
neotechnology
value in relationships

No SQL?

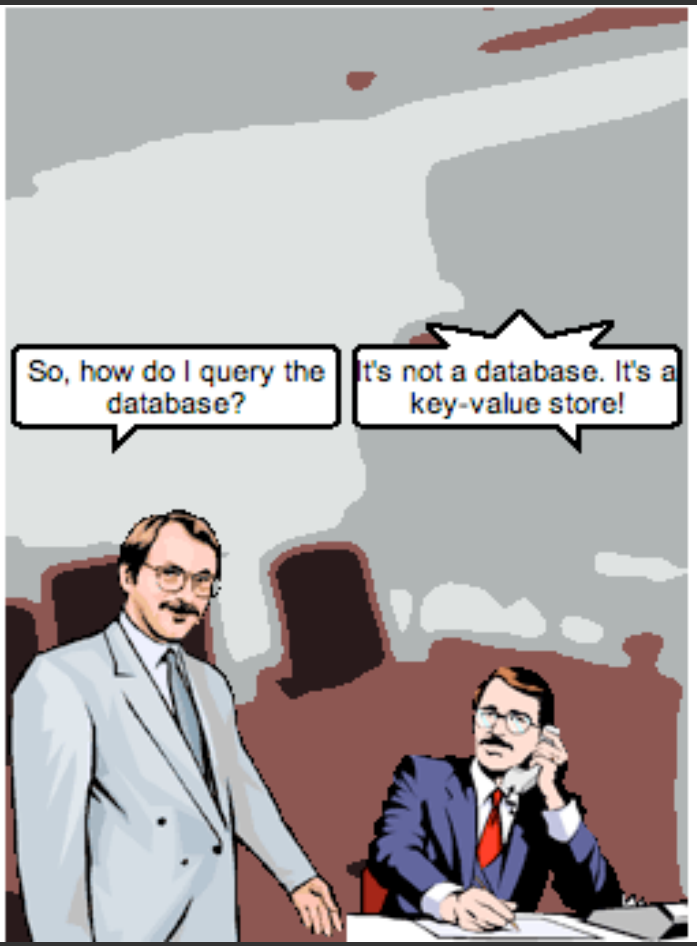


Image credit: http://browsertoolkit.com/fault-tolerance.png

No SQL?

Image credit: http://browsertoolkit.com/fault-tolerance.png

No SQL?

Image credit: http://browsertoolkit.com/fault-tolerance.png

Neo4j

or

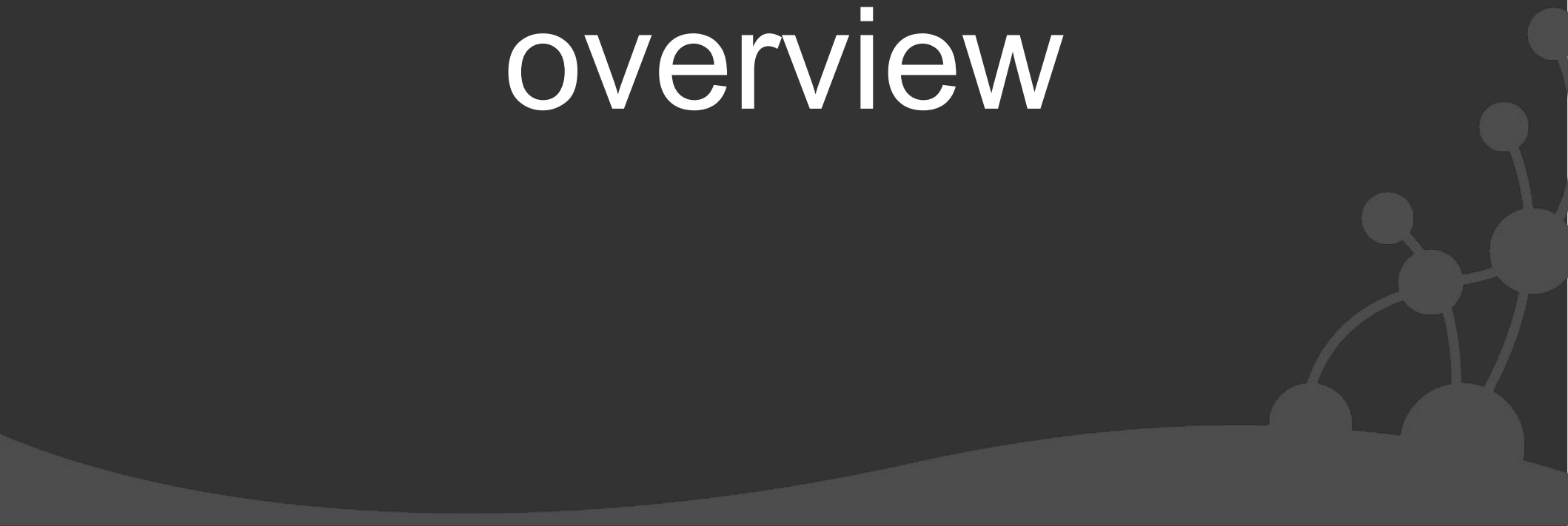
my nosql db is not built
in Erlang but I still <3 u!!!11

Emil Eifrem
CEO, Neo Technology

#neo4j
@emileifrem
emil@neotechnology.com

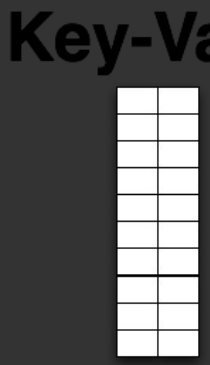
NOSQL

overview

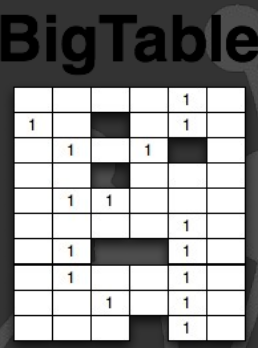


Four (emerging) NOSQL categories

- Key-value stores
  - Based on Amazon's Dynamo paper
  - Data model: (global) collection of K-V pairs
  - Example: Dynomite, Voldemort, Tokyo*

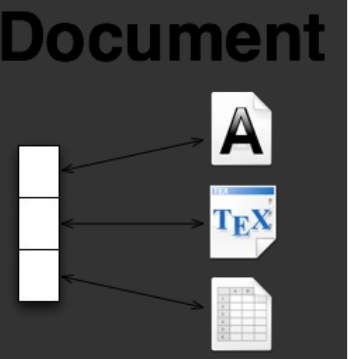


- ColumnFamily / BigTable clones
  - Based on Google's BigTable paper
  - Data model: big table, column families
  - Example: HBase, Hypertable, Cassandra

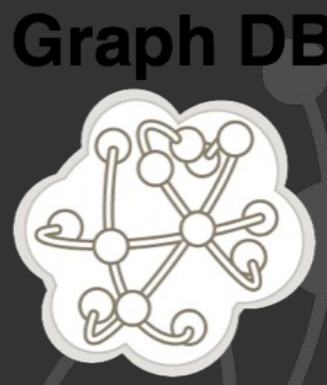


Four (emerging) NOSQL categories

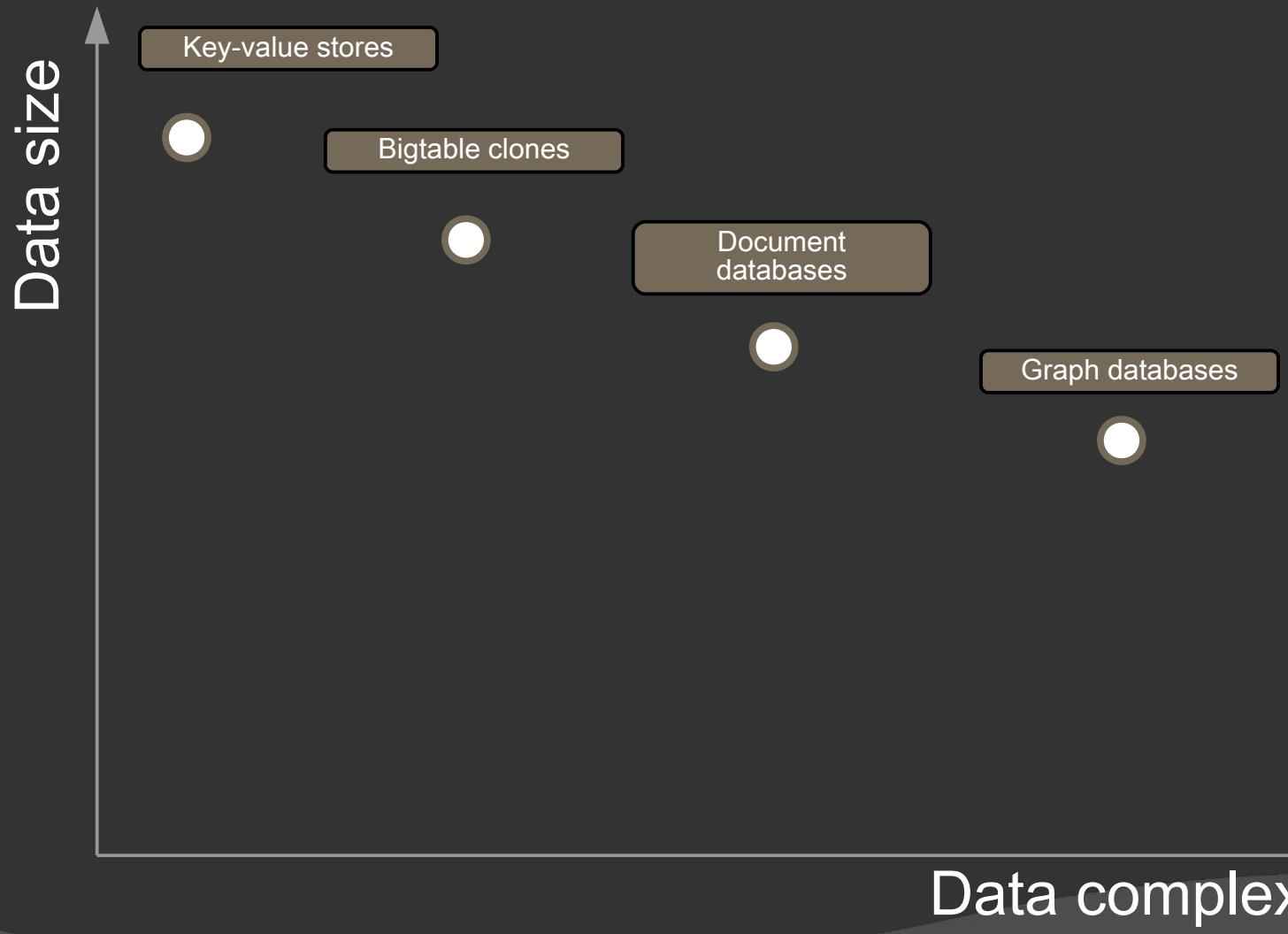
- Document databases
  - Inspired by Lotus Notes
  - Data model: collections of K-V collections
  - Example: CouchDB, MongoDB, Riak



