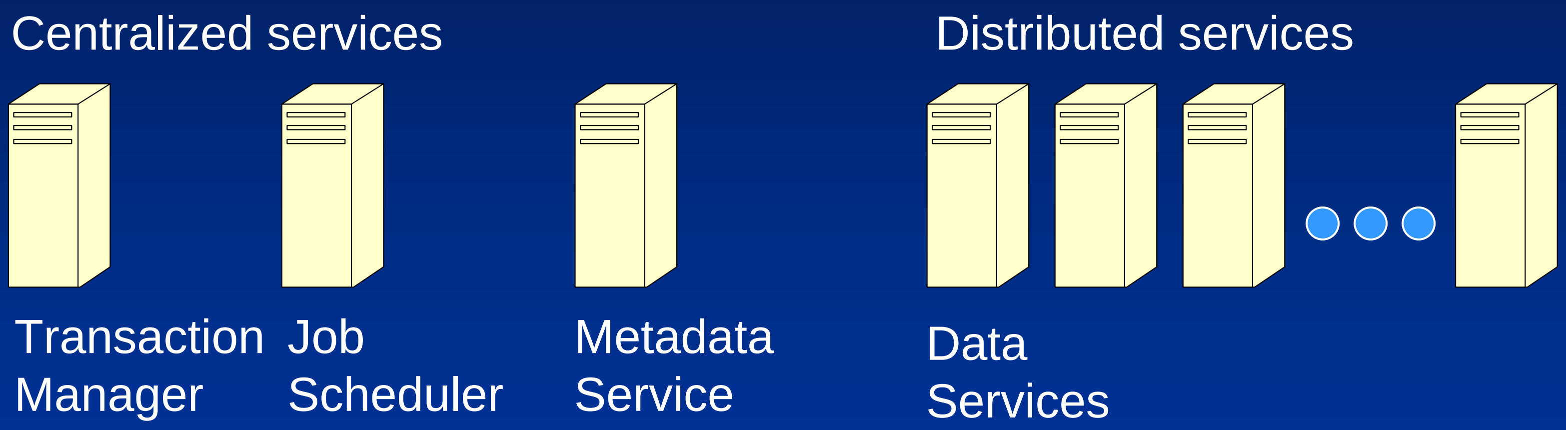
bigdata storage and computing fabric

- Transaction Manager
- Job Scheduler
- (Metadata Service
- Distributed Data Service)

