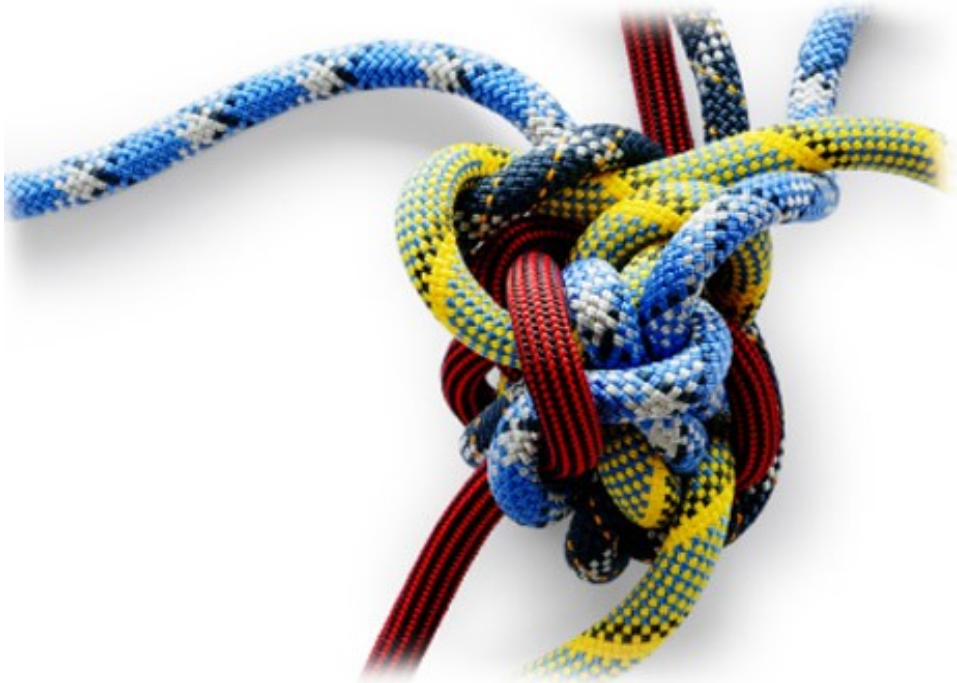
Связи решают все!

Евгений Газдовский

Animotron.org

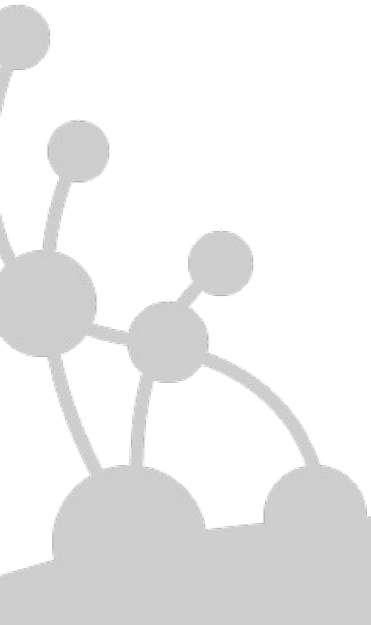
neotechnology
value in relationships

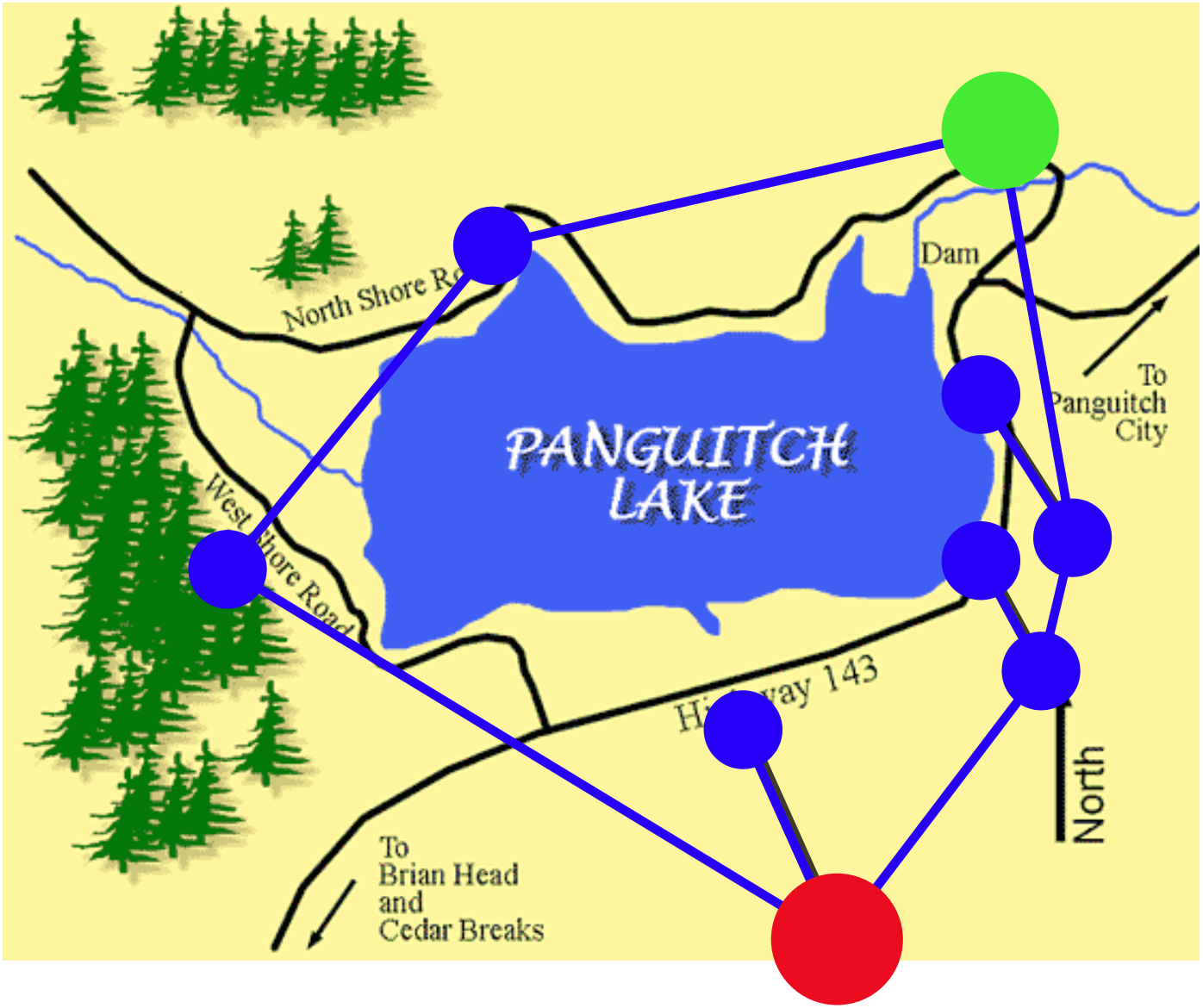
Если имеем сложные запутанные модели

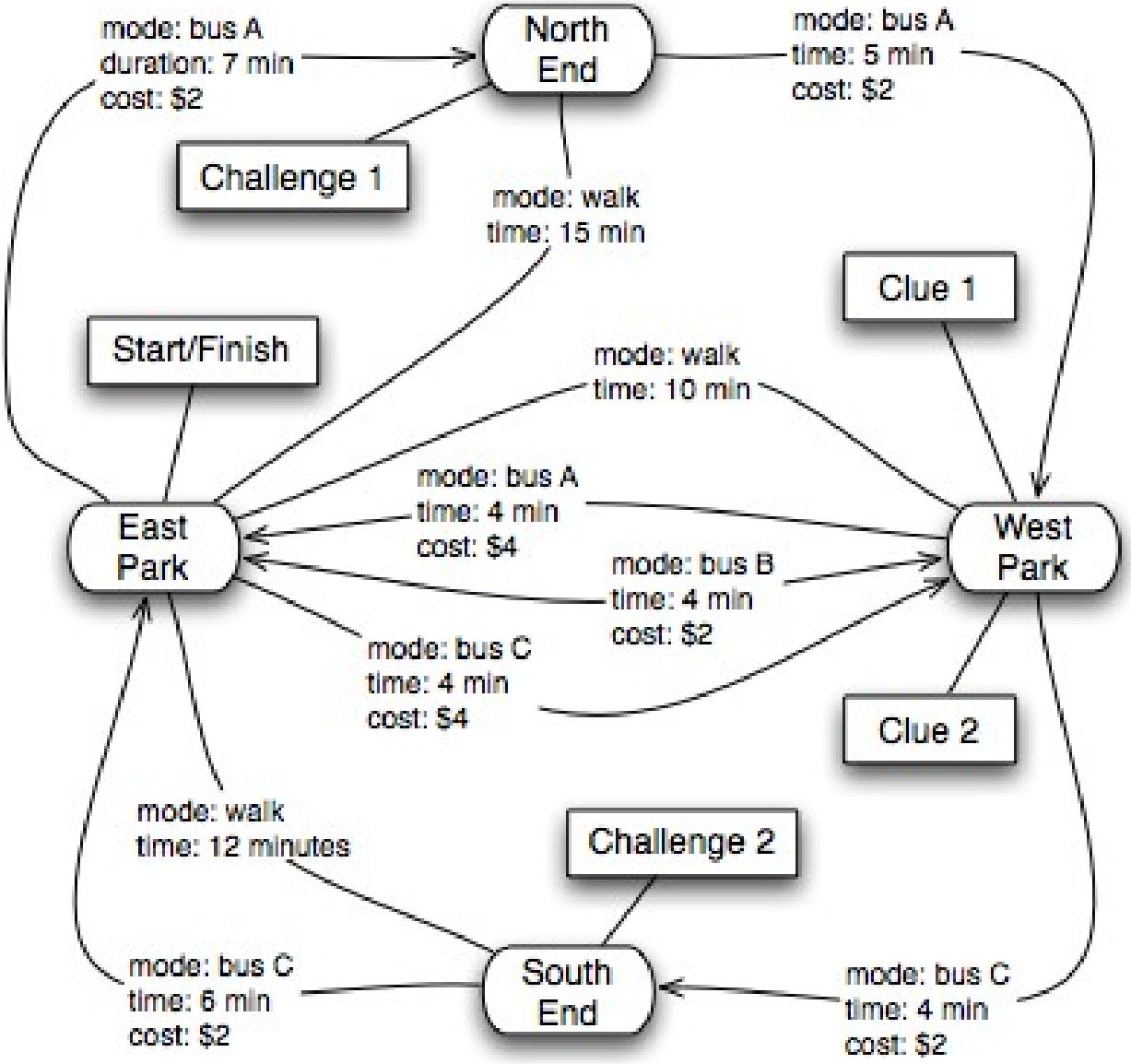


neo technology
value in relationships

Гео информационные системы

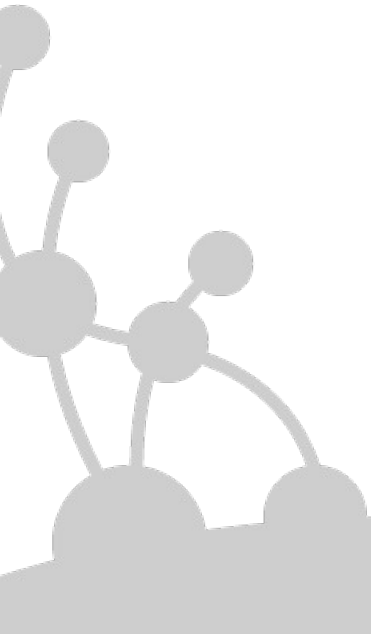



Dam
To
Panguitch
City
PANGUITCH
LAKE
West Shore Road
Highway 143
North
To
Brian Head
and
Cedar Breaks