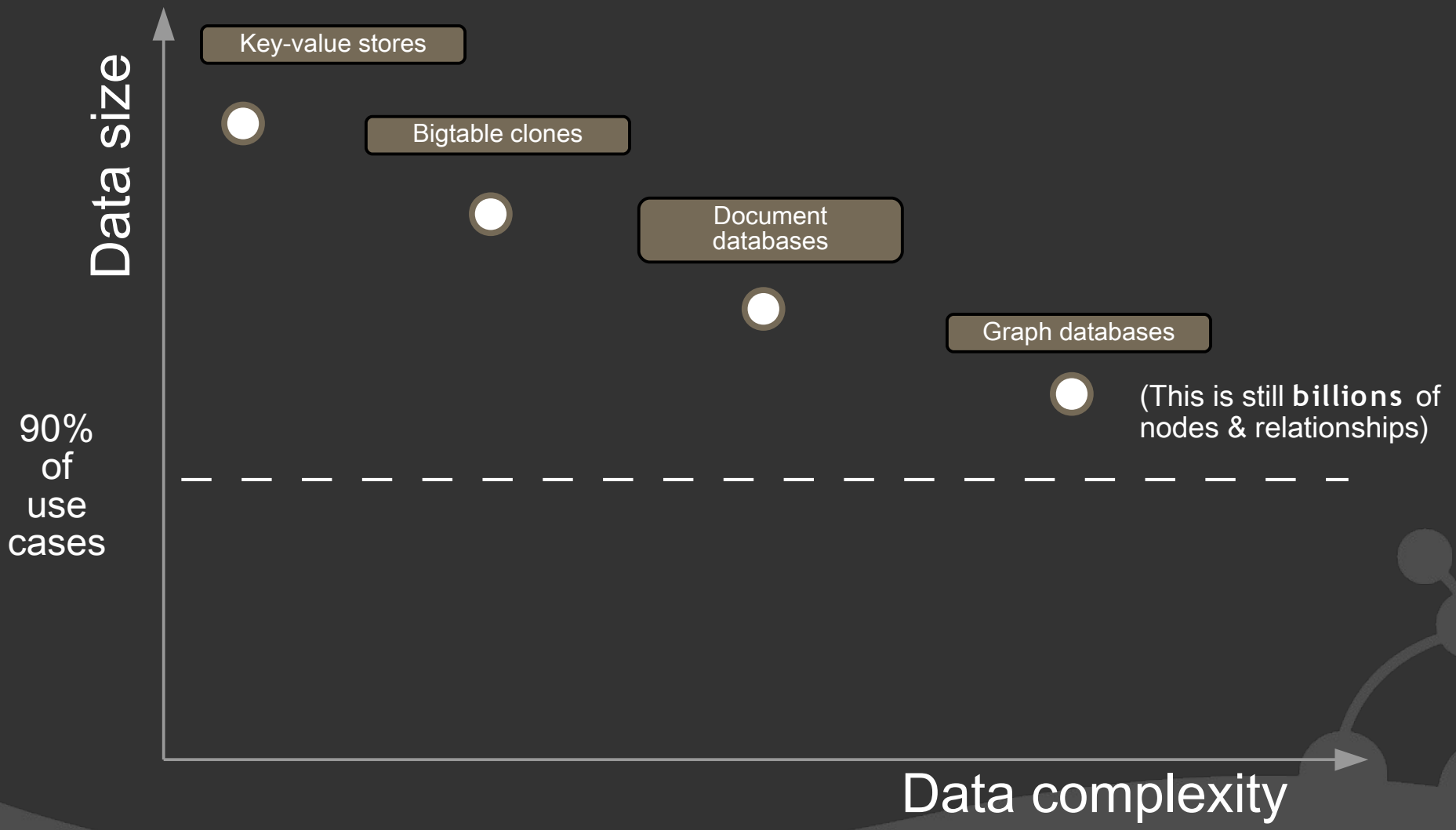
- Graph databases
  - Inspired by Euler & graph theory
  - Data model: nodes, rels, K-V on both
  - Example: AllegroGraph, Sones, Neo4j