bigdata

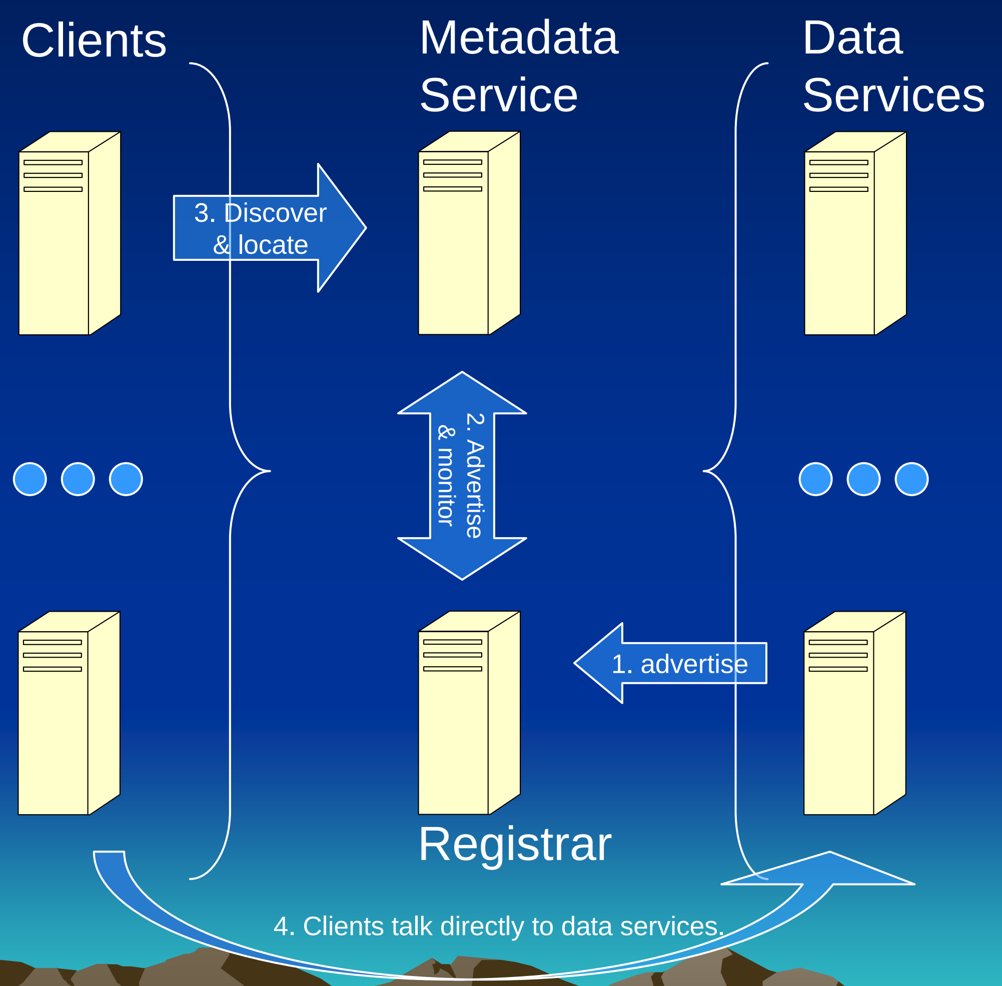
Services all the way down

Distributed Service Architecture



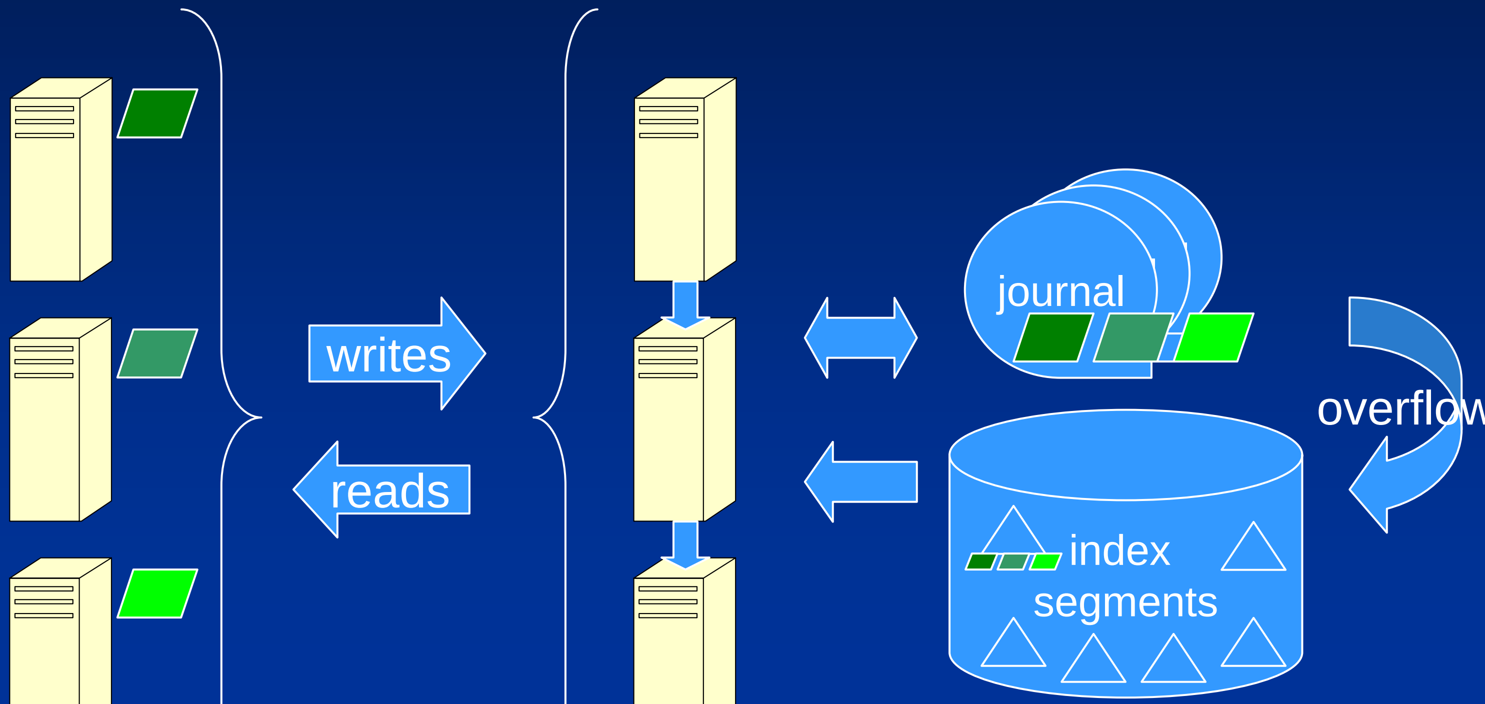
- Grid-enabled cluster.
- Service discovery using JINI.
- Data discovery using metadata service.
- Automatic failover for centralized services