mode: bus A
duration: 7 min
cost: $2
North End
mode: bus A
time: 5 min
cost: $2
Challenge 1
mode: walk
time: 15 min
Clue 1
Start/Finish
mode: walk
time: 10 min
mode: bus A
time: 4 min
cost: $4
East Park
West Park
mode: bus B
time: 4 min
cost: $2
mode: bus C
time: 4 min
cost: $4
Clue 2
mode: walk
time: 12 minutes
Challenge 2
mode: bus C
time: 6 min
cost: $2
South End
mode: bus C
time: 4 min
cost: $2

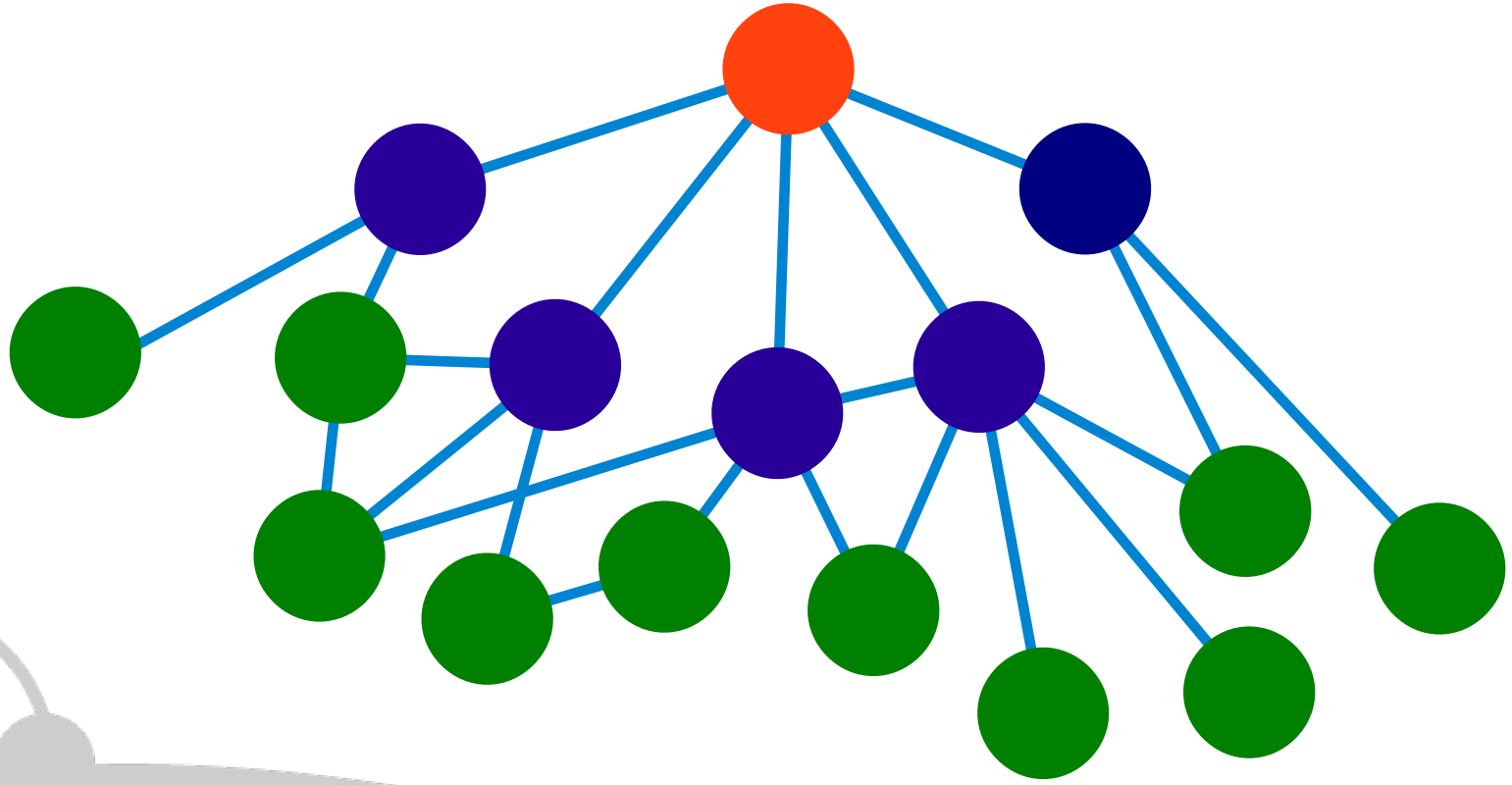
neotechnology
value in relationships

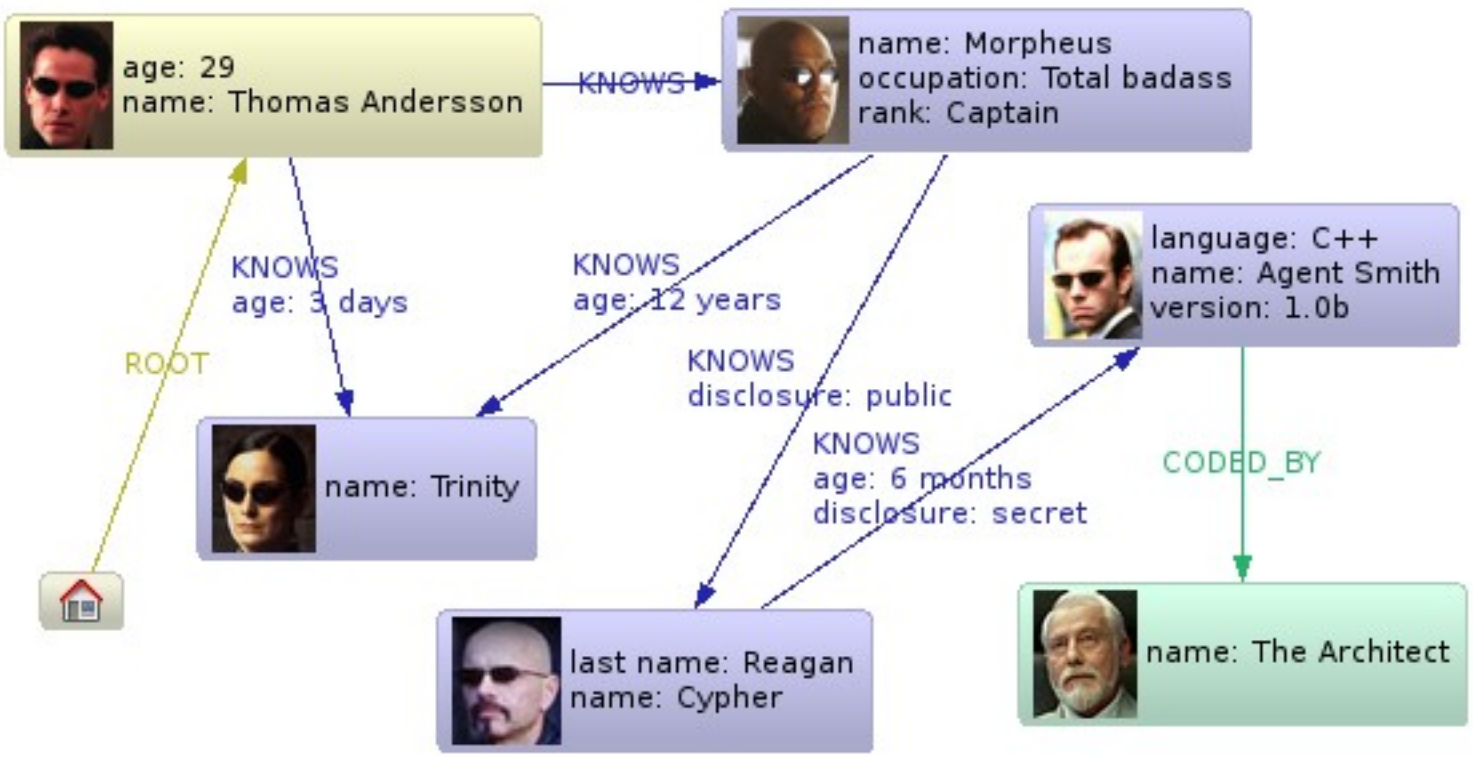
Социальные сети



FOAF

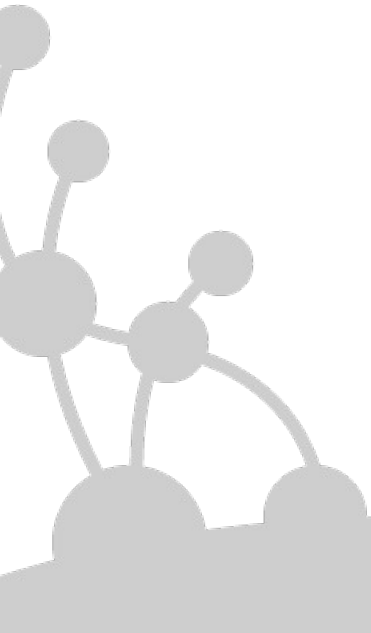
Я, друзья, друзья друзей

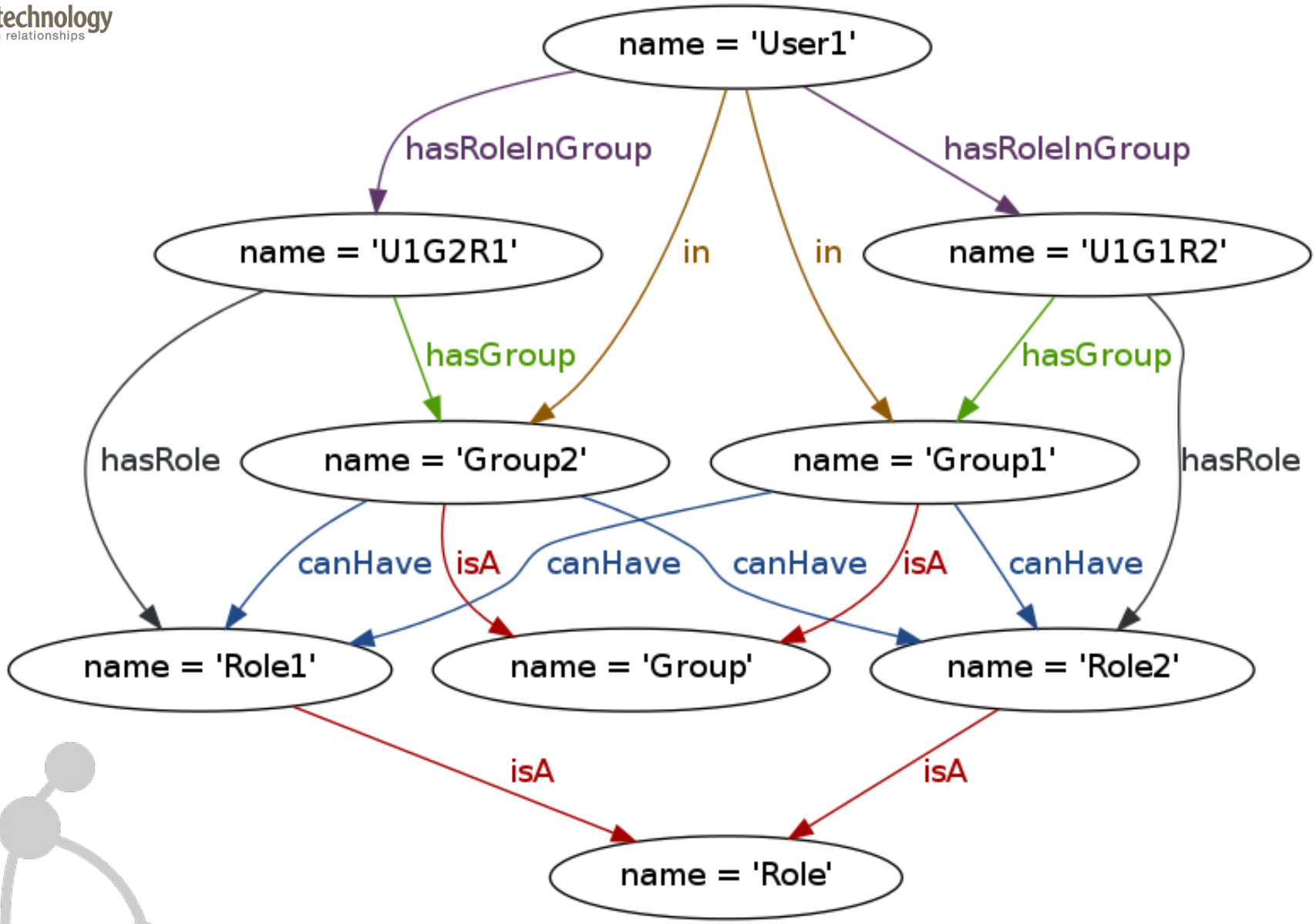



age: 29
name: Thomas Andersson
KNOWS
name: Morpheus
occupation: Total badass
rank: Captain
KNOWS
age: 3 days
KNOWS
age: 12 years