NOSQL data models



NOSQL data models

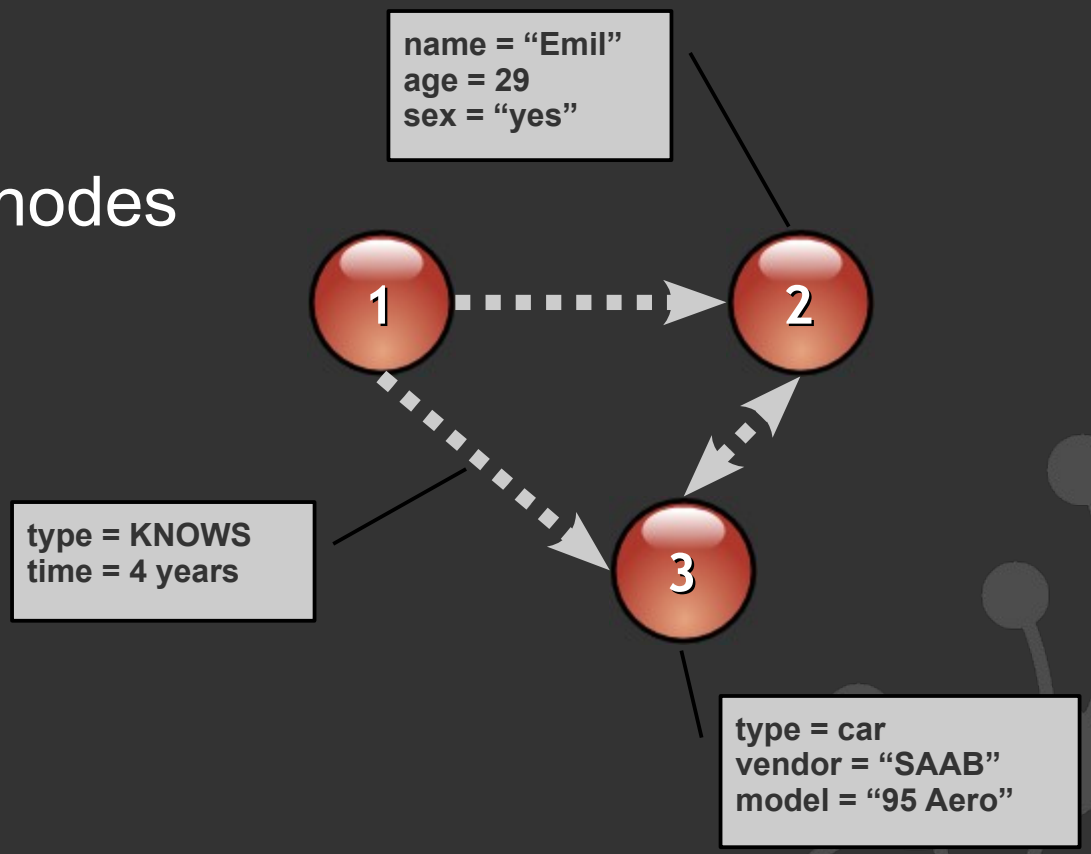


Graph DBs

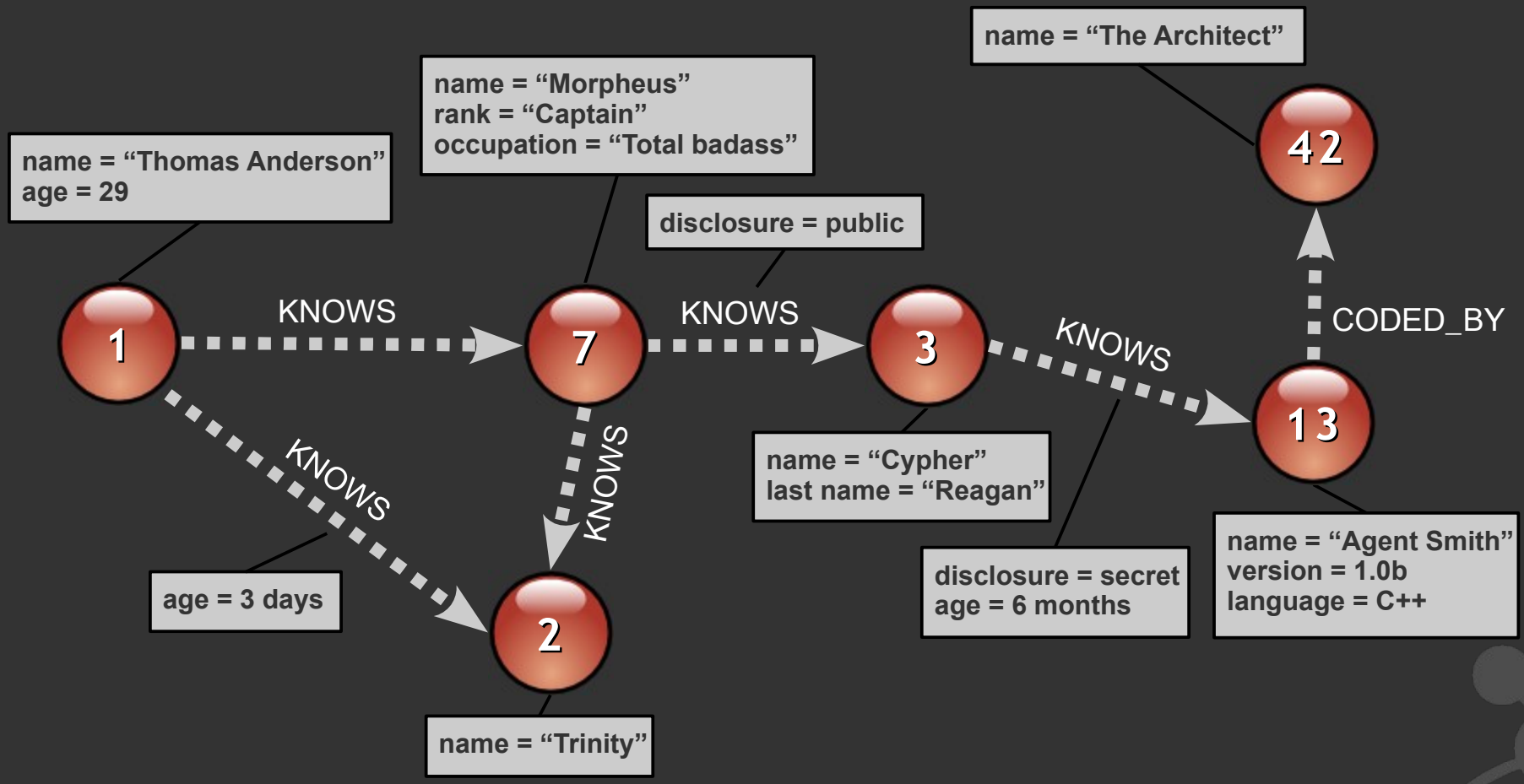
& Neo4j intro

The Graph DB model: representation

- Core abstractions:
  - Nodes
  - Relationships between nodes
  - Properties on both



Example: The Matrix



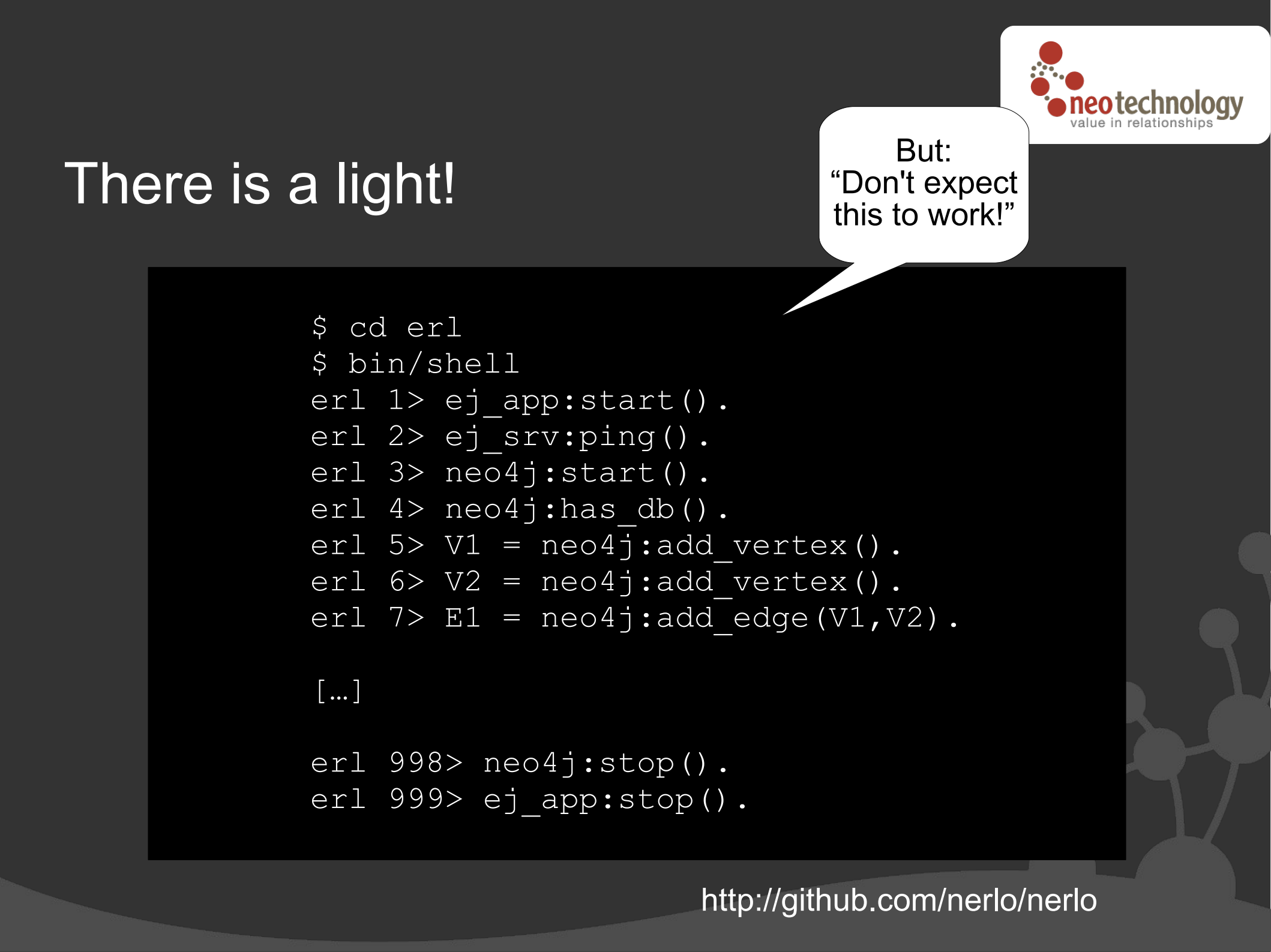
Code (1): Building a node space

```
GraphDatabaseService graphDb = ... // Get factory


// Create Thomas 'Neo' Anderson
Node mrAnderson = graphDb.createNode();
mrAnderson.setProperty( "name", "Thomas Anderson" );
mrAnderson.setProperty( "age", 29 );

// Create Morpheus
Node morpheus = graphDb.createNode();
morpheus.setProperty( "name", "Morpheus" );
morpheus.setProperty( "rank", "Captain" );
morpheus.setProperty( "occupation", "Total bad ass" );

// Create a relationship representing that they know each other
mrAnderson.createRelationshipTo( morpheus, RelTypes.KNOWS );
// ...create Trinity, Cypher, Agent Smith, Architect similarly
```

Code (1): Building a node space

```
GraphDatabaseService graphDb = ... // Get factory
Transaction tx = graphDb.beginTx();

// Create Thomas 'Neo' Anderson
Node mrAnderson = graphDb.createNode();
mrAnderson.setProperty( "name", "Thomas Anderson" );
mrAnderson.setProperty( "age", 29 );

// Create Morpheus
Node morpheus = graphDb.createNode();
morpheus.setProperty( "name", "Morpheus" );
morpheus.setProperty( "rank", "Captain" );
morpheus.setProperty( "occupation", "Total bad ass" );

// Create a relationship representing that they know each other
mrAnderson.createRelationshipTo( morpheus, RelTypes.KNOWS );
// ...create Trinity, Cypher, Agent Smith, Architect similarly

tx.commit();
```

There is a light!



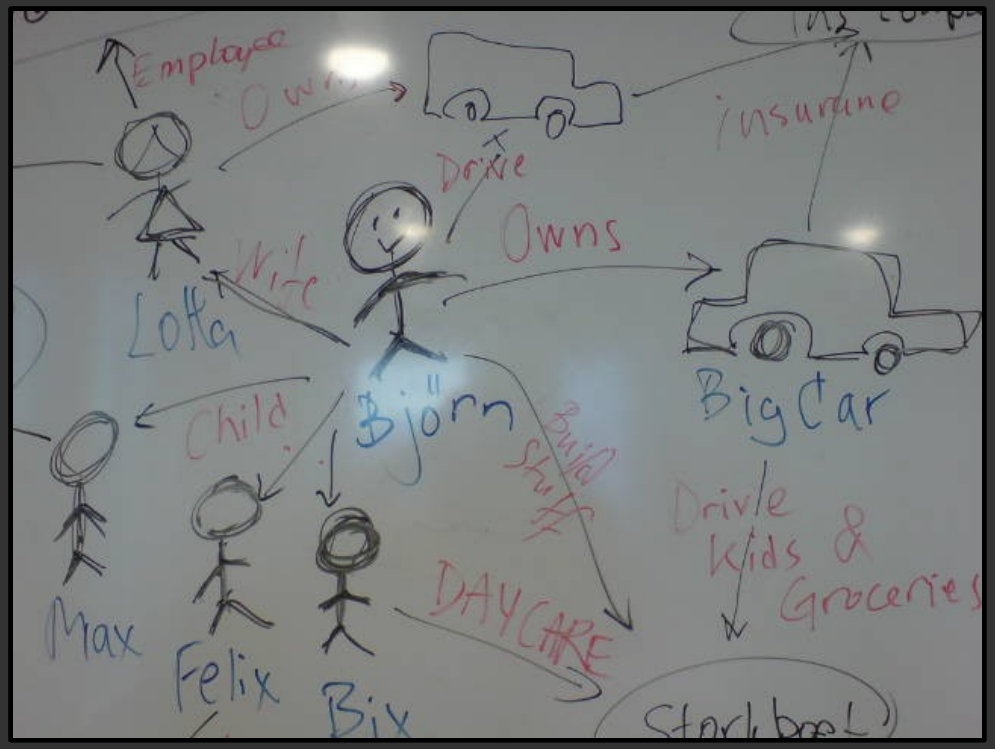

```
$ cd erl
$ bin/shell
erl 1> ej_app:start().
erl 2> ej_srv:ping().
erl 3> neo4j:start().
erl 4> neo4j:has_db().
erl 5> V1 = neo4j:add_vertex().
erl 6> V2 = neo4j:add_vertex().
erl 7> E1 = neo4j:add_edge(V1,V2).

[…]

erl 998> neo4j:stop().
erl 999> ej_app:stop().
```