Service Discovery



1. Data services discover registrars and advertise themselves.
2. Metadata services discover registrars, advertise themselves, and monitor data service leave/join.
3. Clients discover registrars, lookup the metadata service, and use it to locate data services.
4. Clients talk directly to data services.

Dynamic Data Layer

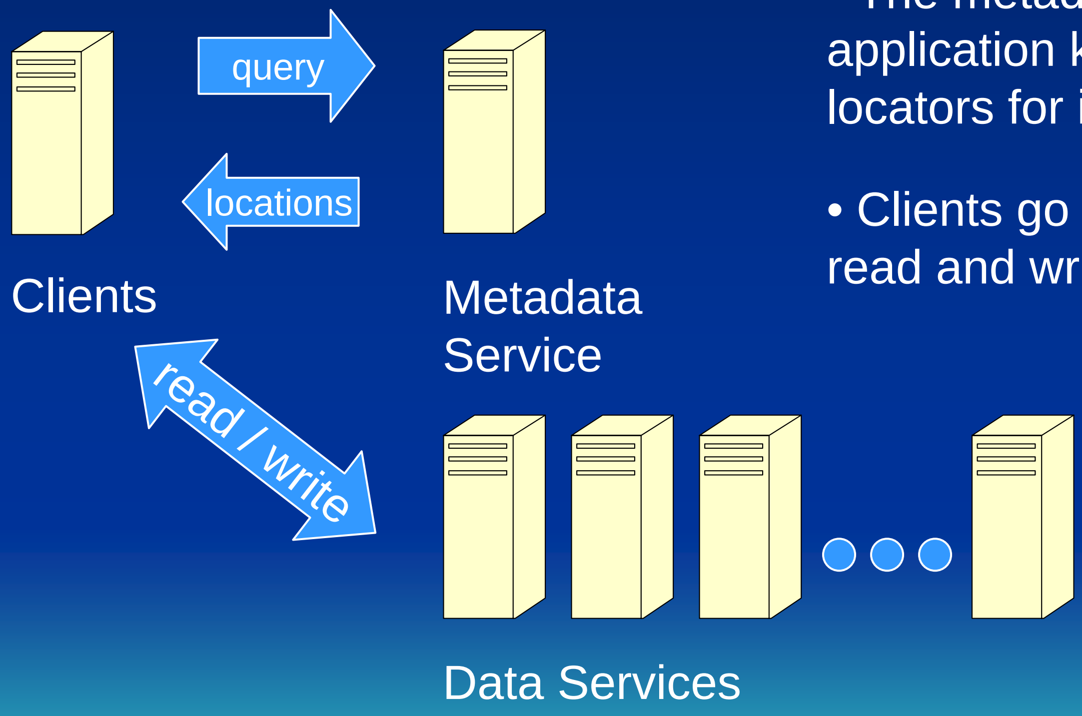


Append only journals and read-optimized index segments are basic building blocks.

