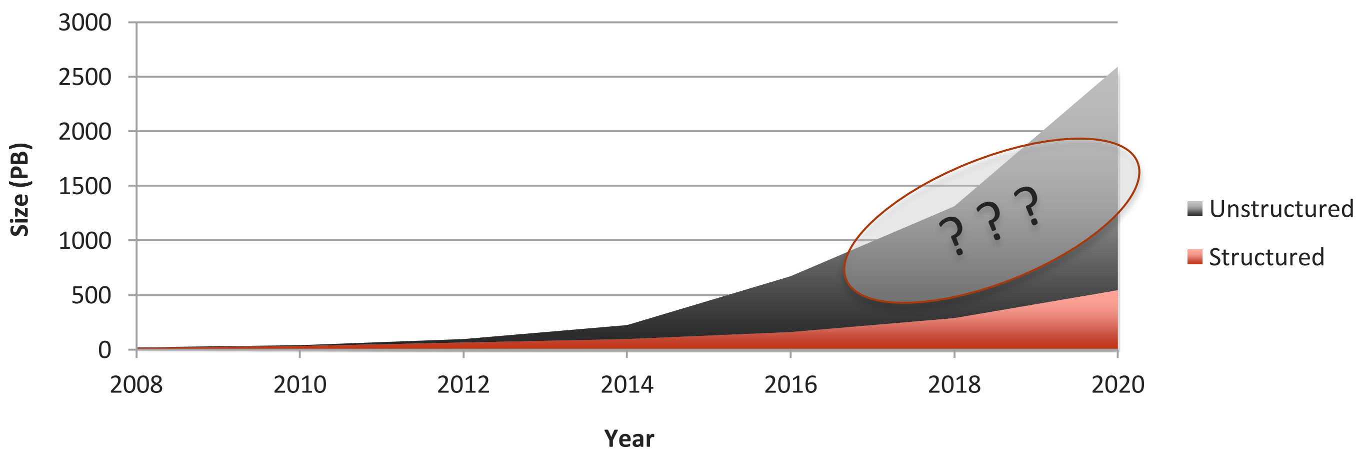
Building Scalable Technologies for Semantic Analysis

JOHN FEO
HIGH PERFORMANCE DATA ANALYTICS PROJECT
PACIFIC NORTHWEST NATIONAL LABORATORY

The problem

- Data is no longer "owner produced," but rather gathered from external sources on the web. *It is unstructured and heterogeneous.*
- The fixed schemas and table formats of relational databases are too rigid for web-gathered data.
- NoSQL databases have emerged, but their chosen approach of distributing data over many systems makes finding complex connections prohibitive.

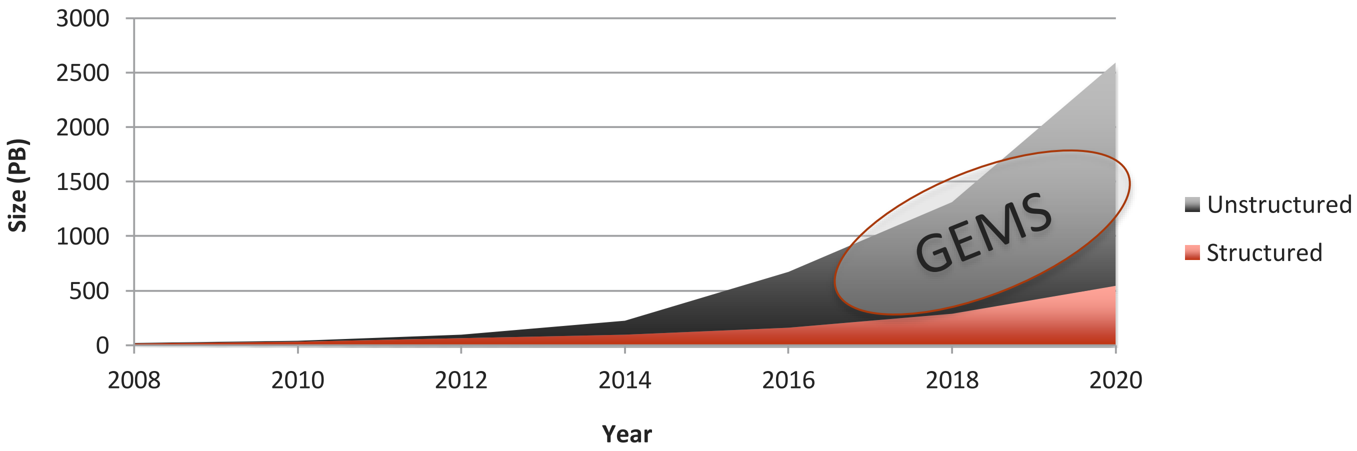


GEMS - A DATABASE ALIGNED WITH FUTURE DATA TRENDS THAT EXTENDS THE CAPABILITIES OF EXISTING SOLUTIONS

Pacific Northwest NATIONAL LABORATORY

Proudly Operated by Battelle Since 1965

- Flexible data model that supports structured and unstructured data in a single form
- In-memory datastore using local, remote, and flash memories
- General parallel programming model – *not record or vertex centric*
- Runs on commodity platforms from desktops to clouds – *no special system requirements*