language: C++
name: Agent Smith
version: 1.0b
ROOT
KNOWS
disclosure: public
KNOWS
age: 6 months
disclosure: secret
CODED_BY
name: Trinity
last name: Reagan
name: Cypher
name: The Architect

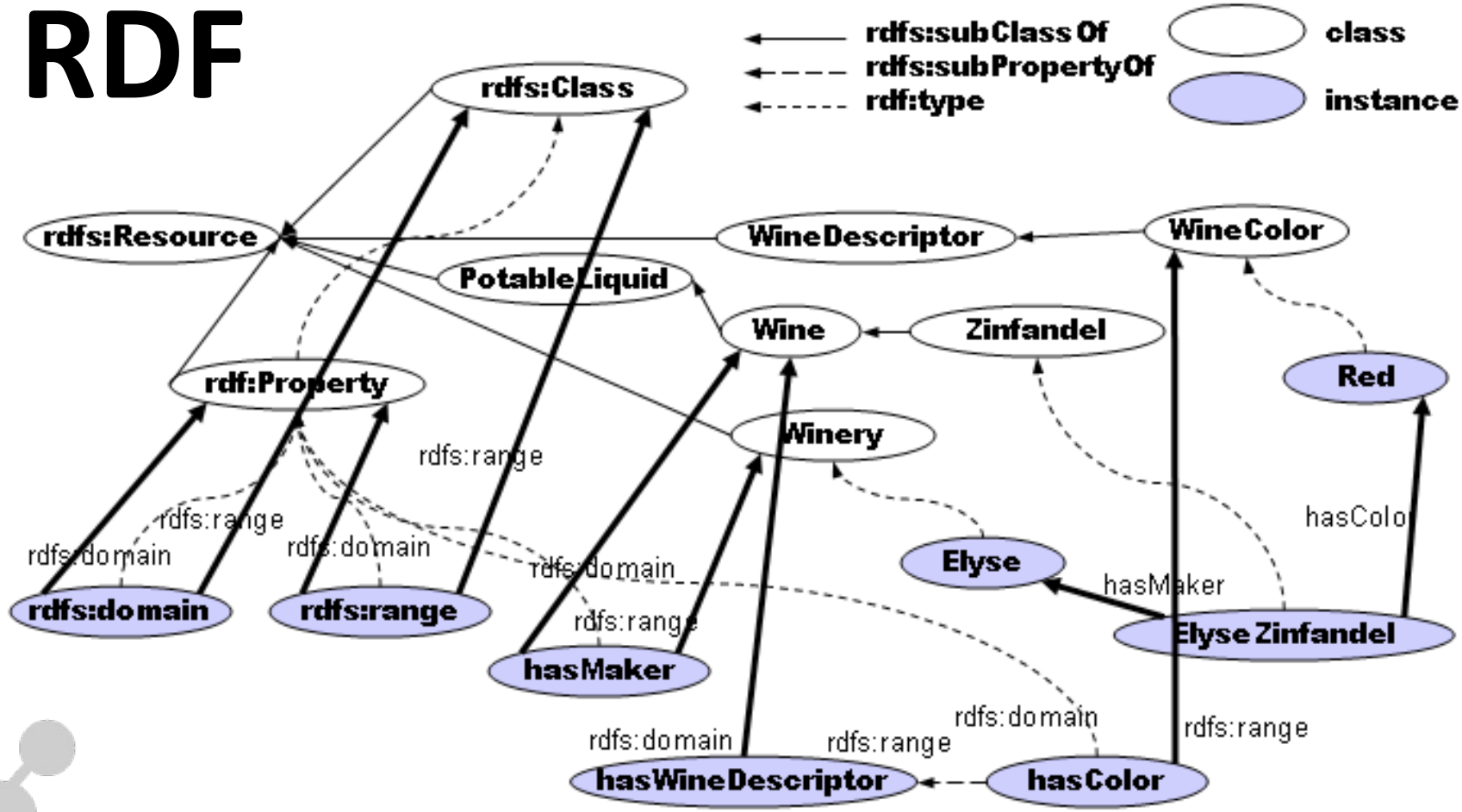
neo technology
value in relationships

Онтологии




name = 'User1'
hasRoleInGroup
hasRoleInGroup
name = 'U1G2R1'
in
in
name = 'U1G1R2'
hasGroup
hasGroup
hasRole
name = 'Group2'
name = 'Group1'
hasRole
canHave
isA
canHave
canHave
isA
canHave
name = 'Role1'
name = 'Group'
name = 'Role2'
isA
isA
name = 'Role'

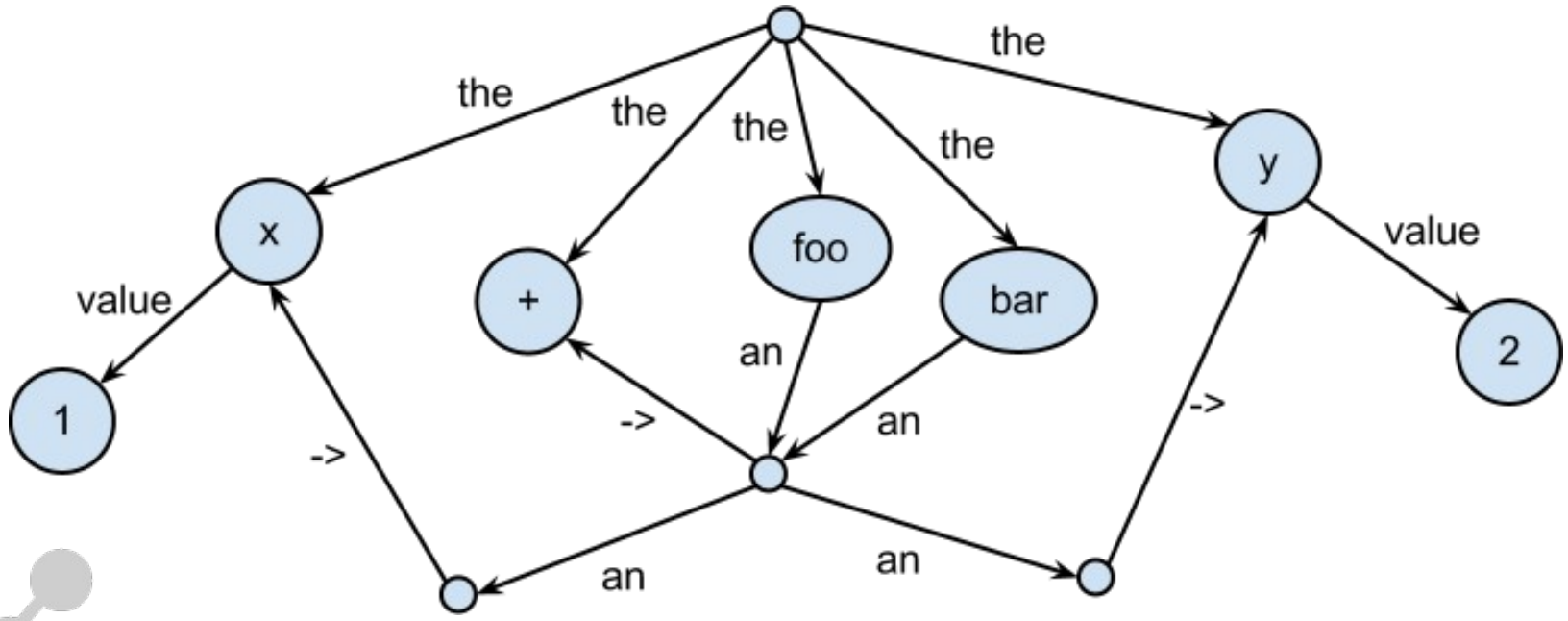
RDF



rdfs:domain: class that is reachable from a start node of a property edge via rdf:type→rdfs:subClassOf
rdfs:range: class that is reachable from a target node of a property edge via rdf:type→rdfs:subClassOf