Data Service

- Batch B+Tree operations
  - Insert, contains, remove, lookup
  - Extensible operators
- Range query
  - Fast key range scans
  - Optional filters
- Submit job
  - Run procedure in local process

Metadata Service



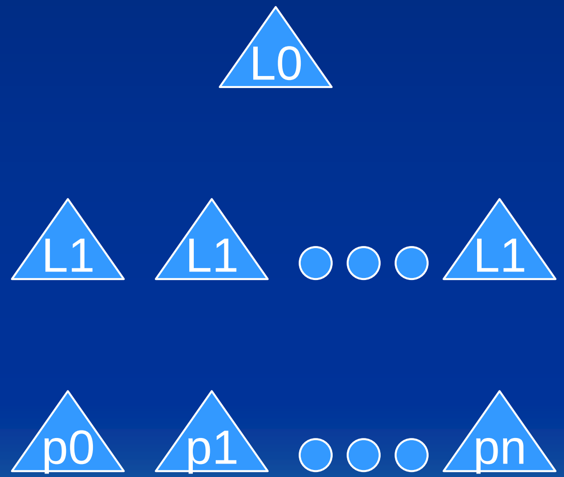
- One metadata index per scale out index.
- The metadata index maps application keys onto data service locators for index partitions.
- Clients go direct to data services for read and write operations

Metadata Service

- Index management
  - Add, drop, provision.
- Index partition management
  - Locate
  - Split, Join, Compact
  - Resource reclamation

Metadata Addressing

- L0 alone can address 16 Terabytes.
- L1 can address 8 Exabytes *per* index.



L0 metadata — 128M L0 metadata partition with 256 byte records

L1 metadata — 128M L1 metadata partition with 1024 byte records.