Why do we perform better than others

Pacific Northwest NATIONAL LABORATORY
Proudly Operated by Battelle Since 1965

- We store unstructured data as a *graph*
- We process graph data using *graph methods*
- We support a general parallel programming model allowing *methods to be written naturally*
- We have developed a multithreaded runtime system that *scales out on commodity hardware*
- We use *standard languages* (*SPARQL, C++*)
- We require *no special systems* (*x86, Linux, MPI*)

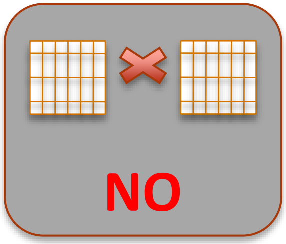
ADVANTAGES

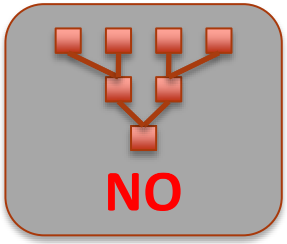
LARGER DATA SIZE

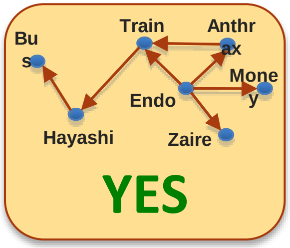
GREATER PRODUCTIVITY

FASTER TIME TO SOLUTION

LOWER COST OF OWNERSHIP







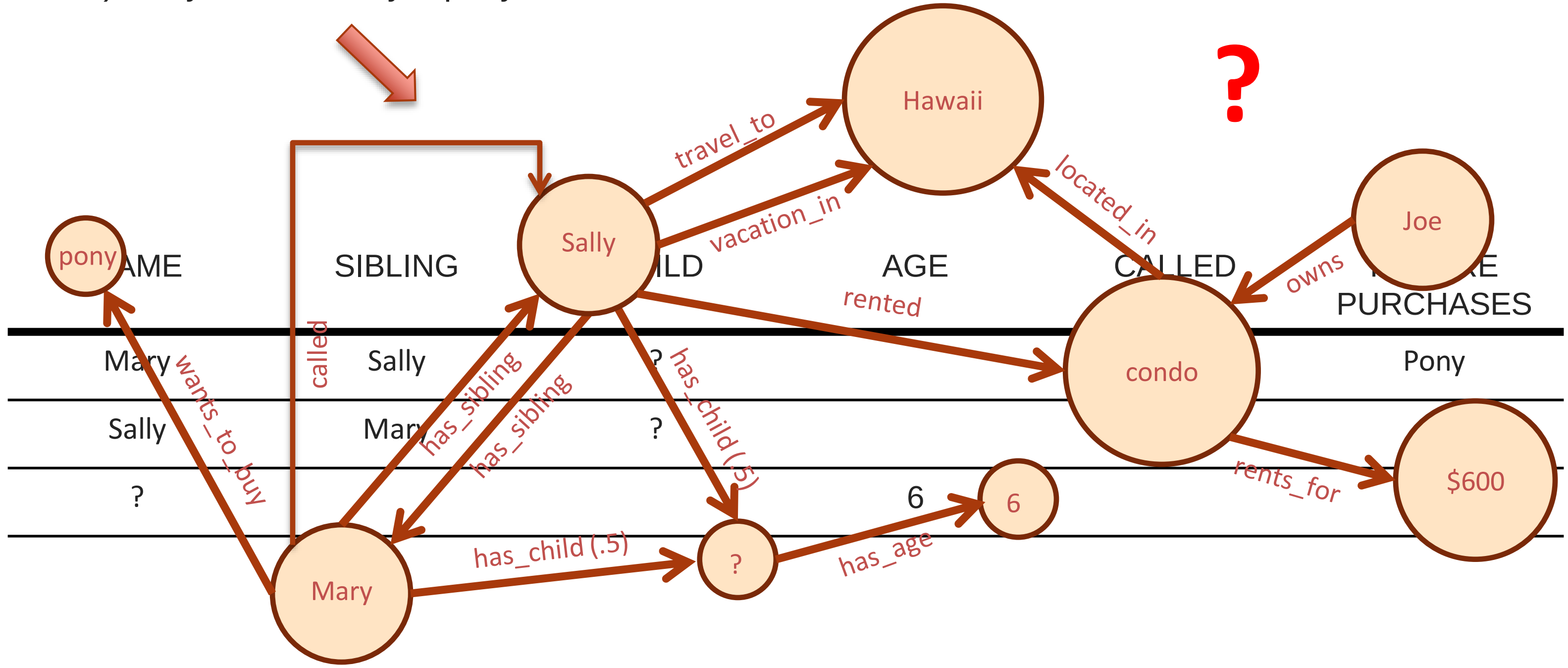
Use *graphs* rather than tables

Mary called her sister Sally to discuss buying her 6-year daughter a pony for Christmas.