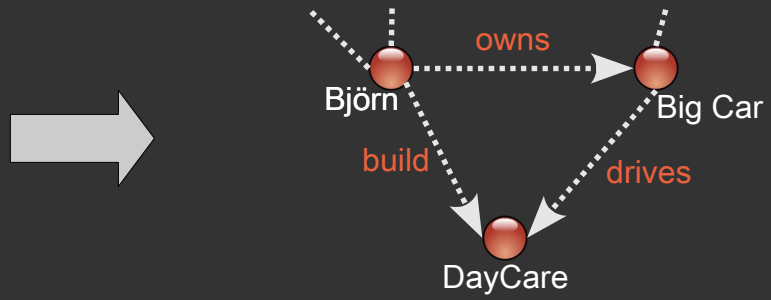
http://github.com/nerlo/nerlo

Code (1b): Defining RelationshipTypes

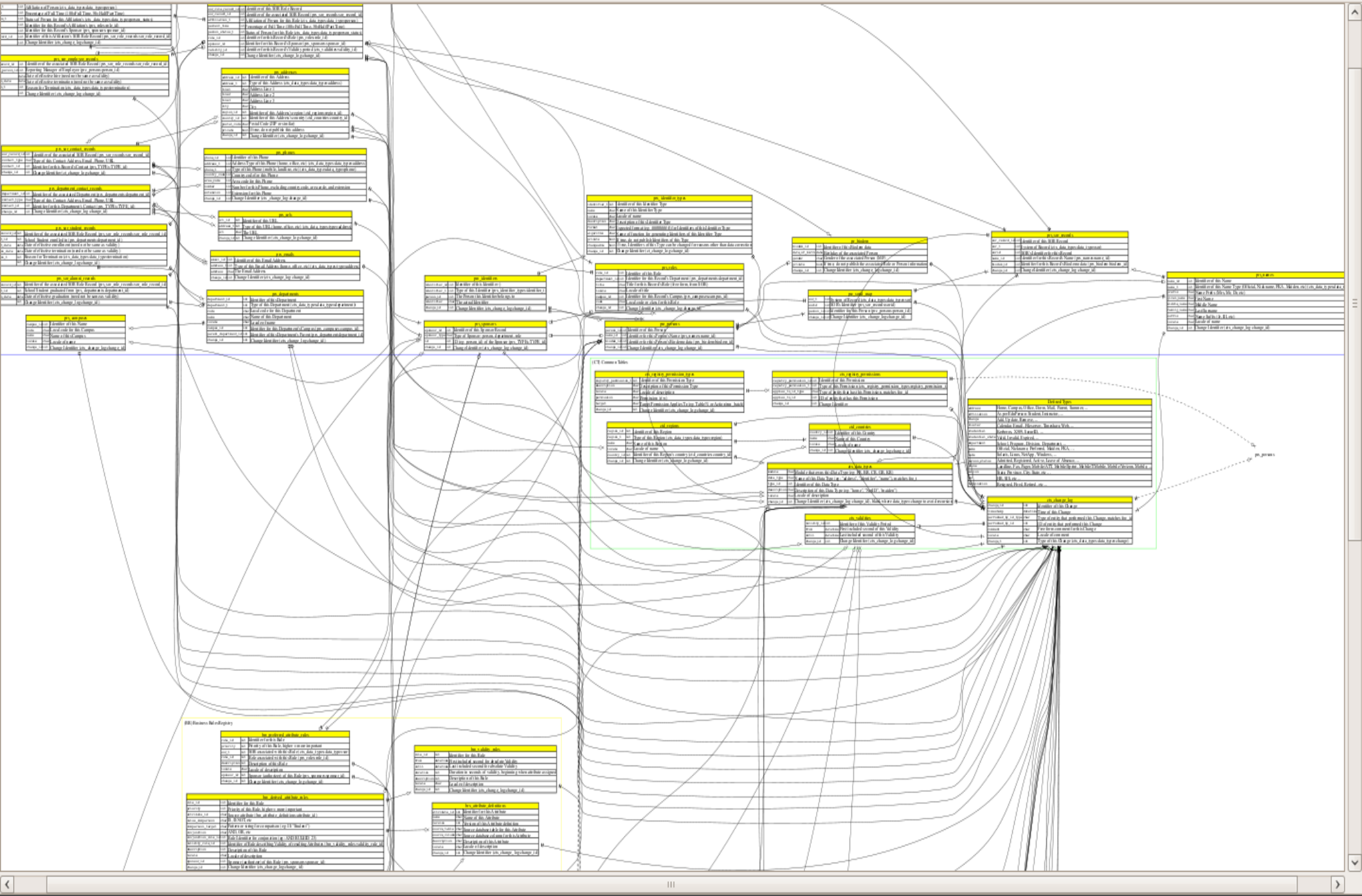
```
// In package org.neo4j.graphdb
public interface RelationshipType
{
    String name();
}

// In package org.yourdomain.yourapp
// Example on how to roll dynamic RelationshipTypes
class MyDynamicRelType implements RelationshipType
{
    private final String name;
    MyDynamicRelType( String name ){ this.name = name; }
    public String name() { return this.name; }
}

// Example on how to kick it, static-RelationshipType-like
enum MyStaticRelTypes implements RelationshipType
{
    KNOWS,
    WORKS_FOR,
}
```

Whiteboard friendly



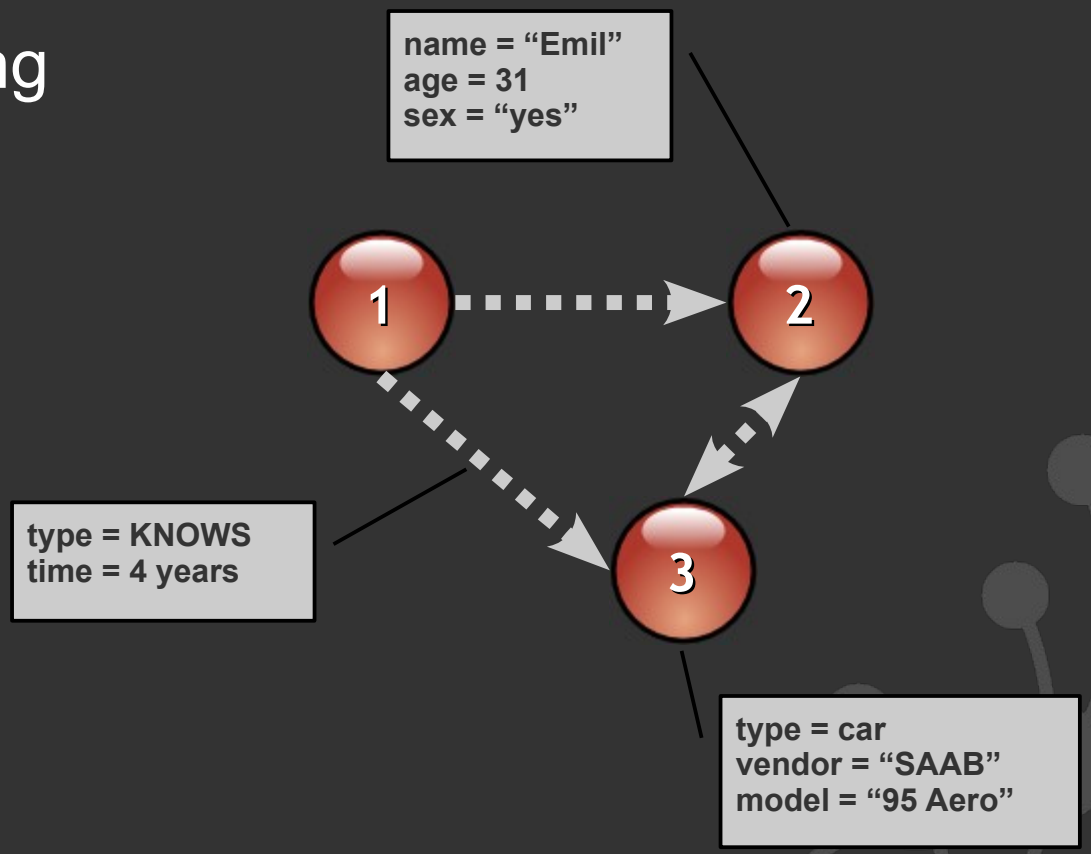


ORDATAMODEL
File Edit View Go Help
Previous Next
1 of 1
Continuous Best Fit Fit Width Zoom In Zoom Out
50%