neotechnology
value in relationships

Animo



neotechnology
value in relationships

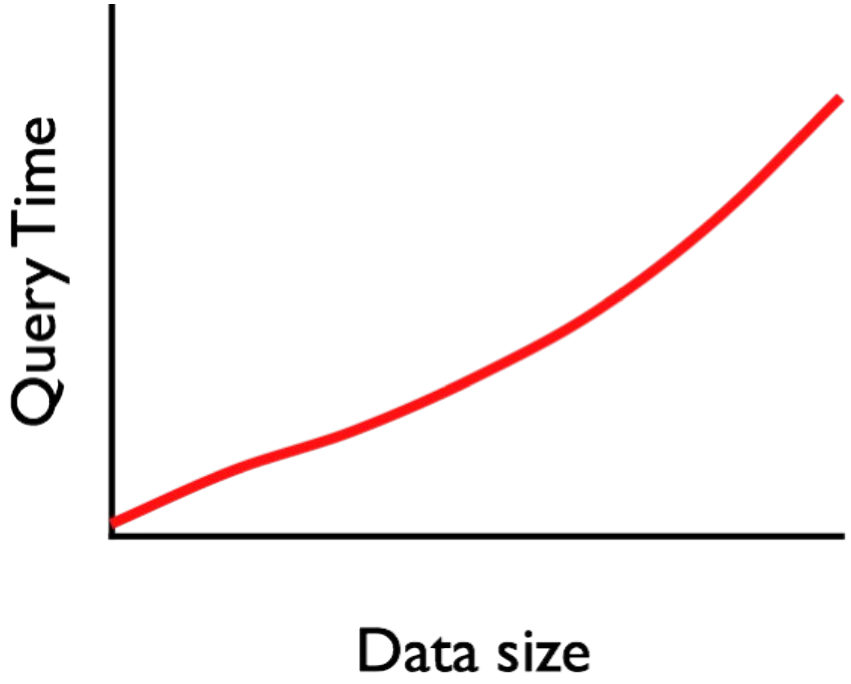
Не хотим головной боли от RDBMS

neotechnology
value in relationships

Моделирование отношений в реляционных СуБД

Уменьшение производительности

во многих случаях



neotechnology
value in relationships

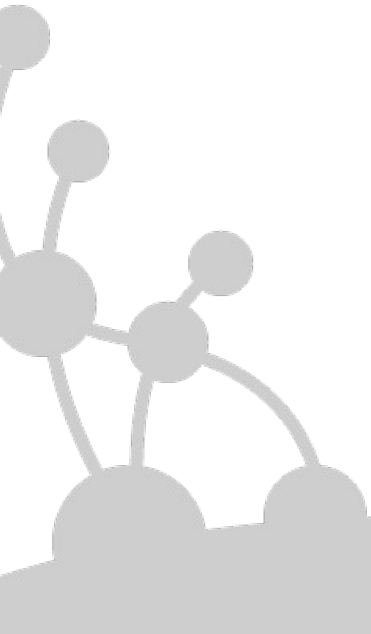
Альтернативы реляционным СуБД

Key/value

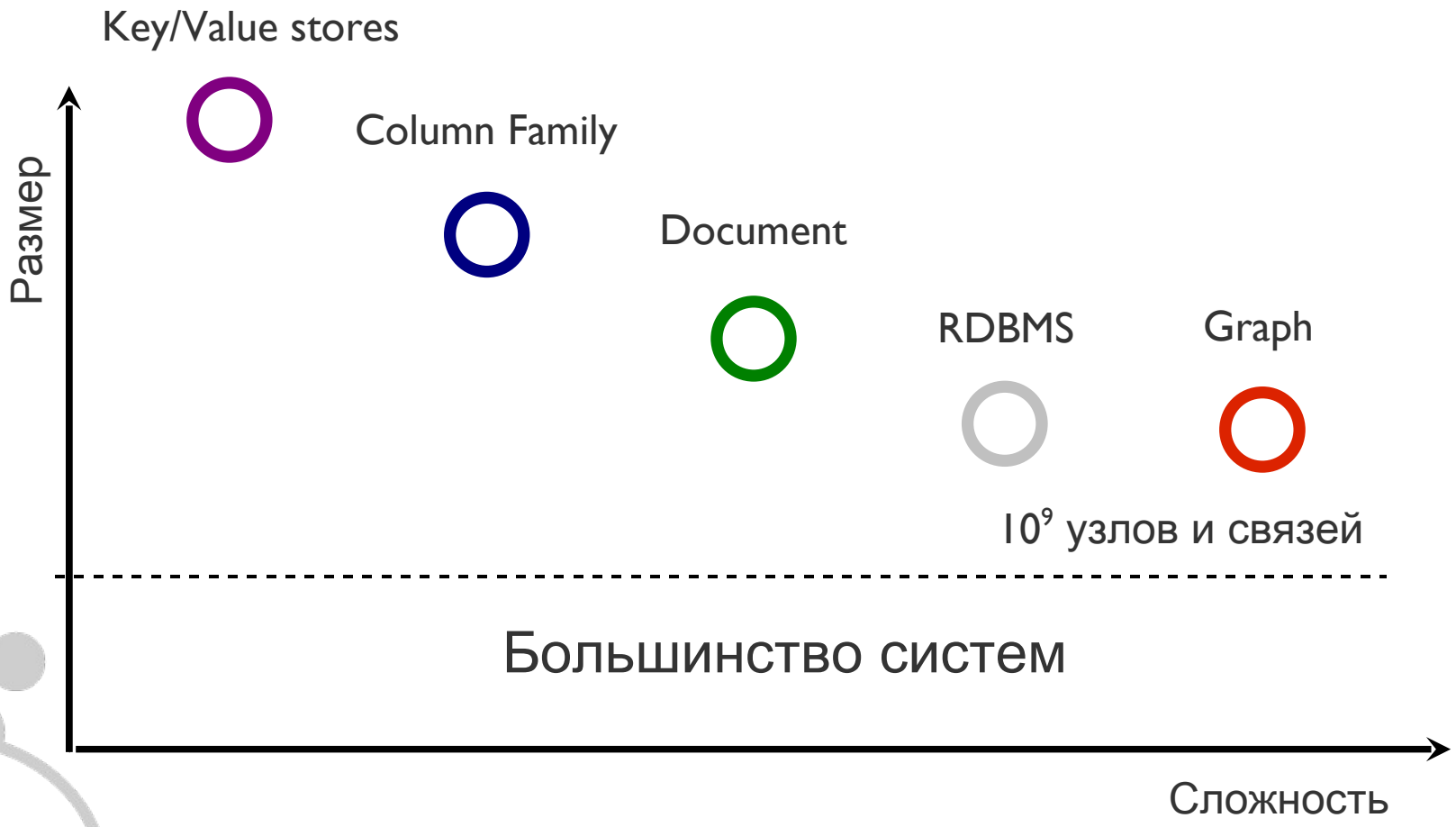
Document

Column Family

Graph



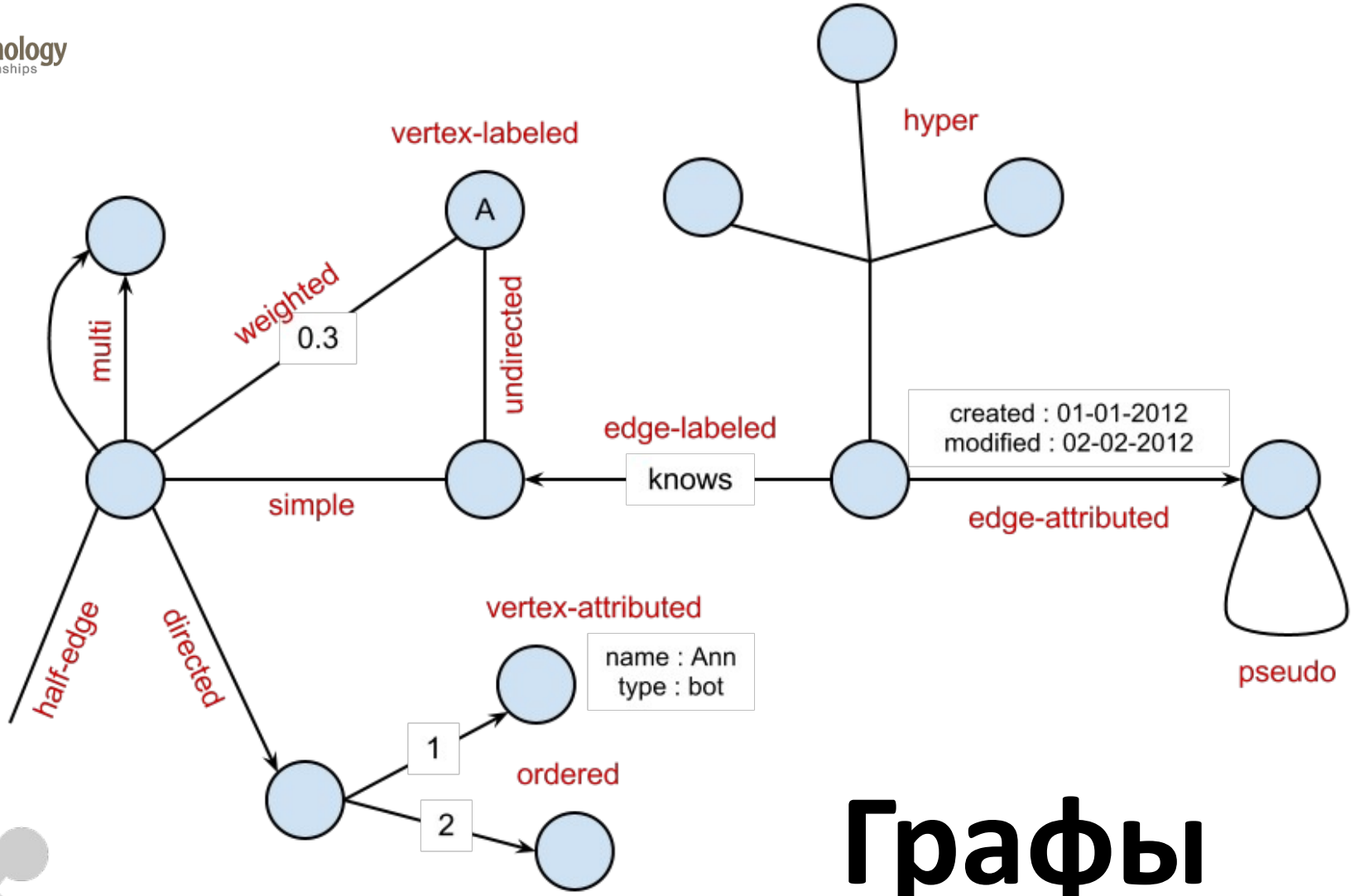
Сложность / размер



Для каждого варианта своё!