1) Mary called Sally
2) Mary has a sister named Sally
3) Sally has a sister named Mary
4) Either Mary or Sally has a daughter
5) The daughter is 6 years old
6) Mary wants to buy a pony

Sally rented Joe's condo in Hawaii for a two week vacation. She paid $1200 rent.

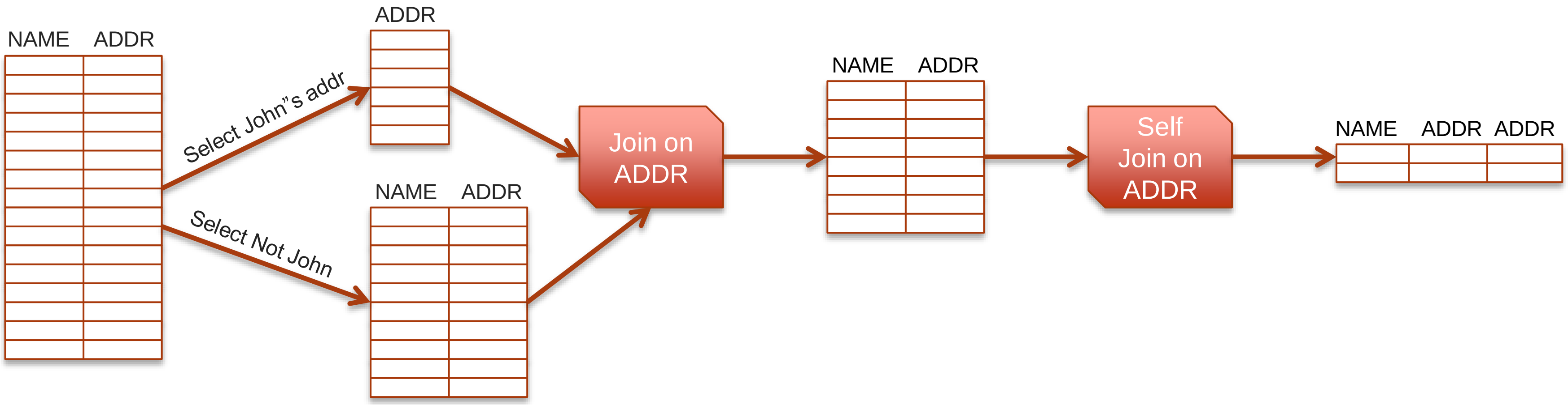
1) Sally traveled to Hawaii
2) Sally vacationed in Hawaii
3) Joe owns a condo
4) Joe's condo is in Hawaii
5) Sally rented Joe's condo
6) Joe rented his condo for $600 per week