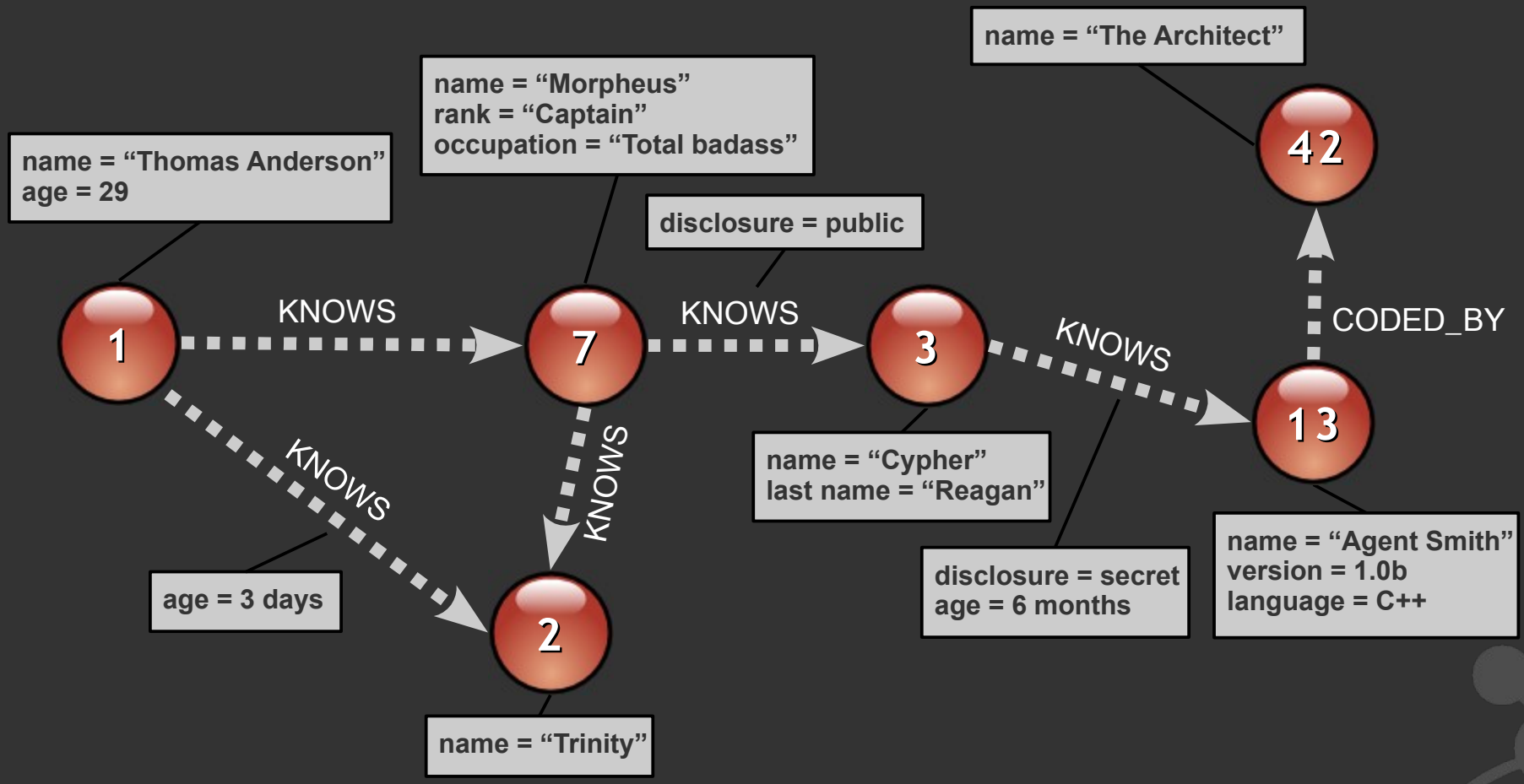


The Graph DB model: traversal

- Traverser framework for high-performance traversing across the node space

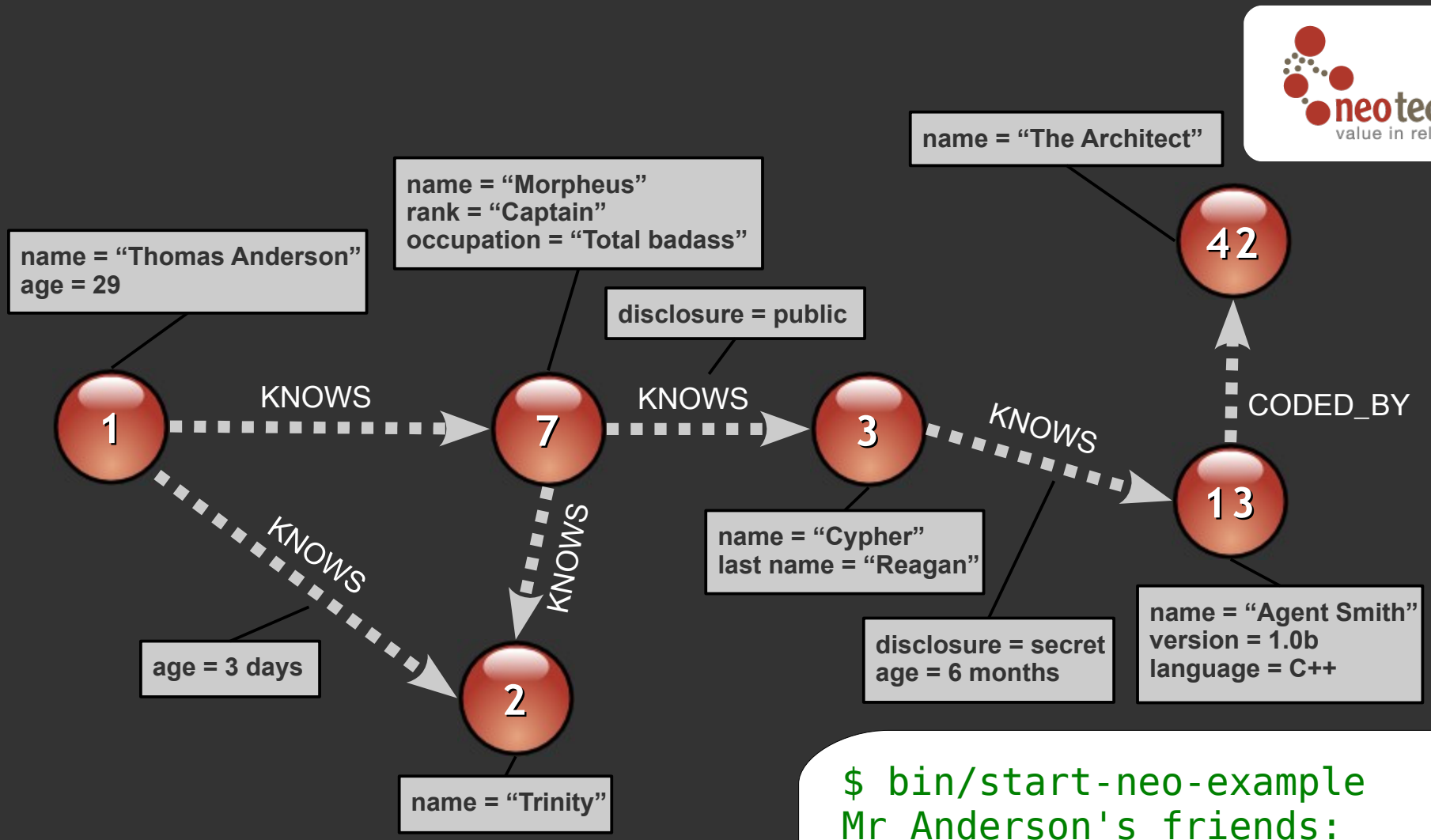


Example: Mr Anderson's friends



Code (2): Traversing a node space

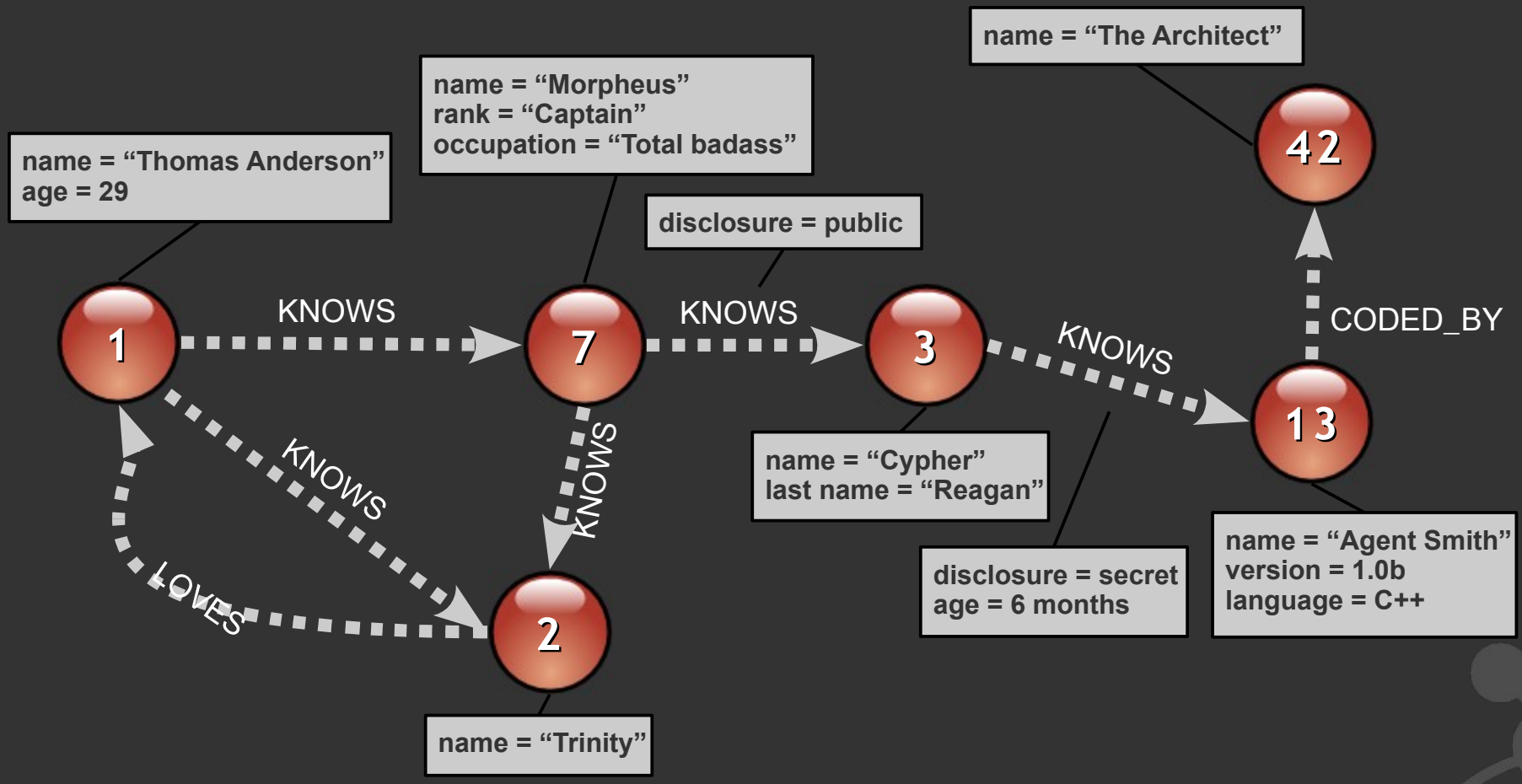
```
// Instantiate a traverser that returns Mr Anderson's friends
Traverser friendsTraverser = mrAnderson.traverse(
    Traverser.Order.BREADTH_FIRST,
    StopEvaluator.END_OF_GRAPH,
    ReturnableEvaluator.ALL_BUT_START_NODE,
    RelTypes.KNOWS,
    Direction.OUTGOING );

// Traverse the node space and print out the result
System.out.println( "Mr Anderson's friends:" );
for ( Node friend : friendsTraverser )
{
    System.out.printf( "At depth %d => %s%n",
        friendsTraverser.currentPosition().getDepth(),
        friend.getProperty( "name" ) );
}
```




```
friendsTraverser = mrAnderson.traverse(
  Traverser.Order.BREADTH_FIRST,
  StopEvaluator.END_OF_GRAPH,
  ReturnableEvaluator.ALL_BUT_START_NODE,
  RelTypes.KNOWS,
  Direction.OUTGOING );
```

```
$ bin/start-neo-example
Mr Anderson's friends:

At depth 1 => Morpheus
At depth 1 => Trinity
At depth 2 => Cypher
At depth 3 => Agent Smith
$
```