Data partitions — 128M per application index partition

bigdata

Data Model Tradeoffs

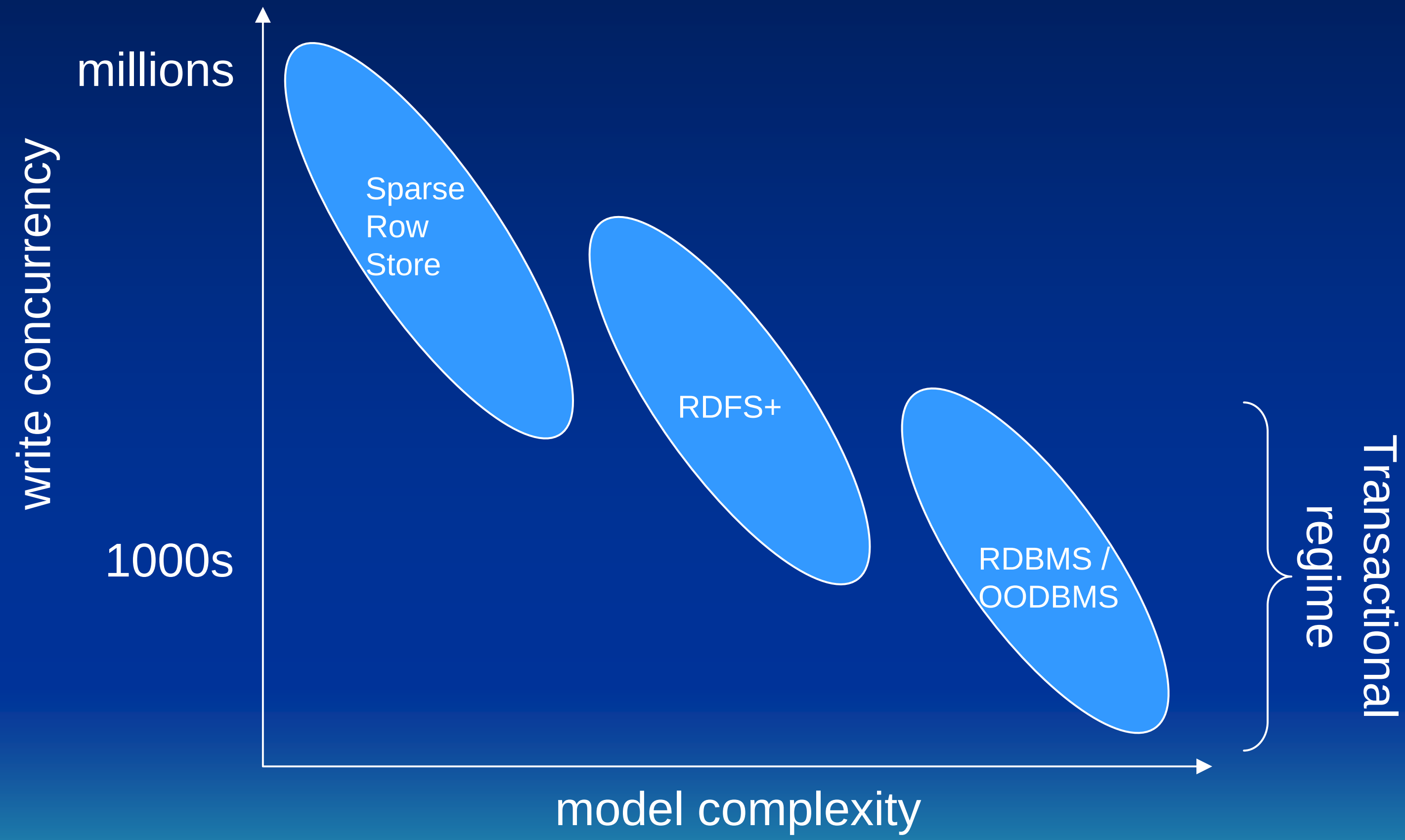
Concurrency Strategies

- MVCC
  - Fully isolated transactions
  - Read-only transactions
  - Read-committed transactions
- Atomic row updates
  - Very high concurrency
  - ACID guarantee for single index, local data only.

Data Models

- Three class of data models
  - OODBMS
    - Object Model
    - Transactional isolation
  - Semantic Web
    - Semantic alignment
    - Declarative query languages
  - Sparse Row Store
    - ACID row operations.
    - Extremely high concurrency.

Trade offs in data models



bigdata

Federation

And

Semantic Alignment

Semantic Web Technology

- Semantic Web Technologies have been applied extensively in federating data within the IC.
  - Well suited to dealing with problems of federation and semantic alignment.
  - Properties such as owl:equivalentClass, owl:equivalentProperty, and owl:sameAs allow for dynamic declarative mapping of classes, properties and instances to one another.