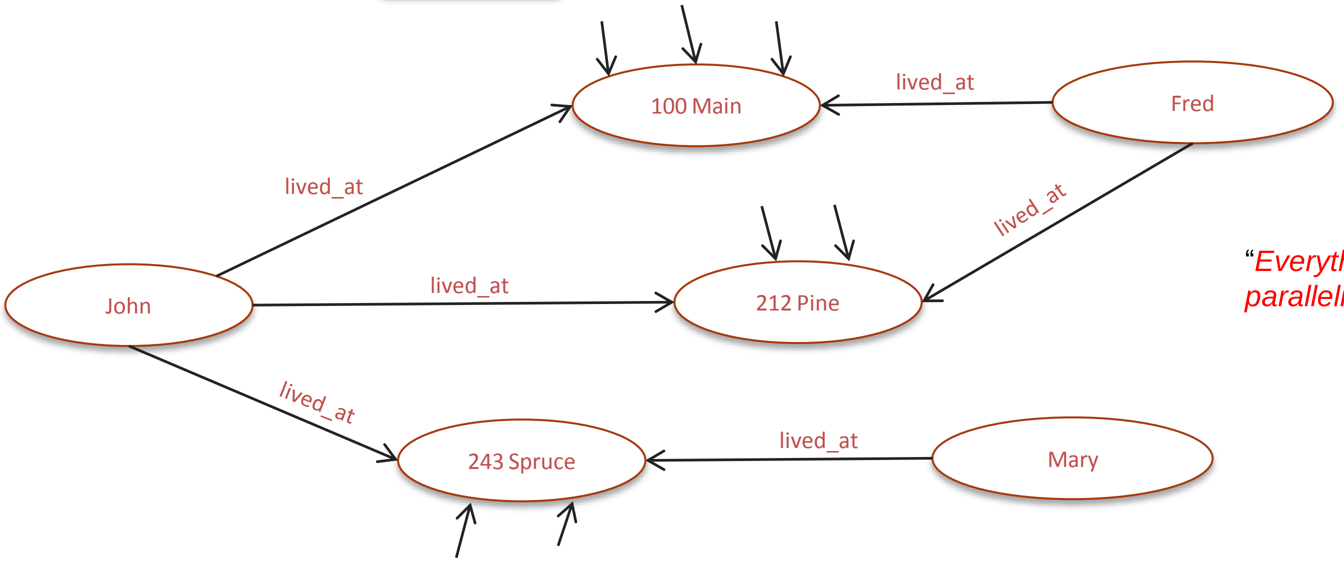


Use *graph algorithms* rather than table joins

Pacific Northwest NATIONAL LABORATORY
Proudly Operated by Battelle Since 1965

RETURN ALL PERSONS WHO HAVE SHARED 2 OR MORE ADDRESSES WITH JOHN





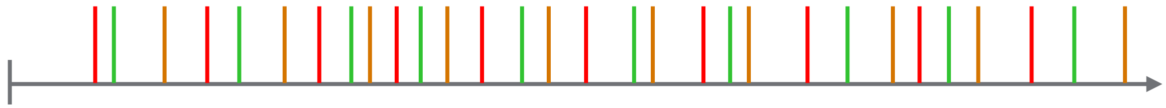
"*Everything you do at scale must be parallelized or it will run forever*"
- Michael Stonebraker

Use *memory* rather than disks

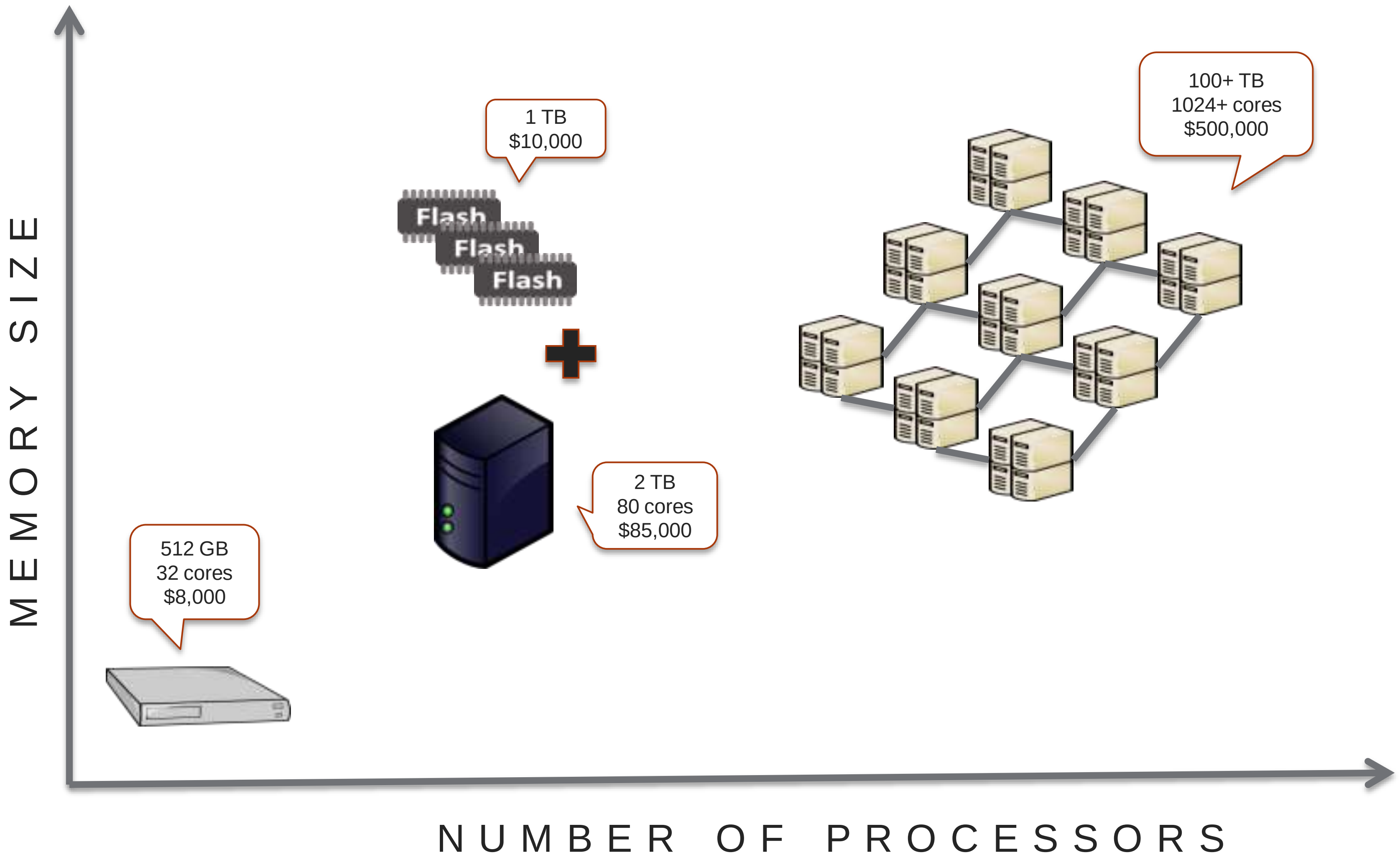
- Graph algorithms cannot take advantage of conventional storage hierarchies or locality-preserving, distributed data structures
  - *So keep everything in memory for fast random access*
- … but memory is very limited
  - *So use a cluster to expand available memory by adding nodes*
- … but distributed data incurs long latencies
  - *So use multithreading to tolerate latencies*