Example: Friends in love?

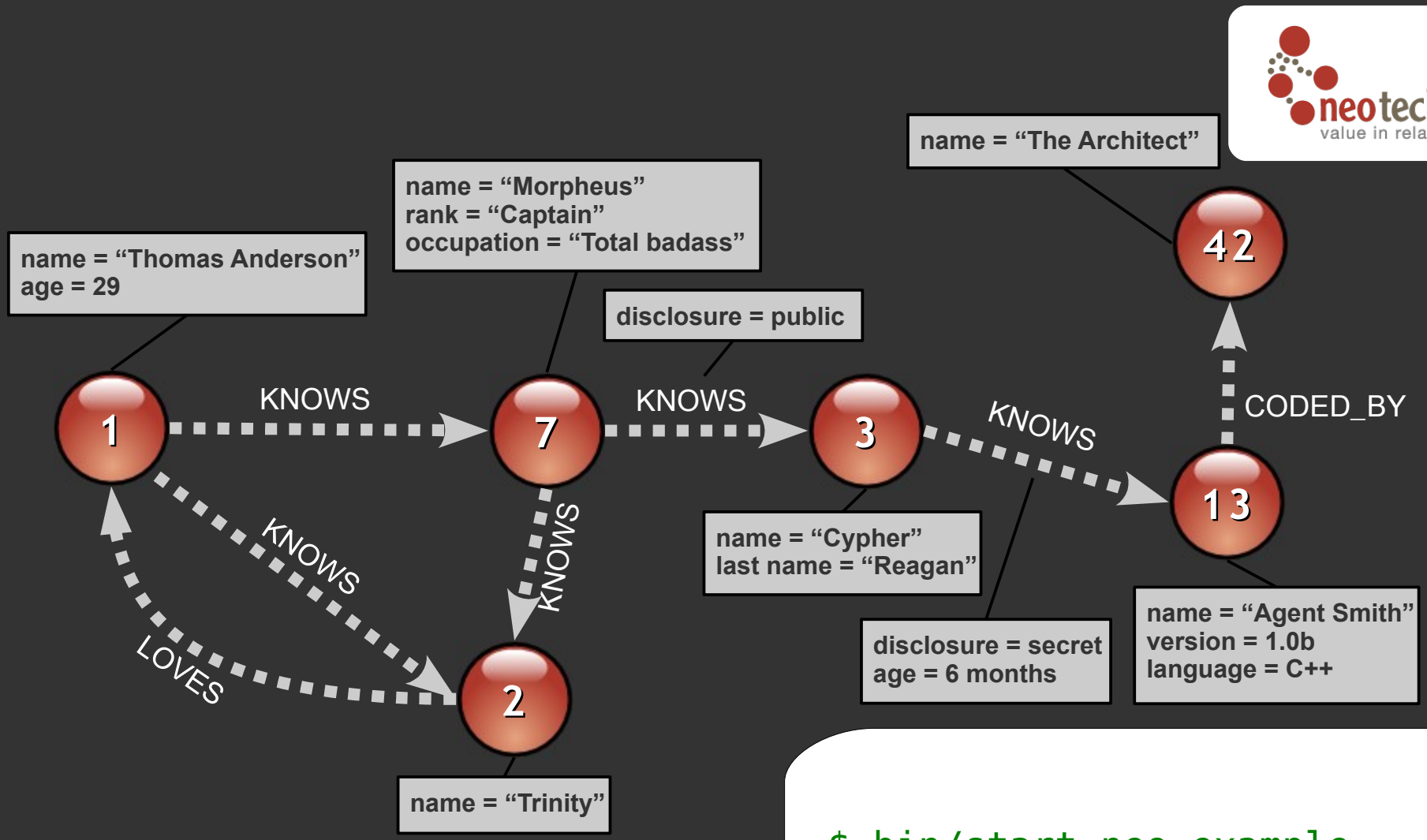


Code (3a): Custom traverser

```
// Create a traverser that returns all "friends in love"
Traverser loveTraverser = mrAnderson.traverse(
    Traverser.Order.BREADTH_FIRST,
    StopEvaluator.END_OF_GRAPH,
    new ReturnableEvaluator()
    {
        public boolean isReturnableNode( TraversalPosition pos )
        {
            return pos.currentNode().hasRelationship(
                RelTypes.LOVES, Direction.OUTGOING );
        }
    },
    RelTypes.KNOWS,
    Direction.OUTGOING );
```

Code (3a): Custom traverser

```
// Traverse the node space and print out the result
System.out.println( "Who's a lover?" );
for ( Node person : loveTraverser )
{
    System.out.printf( "At depth %d => %s%n",
        loveTraverser.currentPosition().getDepth(),
        person.getProperty( "name" ) );
}
```




```
new ReturnableEvaluator()
{
  public boolean isReturnableNode(
    TraversalPosition pos)
  {
    return pos.currentNode().
      hasRelationship( RelTypes.LOVES,
        Direction.OUTGOING );
  }
},
```

```
$ bin/start-neo-example
Who’s a lover?

At depth 1 => Trinity
$
```

Bonus code: domain model

- How do you implement your domain model?
- Use the delegator pattern, i.e. every domain entity wraps a Neo4j primitive:

```
// In package org.yourdomain.yourapp
class PersonImpl implements Person
{
    private final Node underlyingNode;
    PersonImpl( Node node ){ this.underlyingNode = node; }

    public String getName()
    {
        return (String) this.underlyingNode.getProperty( "name" );
    }
    public void setName( String name )
    {
        this.underlyingNode.setProperty( "name", name );
    }
}
```

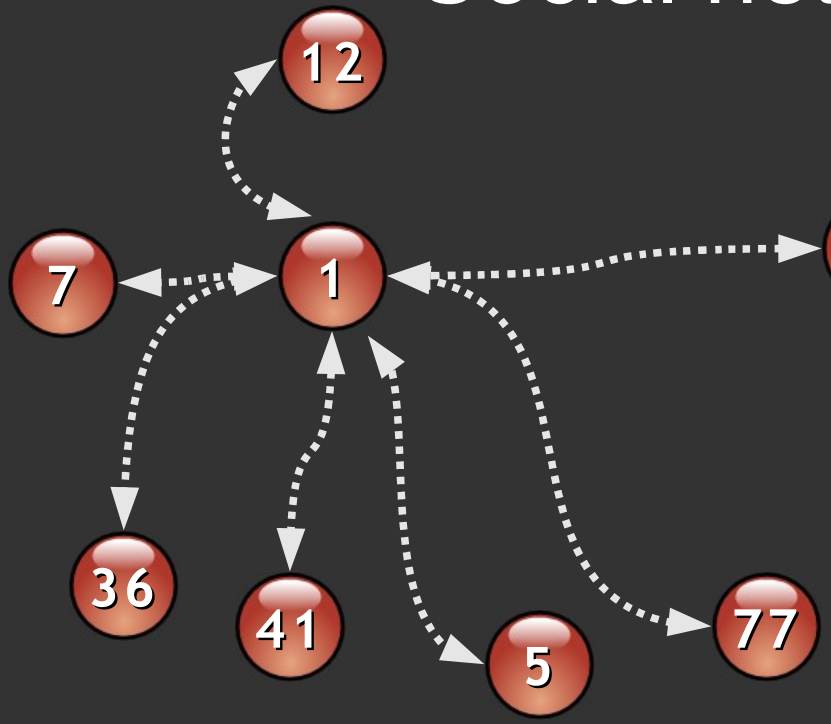
Domain layer frameworks

- Qi4j (www.qi4j.org)
  - Framework for doing DDD in pure Java5
  - Def nes Entities / Associations / Properties
    - Sound familiar? Nodes / Rel's / Properties!
  - Neo4j is an "EntityStore" backend
- Jo4neo (http://code.google.com/p/jo4neo)
  - Annotation driven
  - Weaves Neo4j-backed persistence into domain objects at runtime