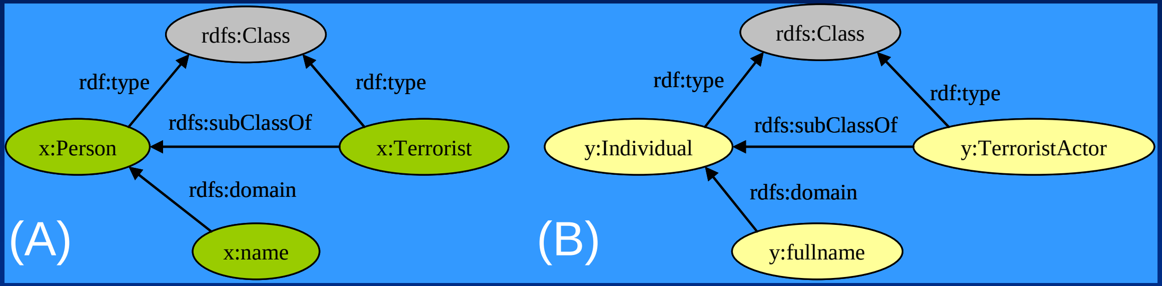
RDF Data Model
Resources

- URIs, “http://www.systap.com”.
- Literals
  - Plain literals, “bigdata”.
  - Language code literals, “guten tag”:“de”
  - Datatype literals: “12.0”: “xsd:double”
  - XML Literals
- Blank nodes
  - An anonymous resource that can be used to construct containers.

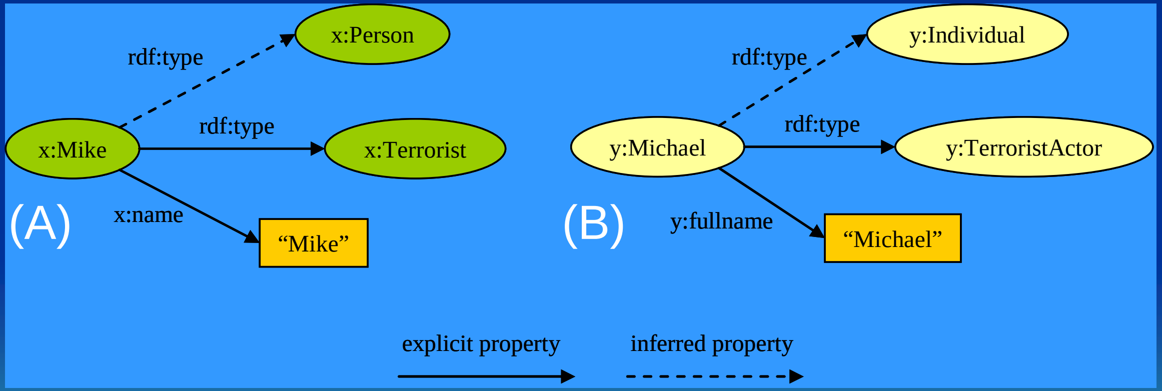
RDF Data Model Statements

- General form is a statement or “assertion”
  - { Subject, Predicate, Object }
  - x:Mike rdf:Type x:Terrorist.
  - x:Mike x:name “Mike”
  - There are constraints on the types of terms that may appear in each position of the statement.
  - Model theory licenses “entailments” (aka inferences).

Semantic Alignment with RDFS

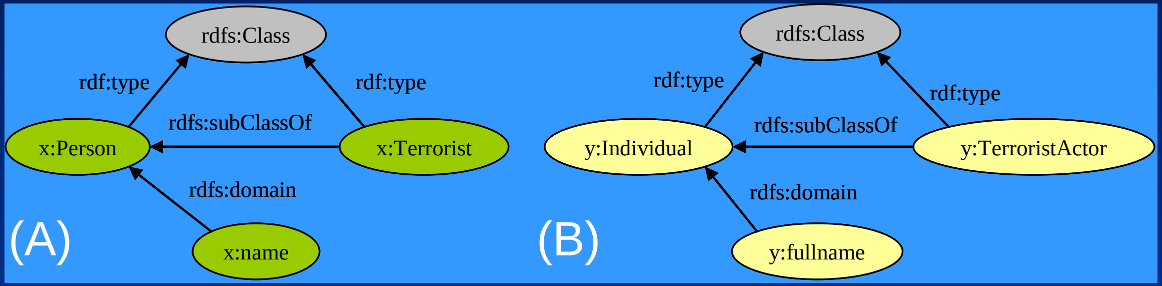


Two schemas for the same problem.