Use multithreading to hide latencies

- Generate hundreds of threads per core
- Rather than execute one thread at-a-time per core (conventional runtime), switch among active threads (multithreading runtime) such that …
- ***Gaps introduced by long latency operations in one thread are filled by instructions in other threads***



GEMS can scale up and scale out



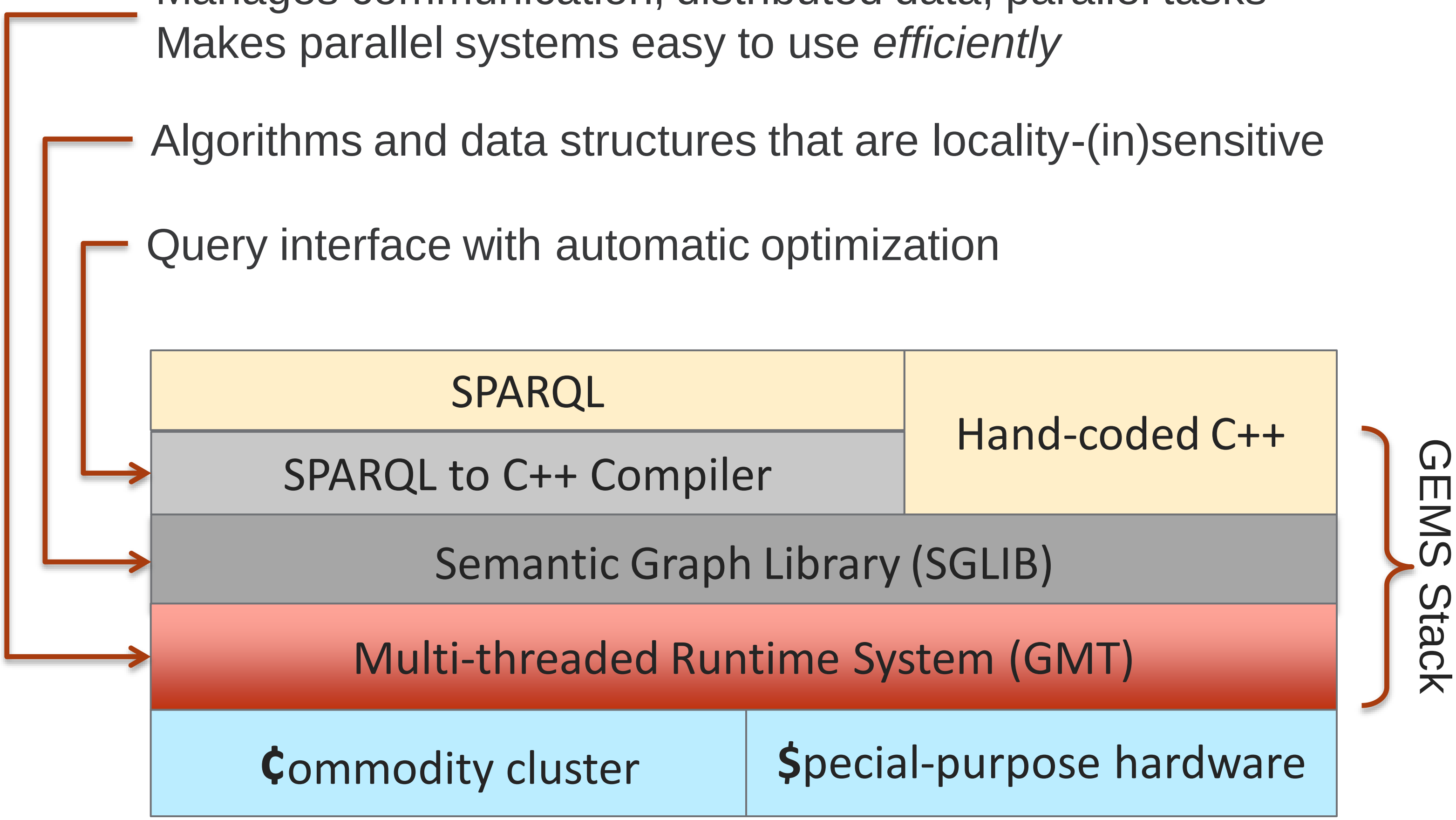
Clip by OCAL, Clker.com

GEMS software stack

Manages communication, distributed data, parallel tasks
Makes parallel systems easy to use *efficiently*