Neo4j system characteristics

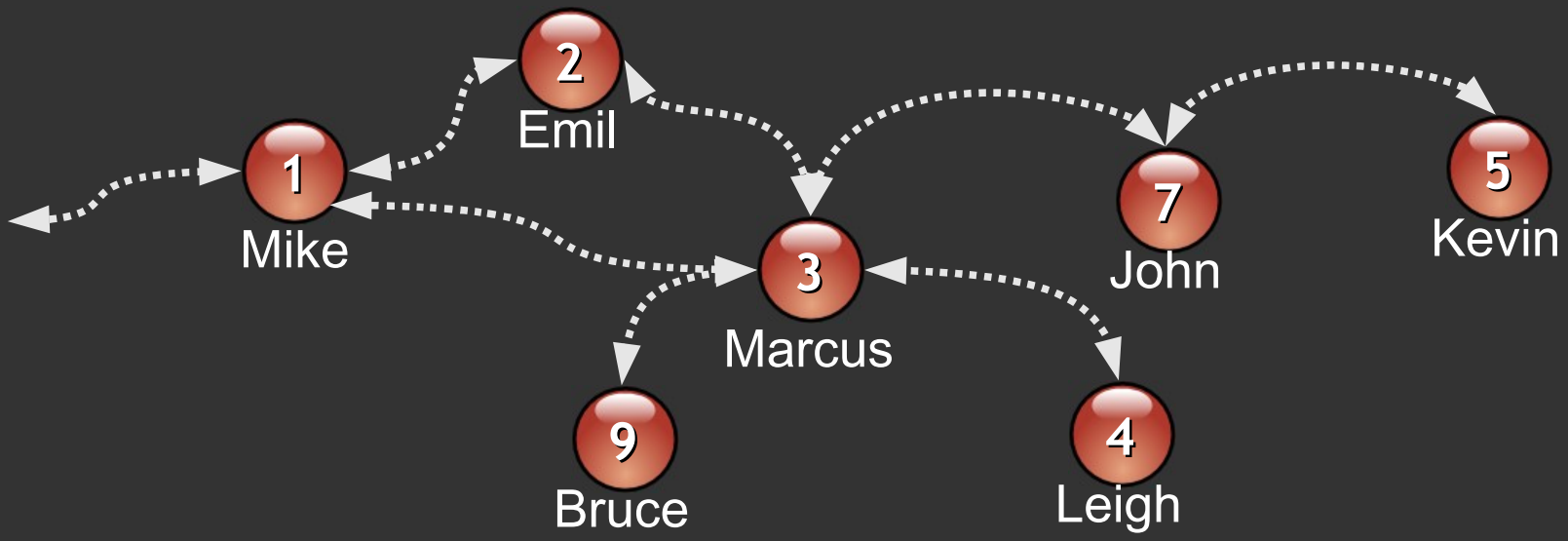
- Disk-based
  - Native graph storage engine with custom binary on-disk format
- Transactional
  - JTA/JTS, XA, 2PC, Tx recovery, deadlock detection, MVCC, etc
- Scales up
  - Many **billions** of nodes/rels/props on single JVM
- Robust
  - 7+ years in 24/7 production

Social network *pathExists()*

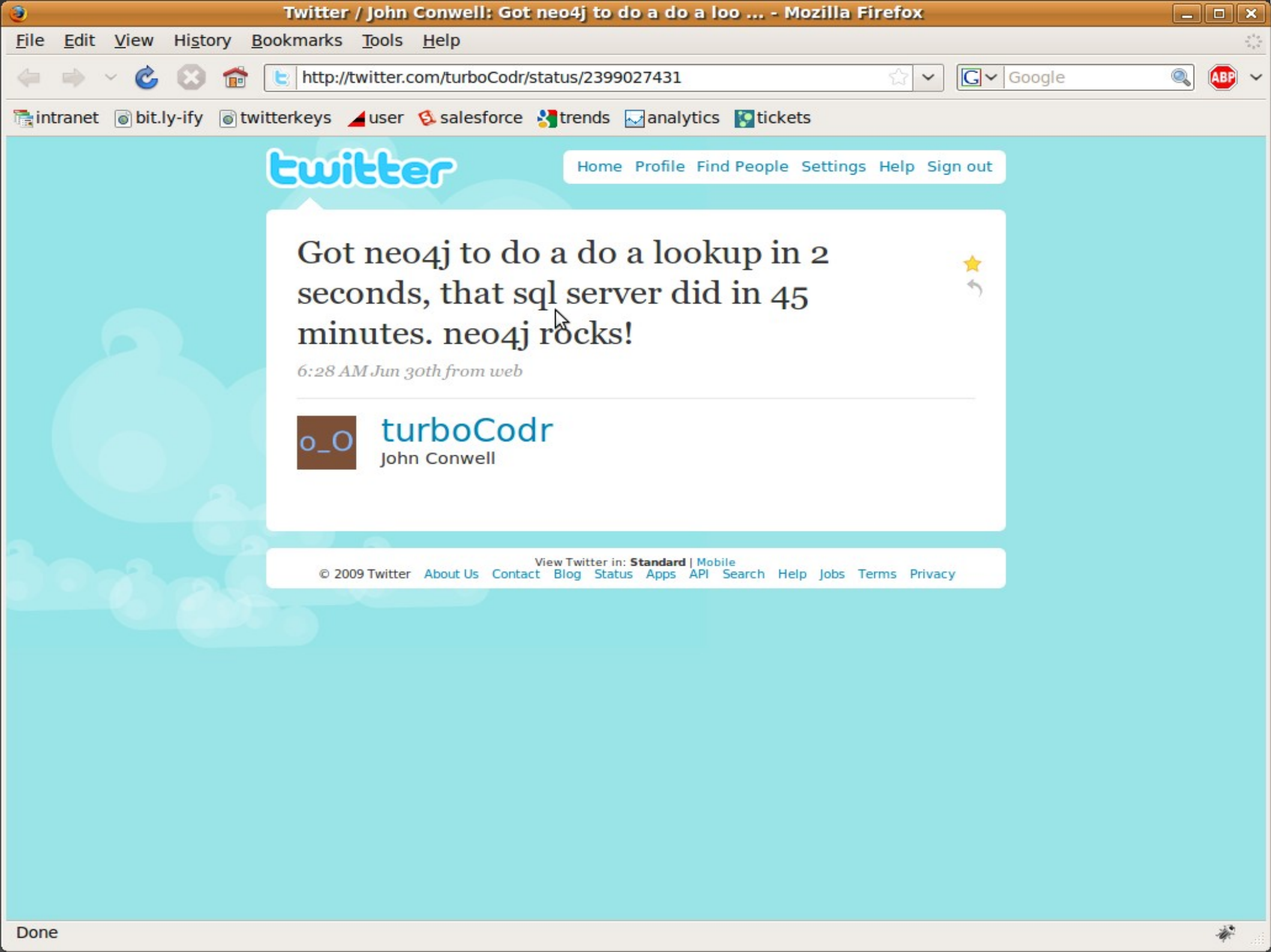


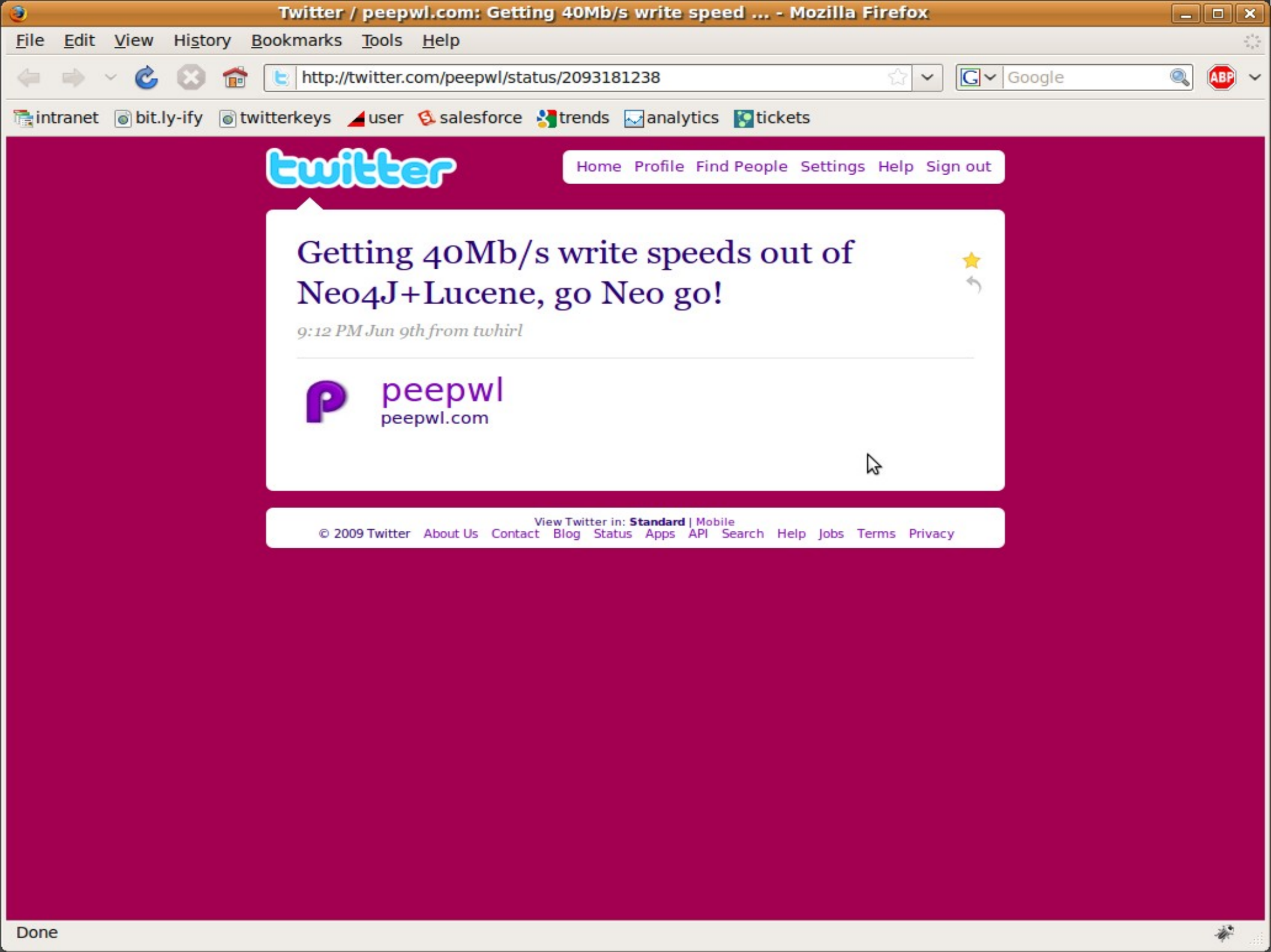
- ~1k persons
- Avg 50 friends per person
- pathExists(a, b) limit depth 4
- Two backends
- Eliminate disk IO so warm up caches