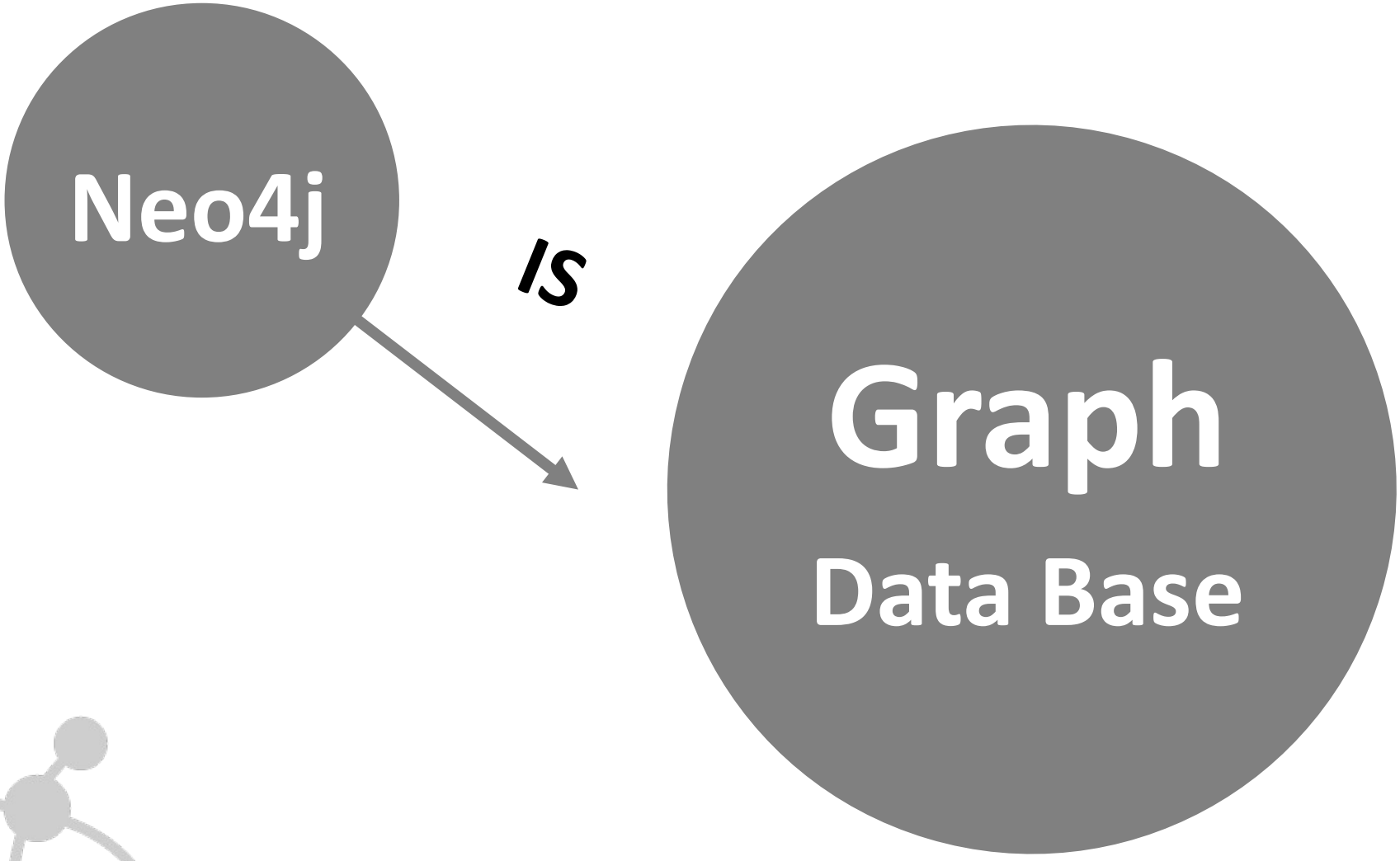
Эпоха использования реляционных СУБД для всех задач закончилась.