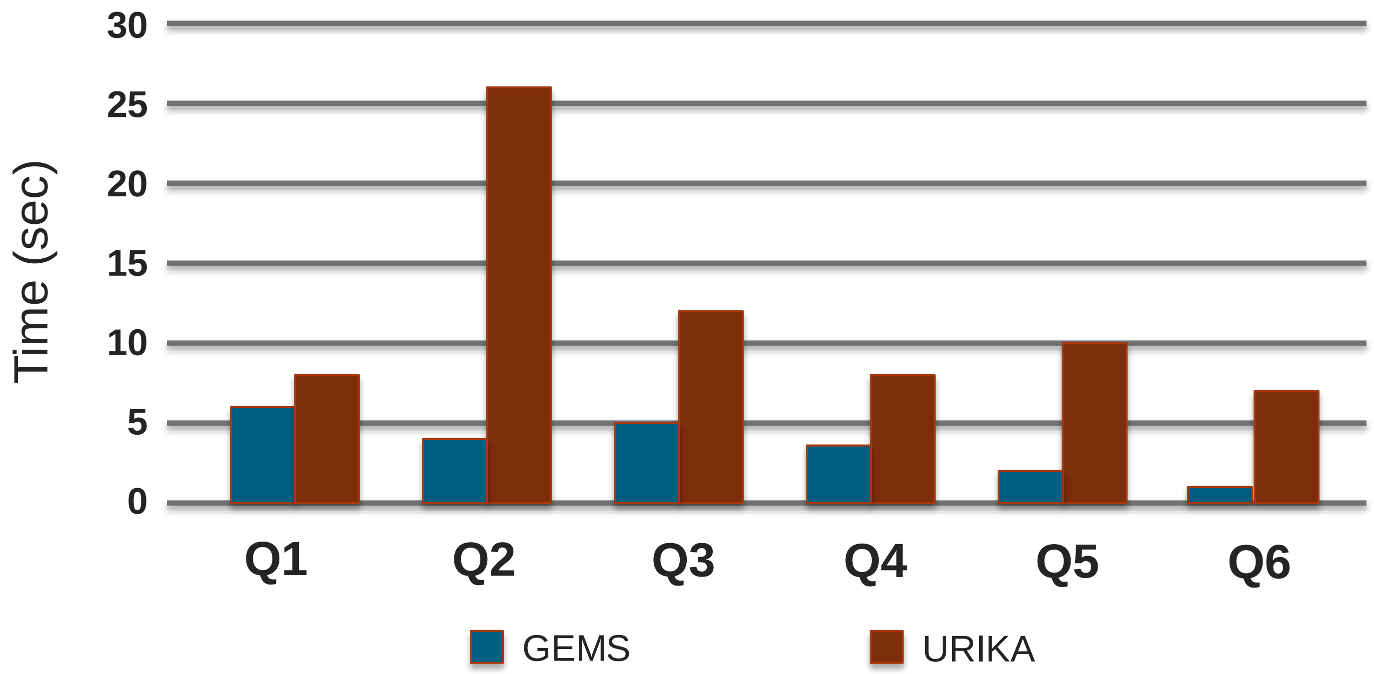
Algorithms and data structures that are locality-(in)sensitive