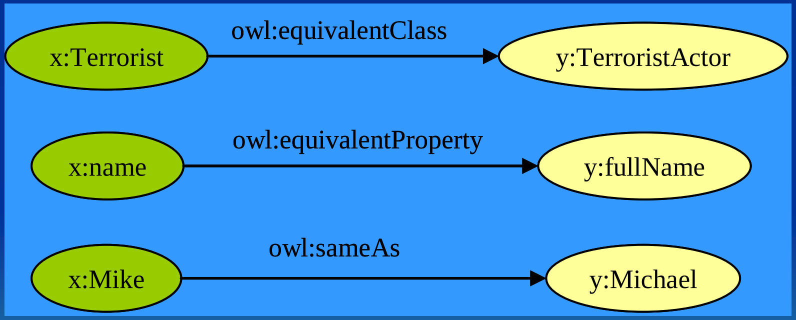


Sample instance data for each schema.

Mapping ontologies together

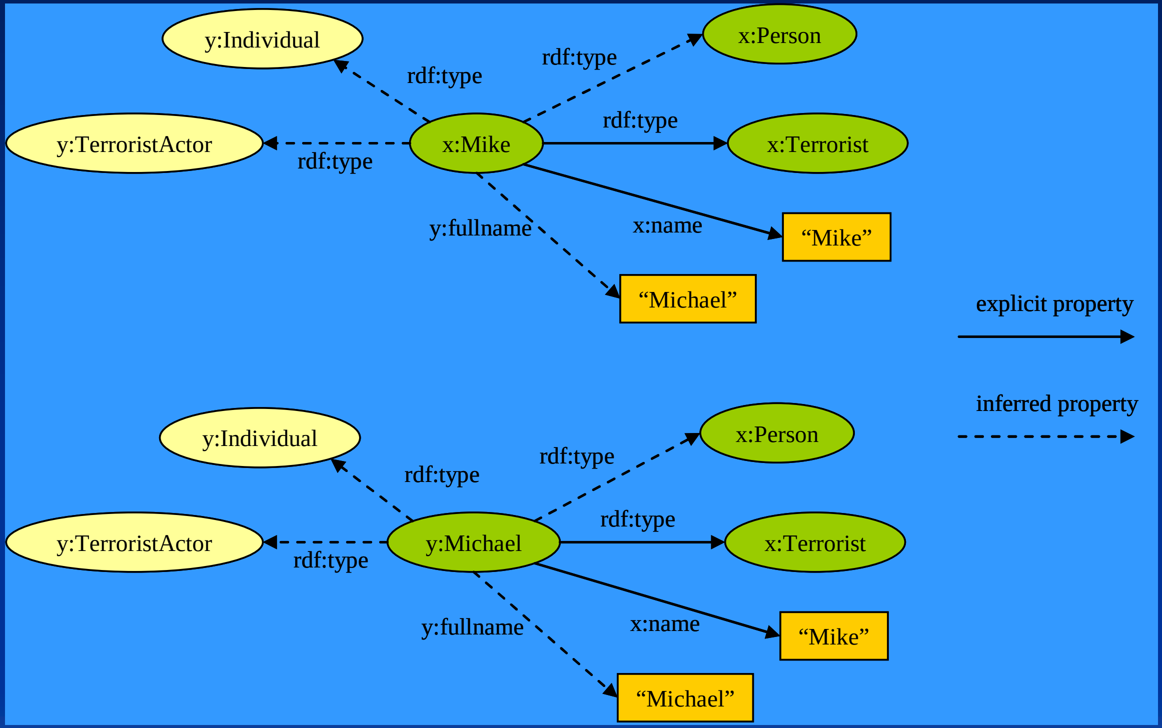


Two schemas for the same problem.



Assertions that map (A) and (B) together.

Semantically Aligned View



The data from both sources are "snapped together" once we assert that x:Mike and y:Michael are the same individual.

Bryan Thompson
Chief Scientist
SYSTAP, LLC
bryan@systap.com
202-462-9888

bigdata ™

Flexible
Reliable
Affordable
Web-scale computing.