neo technology
value in relationships



Графы

neotechnology
value in relationships

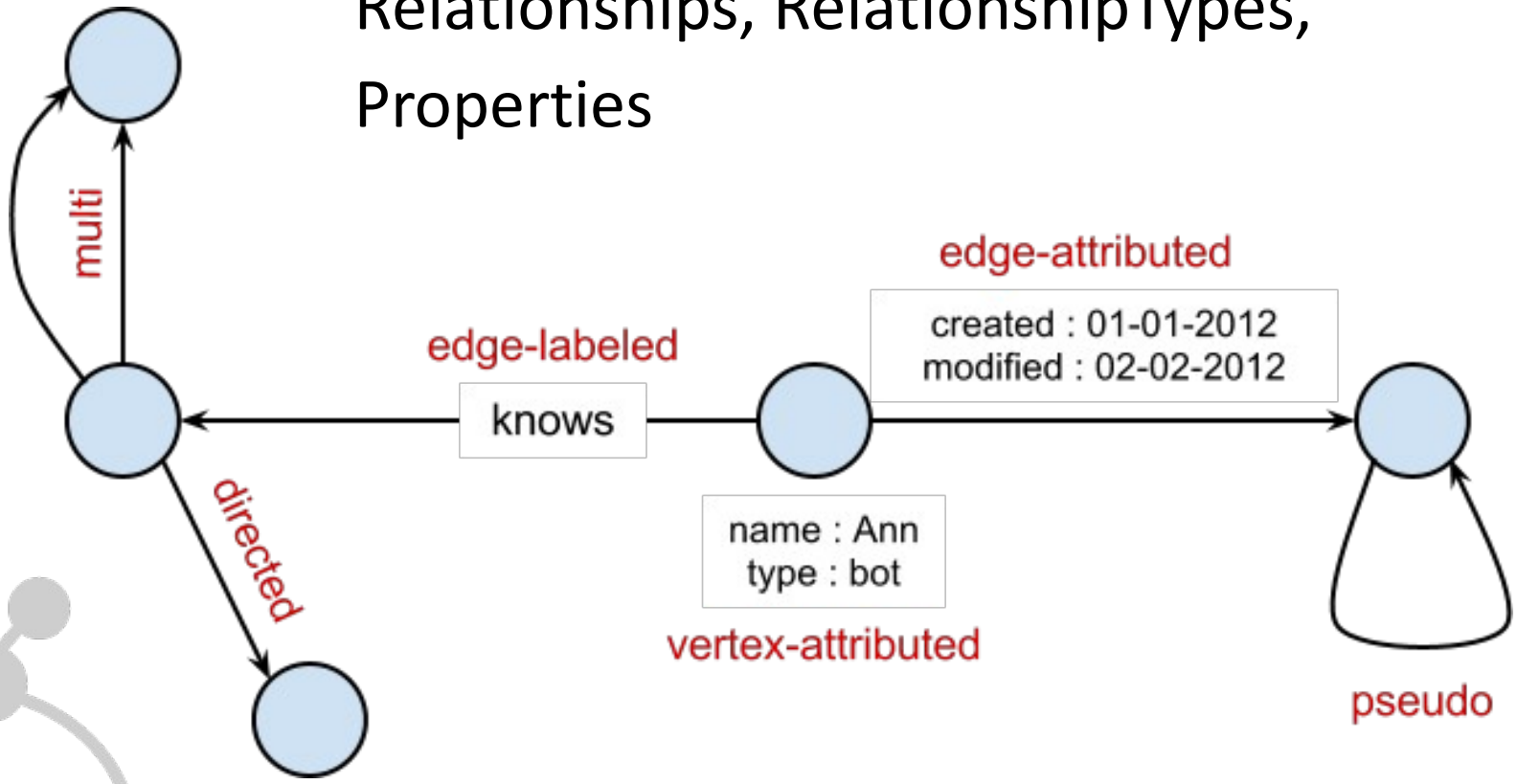


Neo4j — это

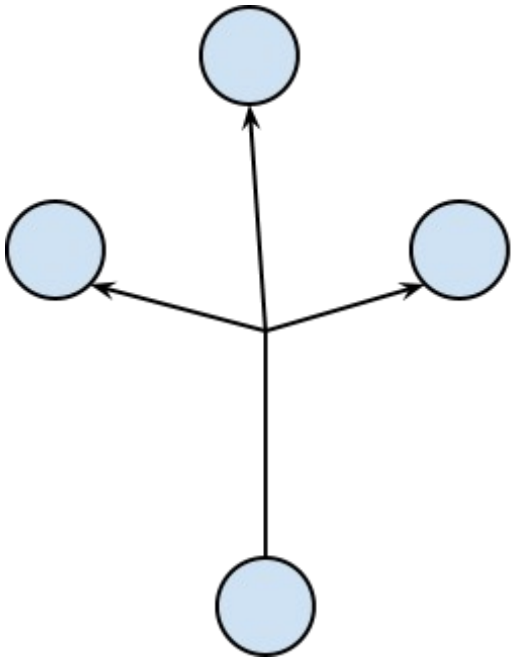
Nodes,

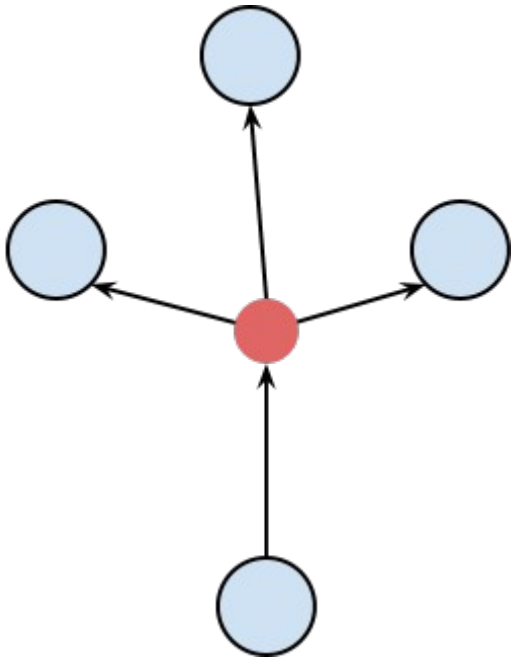
Relationships, RelationshipTypes,

Properties



Моделирование гиперграфа

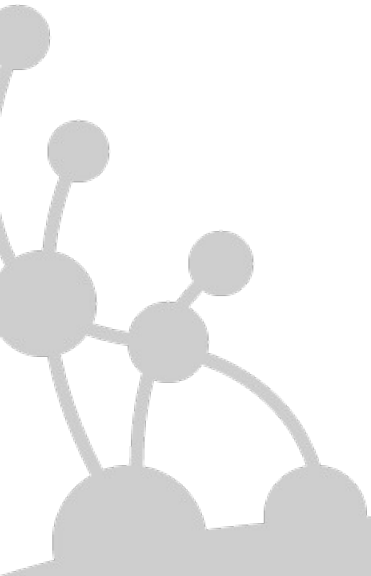




neotechnology
value in relationships

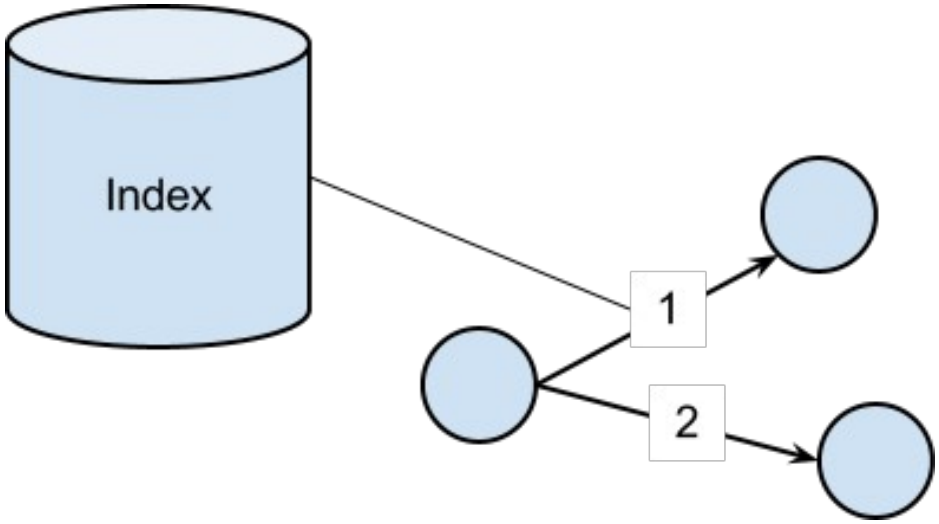
Enhanced Neo4j API

Vertexes, Edges



neotechnology
value in relationships

Моделирование упорядоченого графа

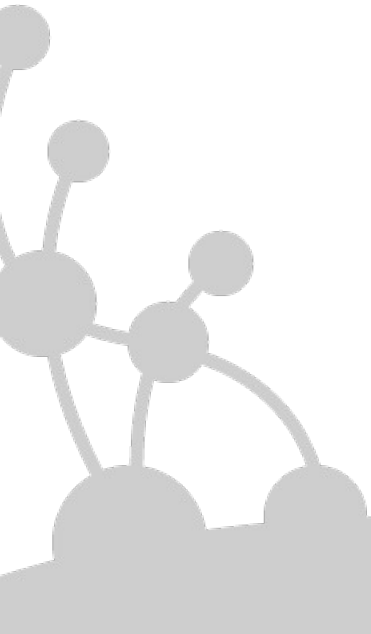


neotechnology
value in relationships

Взаимодействие с Neo4j

Embedded

REST API



neotechnology
value in relationships

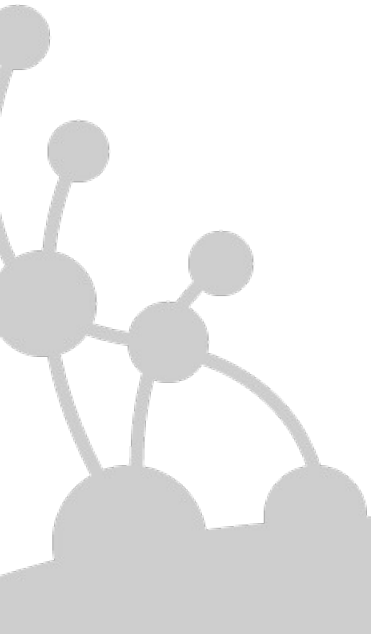
Взаимодействие с Neo4j

Java, Ruby, Python API

Blueprints, Gremlin

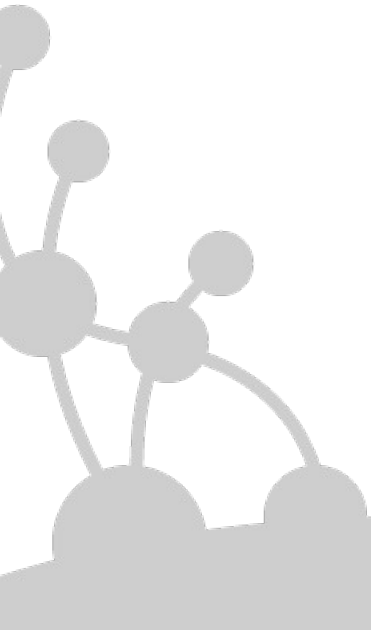
Cypher Query

Spring



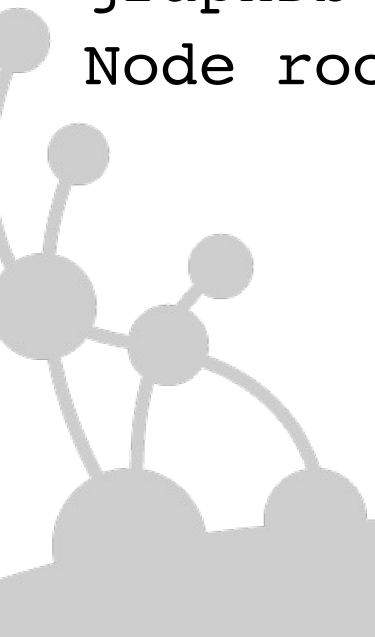
neotechnology
value in relationships

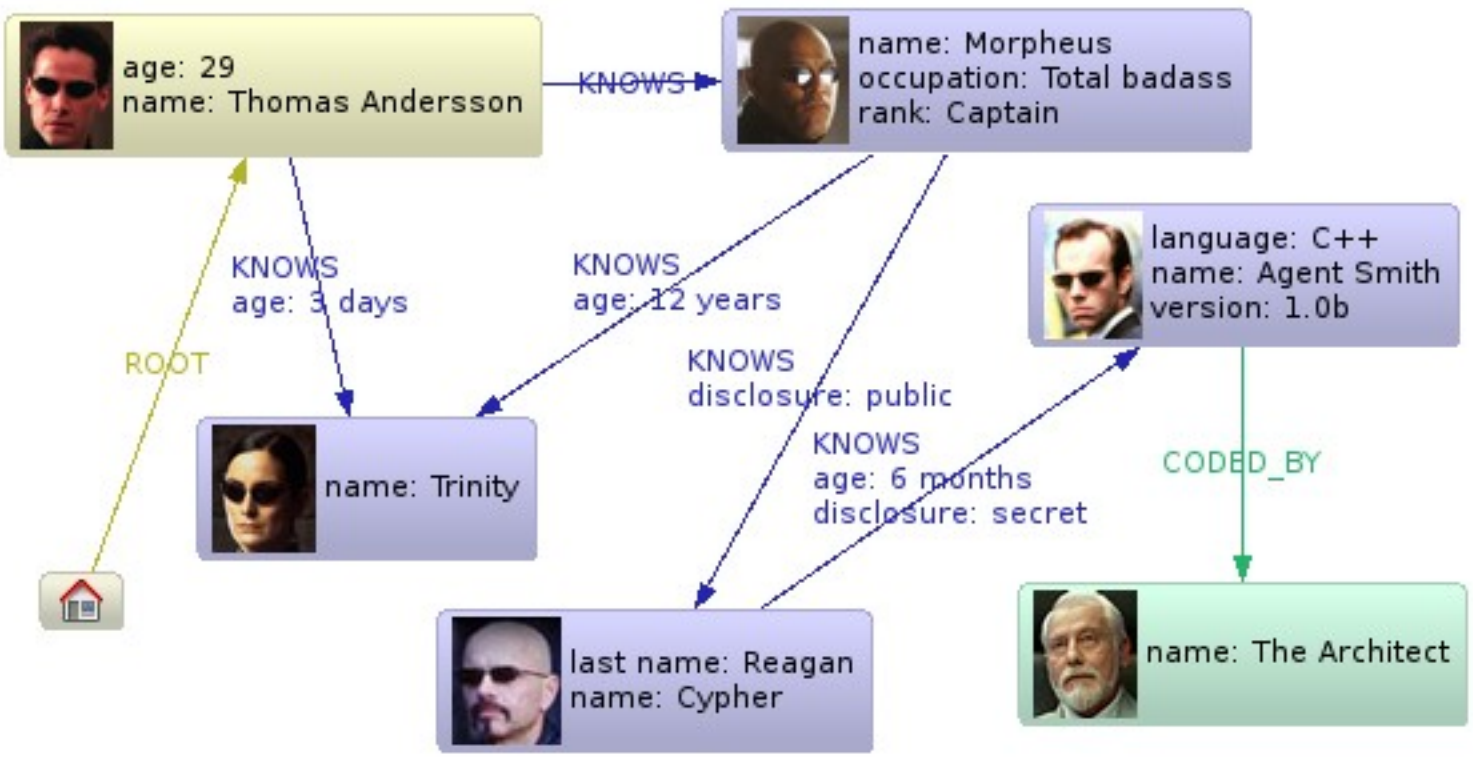
Embedded Java API



Создание БД

```
String STORAGE = "/home/foo/data";
EmbeddedGraphDatabase graphDb;
graphDb = new EmbeddedGraphDatabase(STORAGE);
Node root = graphDb.getReferenceNode();
```




age: 29
name: Thomas Andersson
KNOWS
name: Morpheus
occupation: Total badass
rank: Captain
KNOWS
age: 3 days
KNOWS
age: 12 years
language: C++
name: Agent Smith
version: 1.0b
ROOT
KNOWS
disclosure: public
KNOWS
age: 6 months
disclosure: secret
CODED_BY
name: Trinity
last name: Reagan
name: Cypher
name: The Architect