Query interface with automatic optimization



Berlin Benchmark – GEMS vs Urika

1B triples, 4TB memory systems

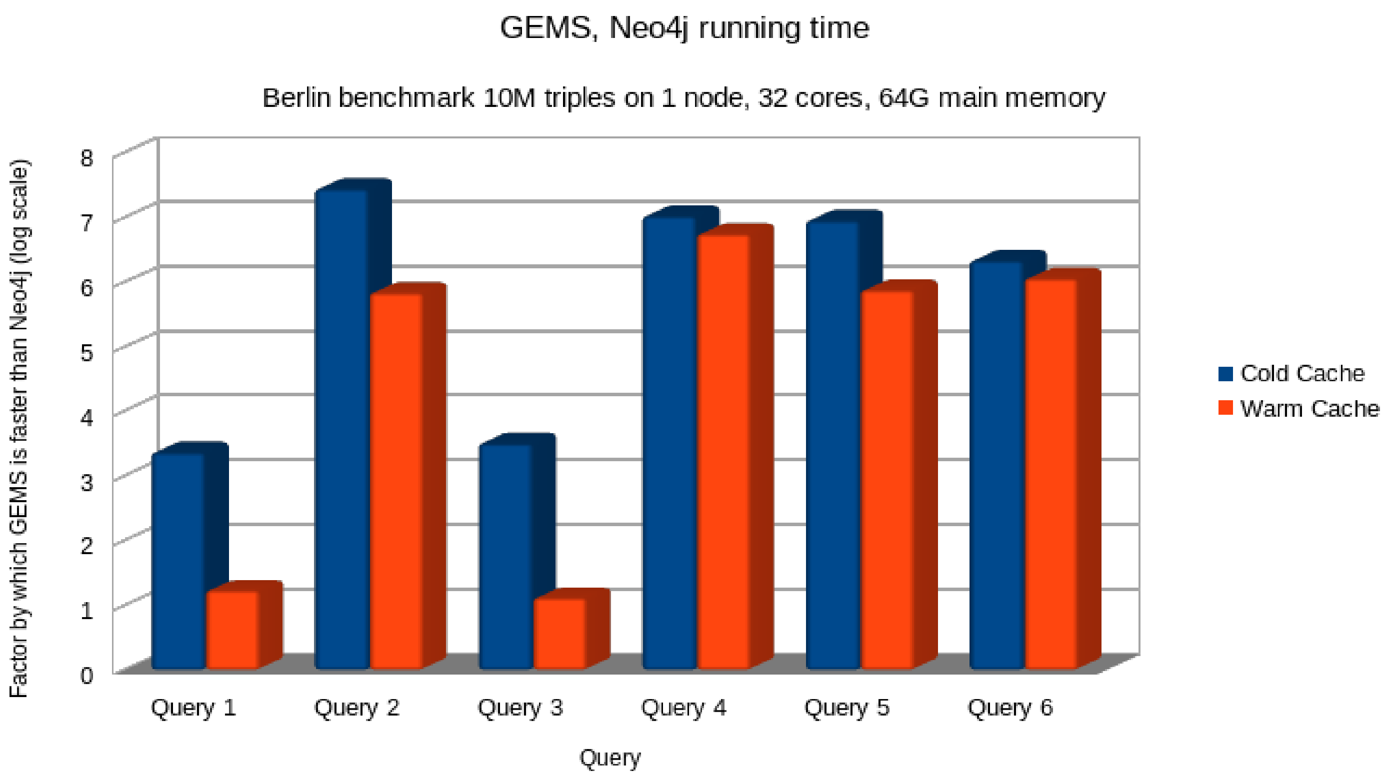


- Same main memory size, but GEMS system had half the processors

Berlin Benchmark – GEMS vs. Neo4j

Pacific Northwest NATIONAL LABORATORY

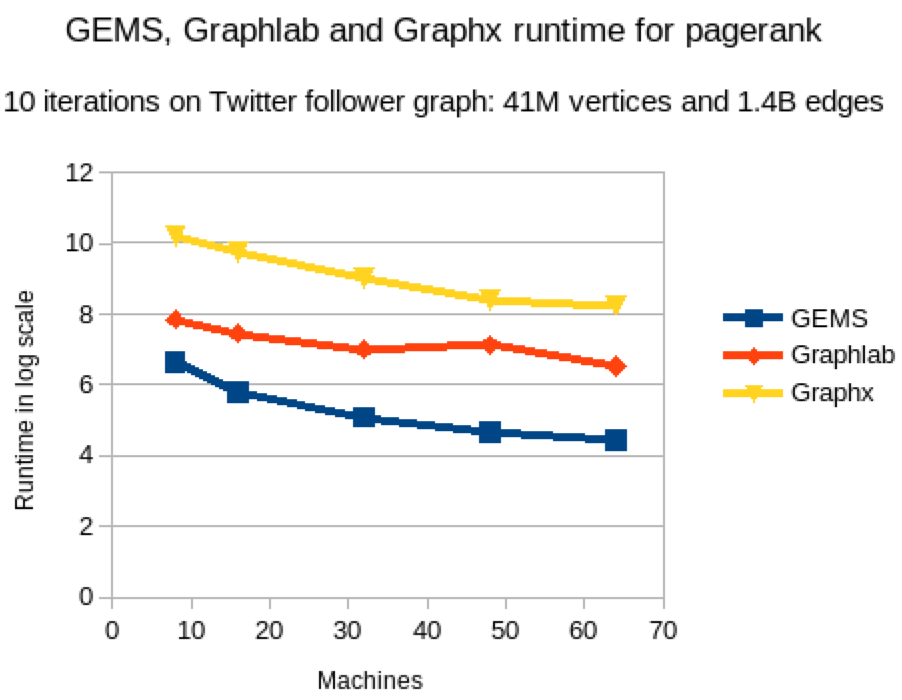
Proudly Operated by Battelle Since 1965



- Ran a data size that fit main memory to minimize Neo4j disk transfers
- Rebooted Neo4j to use best mode for each query
- Hired experienced Neo4j user to conduct test

GEMS vs. GraphLab and GraphX

Pacific Northwest NATIONAL LABORATORY
Proudly Operated by Battelle Since 1965



- Choose an algorithm studied heavily for both graph libraries
- Worked closely with library development groups to insure best performance
- GEMS is 4x faster than GraphLab and 16x faster than GraphX

Setup times – 1B triples

FROM TRIPLES FILE BUILD DICTIONARY, BUILD GRAPH, SAVE GZIP FILE	
16 P	1007 sec
32 P	555 sec
64 P	384 sec

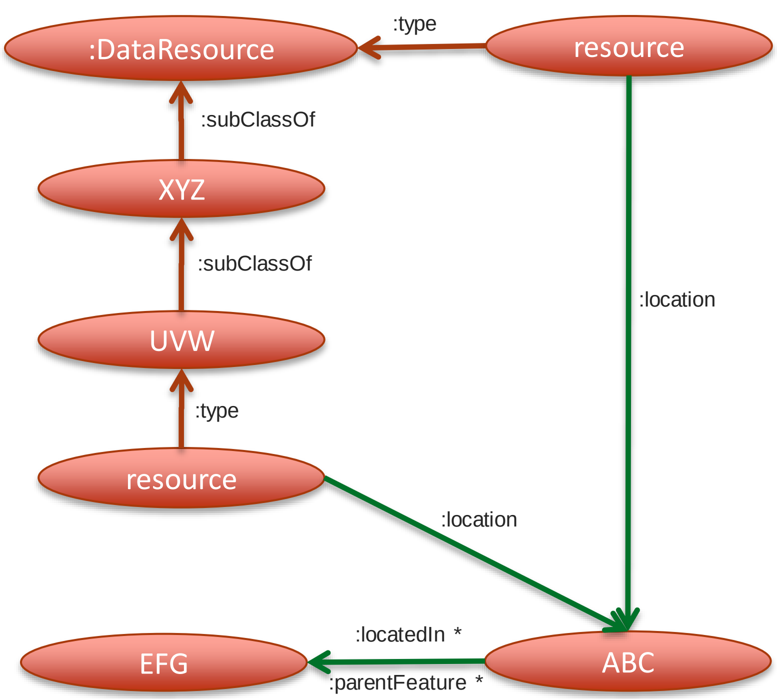
FROM GZIP FILE RESTORE TRIPLES, RESTORE DICTIONARY, BUILD GRAPH	
16 P	906 sec
32 P	432 sec
64 P	238 sec