Social network *pathExists()*



	# persons	*query time*
Relational database		
Graph database (Neo4j)		
Graph database (Neo4j)		


Twitter / John Conwell: Got neo4j to do a do a loo ... - Mozilla Firefox
File Edit View History Bookmarks Tools Help
http://twitter.com/turboCodr/status/2399027431
Google
intranet bit.ly-ify twitterkeys user salesforce trends analytics tickets
twitter
Home Profile Find People Settings Help Sign out
Got neo4j to do a do a lookup in 2 seconds, that sql server did in 45 minutes. neo4j rocks!
6:28 AM Jun 30th from web
o_O
turboCodr
John Conwell
View Twitter in: Standard | Mobile
© 2009 Twitter About Us Contact Blog Status Apps API Search Help Jobs Terms Privacy
Done


Twitter / peepwl.com: Getting 40Mb/s write speed ... - Mozilla Firefox
File Edit View History Bookmarks Tools Help
http://twitter.com/peepwl/status/2093181238
Google
intranet bit.ly-ify twitterkeys user salesforce trends analytics tickets
twitter
Home Profile Find People Settings Help Sign out
Getting 40Mb/s write speeds out of Neo4J+Lucene, go Neo go!
9:12 PM Jun 9th from twhirl
peepwl
peepwl.com
View Twitter in: Standard | Mobile
© 2009 Twitter About Us Contact Blog Status Apps API Search Help Jobs Terms Privacy
Done

Pros & Cons compared to RDBMS

+ No O/R impedance mismatch (*whiteboard friendly*)
+ Can easily evolve schemas
+ Can represent semi-structured info
+ Can represent graphs/networks (*with* performance)

- Lacks in tool and framework support
- Few other implementations => potential lock in
- No support for ad-hoc queries

Query languages

- SPARQL – “SQL for linked data”
  - Ex:
```
"SELECT ?person WHERE {
        ?person neo4j:KNOWS ?friend .
        ?friend neo4j:KNOWS ?foe .
        ?foe neo4j:name “Larry Ellison” .
    }"
```
- Gremlin – “perl for graphs”
  - Ex: `"./outE[@label='KNOWS']/inV[@age > 30]/@name"`
  - Check it out at http://try.neo4j.org

The Neo4j ecosystem

- Neo4j is an embedded database
  - Tiny teeny lil jar f le
- Component ecosystem
  - index
  - meta-model
  - graph-matching
  - remote-graphdb
  - sparql-engine
  - ...
- See http://components.neo4j.org

Language bindings

- Neo4j.py – bindings for Jython and CPython
  - http://components.neo4j.org/neo4j.py
- Neo4jrb – bindings for JRuby (incl RESTful API)
  - http://wiki.neo4j.org/content/Ruby
- Clojure
  - http://wiki.neo4j.org/content/Clojure
- Scala (incl RESTful API)
  - http://wiki.neo4j.org/content/Scala
- **Nerlo** (really early release, like 4 am)
  - http://github.com/nerlo/nerlo

Scale out – replication

- Neo4j HA in internal beta testing with cust's now
- Master-slave replication, 1st configuration
  - MySQL style... ish
  - Except all instances can write, synchronously between writing slave & master (strong consistency)
  - Updates are asynchronously propagated to the other slaves (eventual consistency)
- This can handle billions of entities...
- … but not 100B

Scale out – partitioning

- “Sharding possible today”
  - … but you have to do manual work
  - … just as with MySQL
- Transparent partitioning? Neo4j 2.0
  - 100B? Easy to say. Sliiiiightly harder to do.
  - Fundamentals: BASE & eventual consistency
  - Generic clustering algorithm as base case, but give lots of knobs for developers

While we're on future stuff

- Neo4j 1.1 [July 2010]
  - Full JMX / SNMP support
  - Event framework
  - Traverser framework iteration
  - …
- Neo4j HA private beta now, GA before end of year
- REST-ful API (already out in 0.8, please give feedback for 1.0!)
- Framework integration: Roo, Grails
- Database integration: CouchDB, MySQL
- Neo4j Spatial (uDig integr, {R,Quad}-tree etc, OSM imports)
- Client tools: Webling, Neoclipse, Monitoring console

How ego are you? (aka other impls?)

- Franz' **AllegroGraph** (http://agraph.franz.com)
  - Proprietary, Lisp, RDF-oriented but real graphdb
- Sones **graphDB** (http://sones.com)
  - Proprietary, .NET, in beta
- Twitter's **FlockDB** (http://github.com/twitter/flockdb)
  - Twitter's graph database for large and shallow graphs
- Google **Pregel** (http://bit.ly/dP9IP)
  - We are oh-so-secret
- Some academic papers from ~10 years ago
  - G = {V, E} #FAIL

Conclusion

- Graphs && Neo4j => teh awesome!
- Available NOW under AGPLv3 / commercial license
  - AGPLv3: “if you’re open source, we’re open source”
  - If you have proprietary software? Must buy a commercial license
  - But the first one is free!
- Download
  - http://neo4j.org
- Feedback
  - http://lists.neo4j.org

Twitter / julianbrowne: Going to play with Neo4j t ... - Mozilla Firefox

File Edit View History Bookmarks Tools Help

http://twitter.com/julianbrowne/status/2455417887

Google

intranet bit.ly-ify twitterkeys user salesforce trends analytics tickets

:julianbrowne twitter

Home Profile Find People Settings Help Sign out

Going to play with Neo4j this w/e. Seems to me that even after arguments about ACID/scale/CAP it's just more human & agile to be graph-based

5:42 PM Jul 3rd from web

julianbrowne

View Twitter in: **Standard** | Mobile

 About Us Contact Blog Status Apps API Search Help Jobs Terms Privacy

Done

Questions?

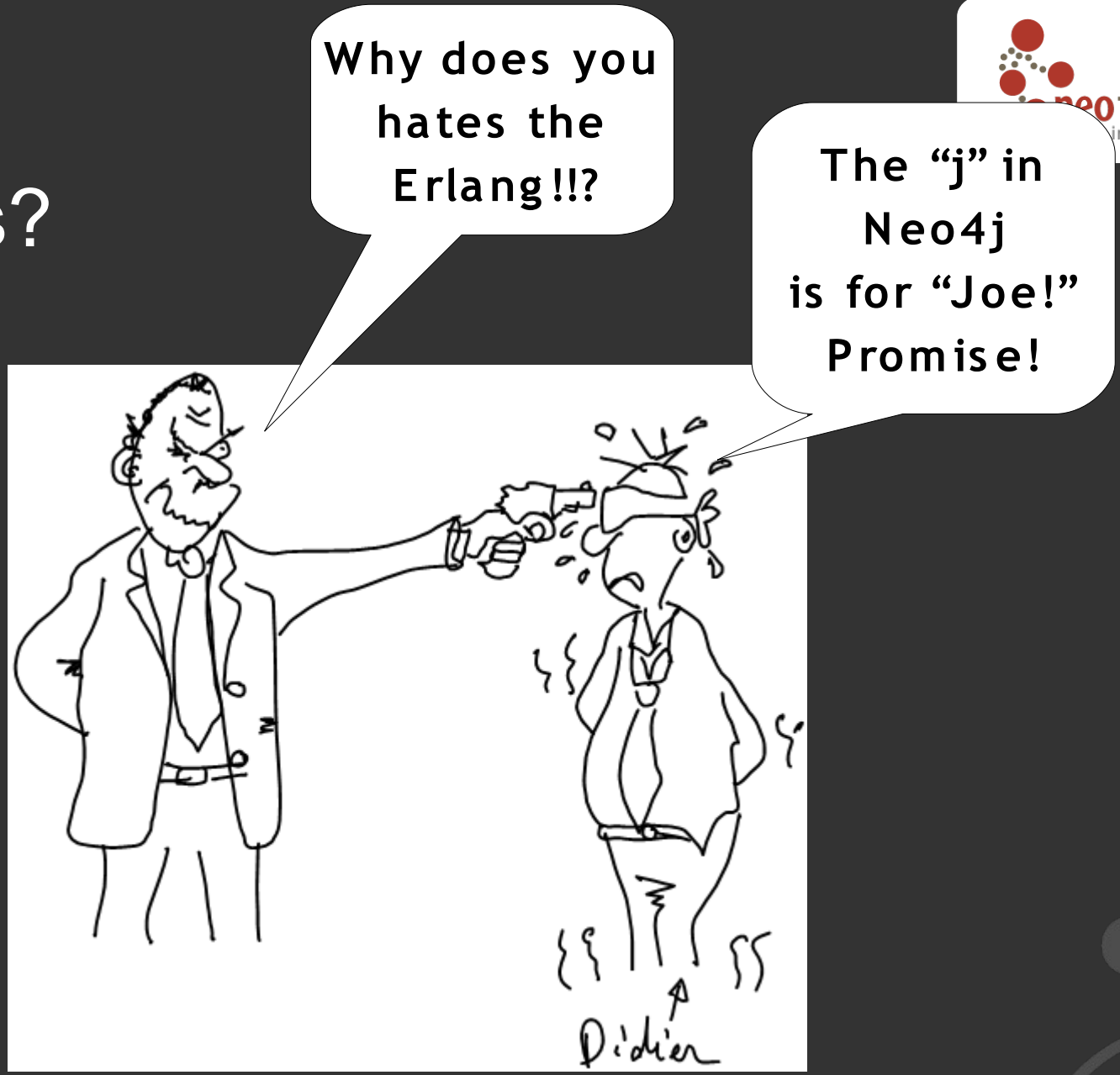


Image credit: lost again! Sorry :(