Добавление узлов

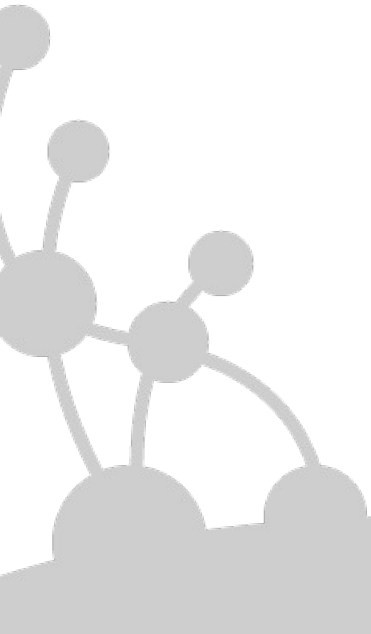
```
Node neo = graphDb.createNode();
neo.setProperty("name", "Thomas Anderson");
neo.setProperty("age", 29);
Node trinity = graphDb.createNode();
trinity.setProperty("name", "Trinity");
```

Типы связей

```
enum NeoRelType implements RelationshipType
{
    ROOT, KNOWS, CODED_BY
}
```

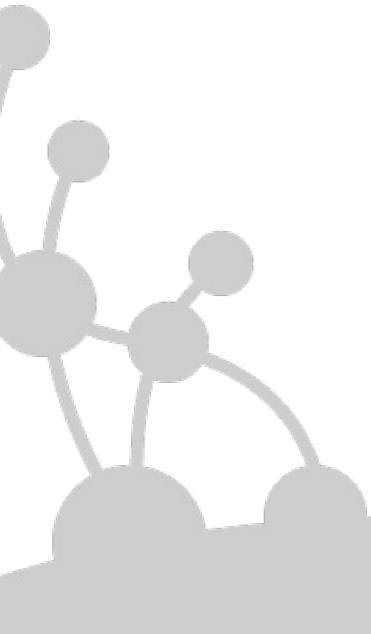
Добавление связей

```
root.createRelationshipTo(neo, ROOT);
Relationship r;
r = neo.createRelationshipTo(trinity, KNOWS);
r.setProperty("age", "3 days")
```



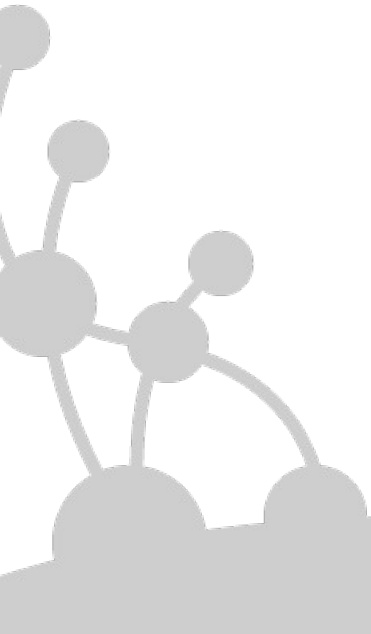
Начало и конец

```
Node start = r.getStartNode();
Node end = r.getEndNode();
Node n = r.getOtherNode(start);
```



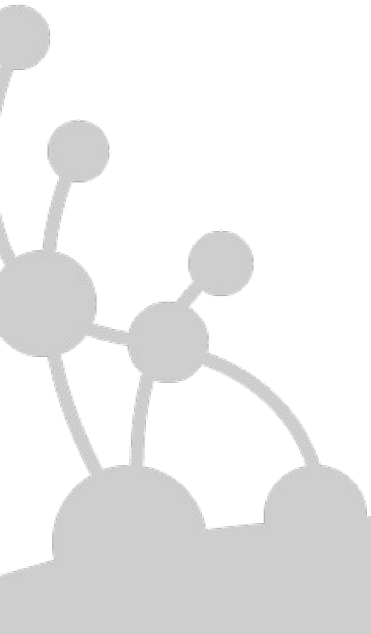
Связи

```
Relationship x = neo.getSingleRelationship(Direction.OUTGOING);
Relationship y = trinyty.getSingleRelationship(Direction.INCOMING);
Itreator<Relationship> it = neo.getRelationships(ROOT);
```



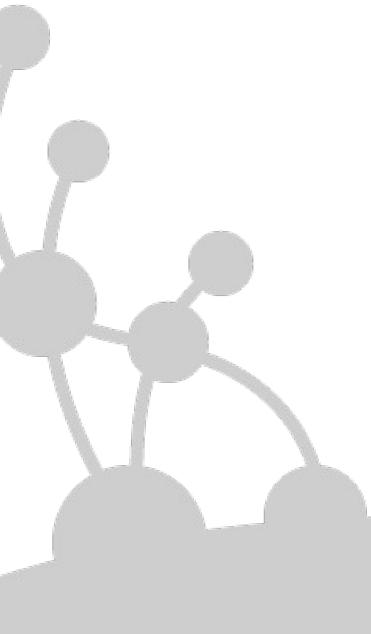
Свойства

```
boolean hasName = neo.hasPropperty("name")
String name = (String)neo.getProperty("name");
String age = (String)r.getProperty("age");
```



Удаление узлов, связей

```
r.delete(); trinity.delete();
```

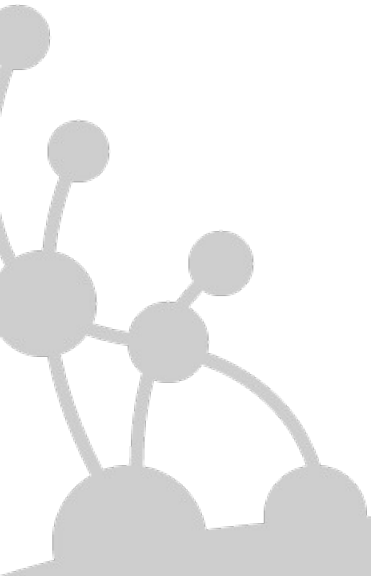


neotechnology
value in relationships

Фреймворк построения индексов

Relationship

Node



neo technology
value in relationships

Фреймворк построения индексов

Property

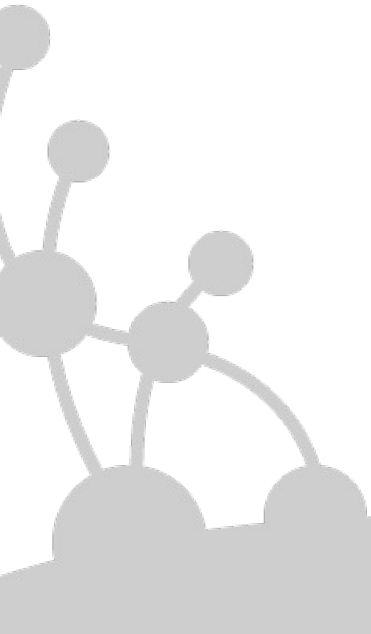
Произвольный ключ

neotechnology
value in relationships

Фреймворк построения индексов

Lucene

Berkeley Database



neotechnology
value in relationships

Полнотекстовый поиск

создание индекса Lucene

```
IndexManager index = graphDb.index();
Index<Node> fulltextMovies =
index.forNodes( "movies-fulltext",
    MapUtil.stringMap( IndexManager.PROVIDER,
    "lucene", "type", "fulltext" ) );
```

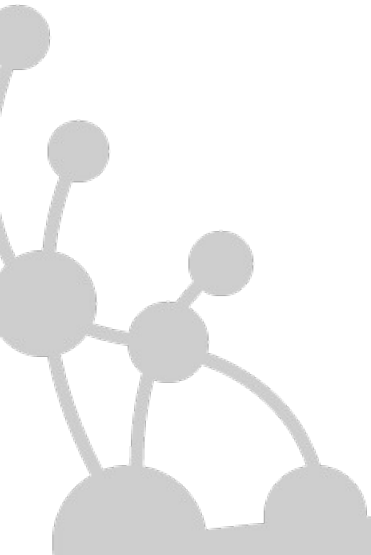
Полнотекстовый поиск

добавление

```
fulltextMovies.add( theMatrix, "title",
                                "The Matrix" );
fulltextMovies.add( theMatrixReloaded,
"title", "The Matrix Reloaded" );
```

Собственно поиск

```
Node found = fulltextMovies.query( "title",
        "reloaded" ).getSingle();
```