Property paths


```
SELECT ?resource ?location WHERE {
  ?resource rdfs:a/rdfs:subClassOf* rdesc:DataResource .
  ?resource wgs84:location/(gn:locatedIn|gn:parentFeature)* ?location .
}
```

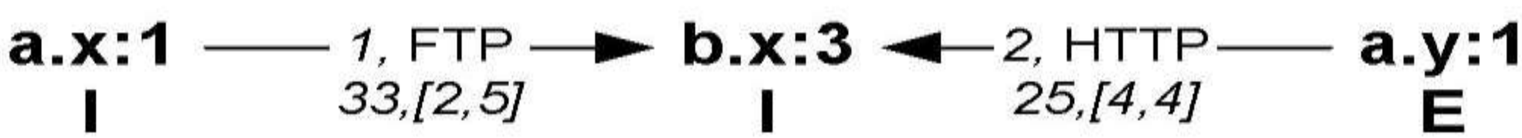
- A path (+, *) is just a recursive call

```
  int DR_Node = dictionary.lookup( “:DataResources” );
  forEach(ANY,  “:subClassOf” , DR_Node, Loop1);
  forEach(ANY,  “:type” , DR_Node, Loop2);
……
……

// ?dataResource :subClassOf :DataResource
static void Loop1(subject, predicate, object) {
  forEach(ANY,  “:subClassOf” , subject, Loop1);
  forEach(ANY,  “:type” , subject, Loop2);
}

// ?resource :type :DataResource
static void Loop2(subject, predicate, object) {
  args_t args;
  args.resource = subject;
  forEach(subject,  “:location” , ANY, Loop3, args);
}
```



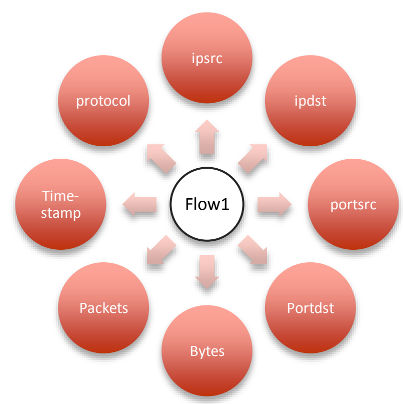
Attributed edges

- In many problem domains, relationships have many attributes



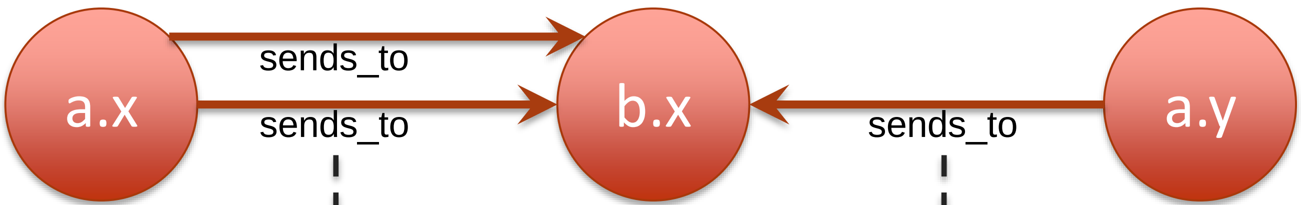
  - Node ID: complex structure, two octets, *A.B:P*
  - Node label: internal/external
  - Edge ID: unique number
  - Edge label: application protocol
  - Edge attributes: # packets, # bytes, time interval, …

- Creating “*star patterns*” wastes space and complicates query processing



Thick edges

- Recognize the distinction between relationships and attributes
  - Store relationships as a graph
  - Store attributes in a table



Left IP	Left Port	Left Label	Edge Id	Protocal	Size	Stime	Ftime	Right IP	Right Port	Right Label
a.x	1	I	1	FTP	33	2	5	b.x	3	I
a.y	1	E	2	HTTP	25	4	4	b.x	3	I

- Special predicates (UIDs) indicate record #
- Can enrich with traditional RDF edges

Conclusions

- We are developing a scalable, in-memory triplestore capable of knowledge discovery on web-scale data warehouse
  - Scales with data size
  - Multiple programming entry points
  - Conventional cluster and cloud systems
- We are working with government agencies and early adopters on real world problems
- We seek partners in transitioning our platform from prototype to production