neotechnology
value in relationships

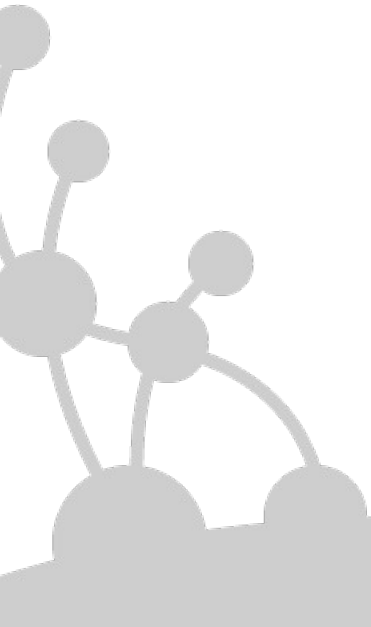
Все возможности Lucene

сортировка

числовые диапазоны

маски

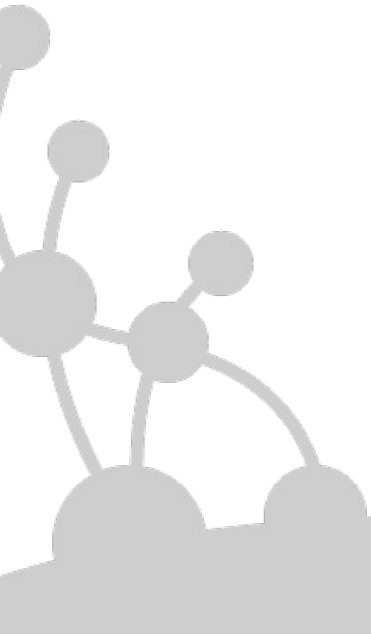
составные запросы



neotechnology
value in relationships

Автоматическое индексирование

через имена и значения свойств

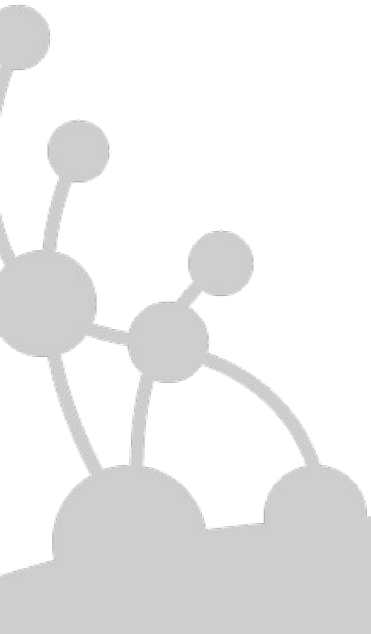


neotechnology
value in relationships

Запросы в Neo4j — это

Поиск узлов и связей по индексу

Траверс графа

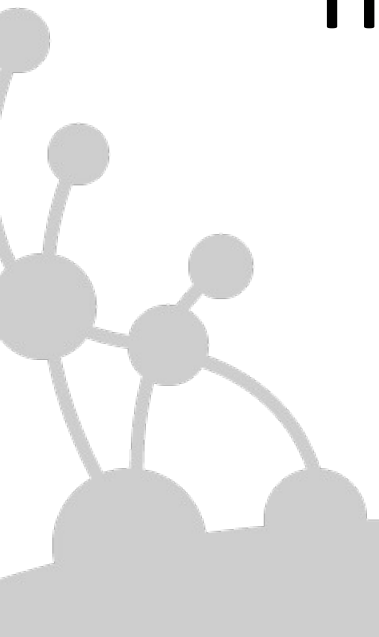


neotechnology
value in relationships

Траверс графа — это

поиск множества путей

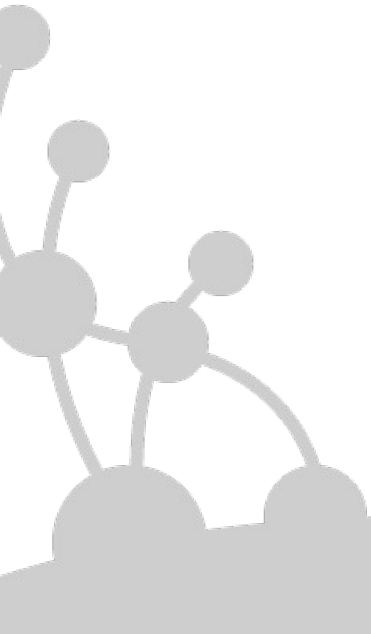
по заданным условиям



neotechnology
value in relationships

Траверс графа

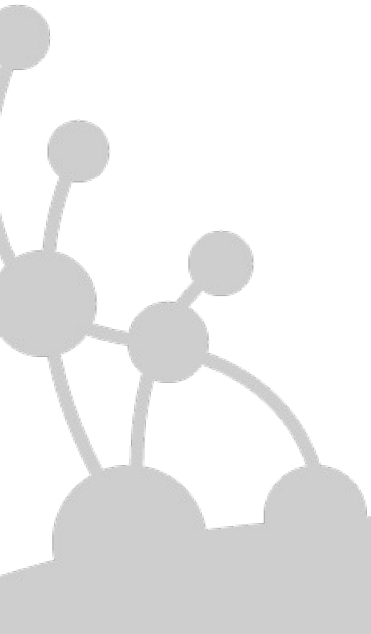
```
TraversalDescription td_KNOWS =
        Traversal.description().
           breadthFirst().
           relationships(KNOWS, OUTGOING);
```



neotechnology
value in relationships

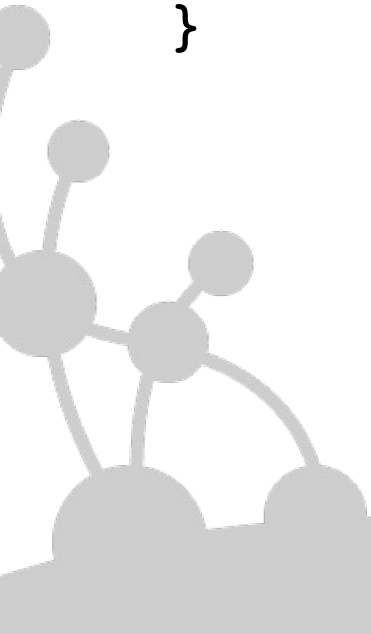
Траверс графа

```
Traverce t = td_KNOWS.traverce(neo);
```



Траверс графа

```
Iterateriable<Nodes> nodes = t.nodes();
for (Node n : nodes) {
   …
}
```



Траверс графа

```
Iteratrable<Relationship> relationships;
relationships = t.relationships();
for (Relationship n : relationships) {
   …
}
```

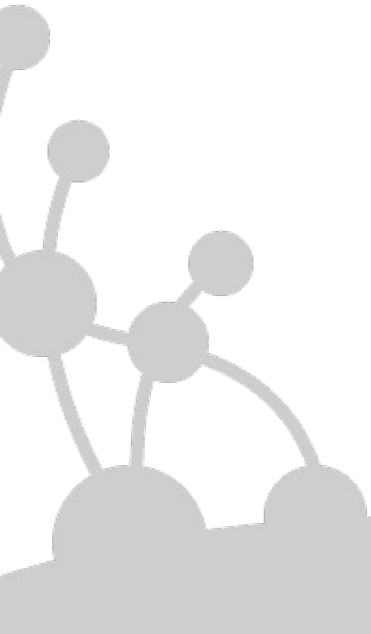
neotechnology
value in relationships

Траверс графа

```
Path p;
Iterator<Path> paths = t.iterator();
while (paths.hasNext()) {
    //…
    p = paths.next();
    //…
}
```

neotechnology
value in relationships

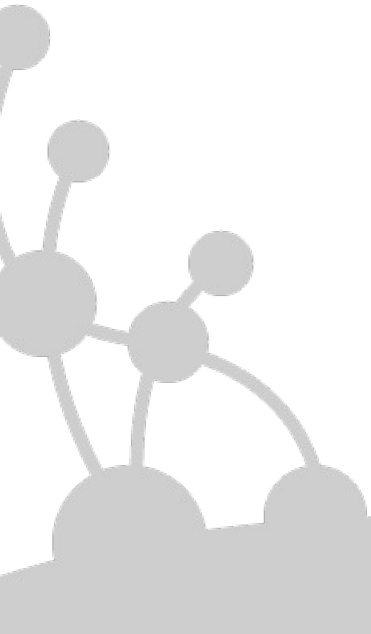
Все алгоритмы ленивые и могут быть распараллелены



neotechnology
value in relationships

Встроенные алгоритмы обхода графа

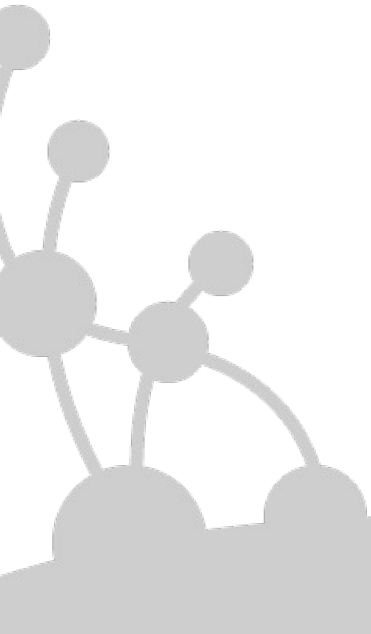
в глубину и ширину



neotechnology
value in relationships

Коллекция алгоритмов поиска путей

Дейкстры, A*,
всех путей



neotechnology
value in relationships

Транзакции

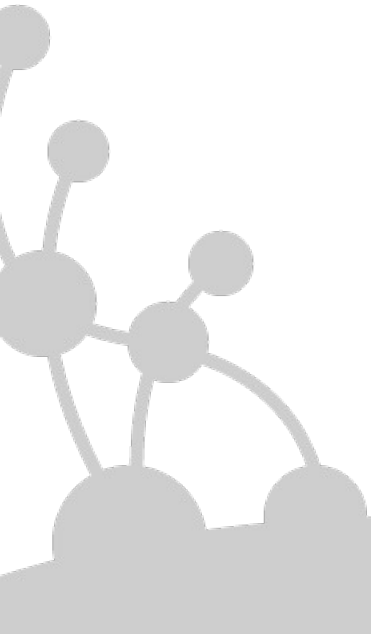
ACID

Многопоточность

neo technology
value in relationships

Транзакции

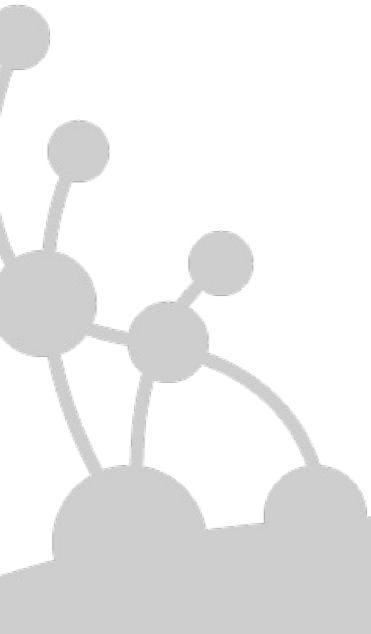
привязаны к thread



neo technology
value in relationships

Транзакции

только на запись



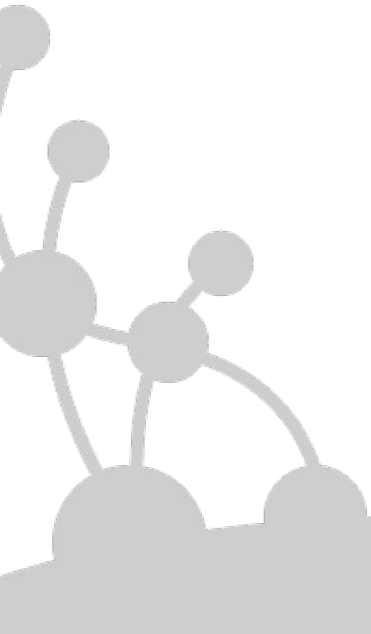
neotechnology
value in relationships

Транзакции

```
Transaction tx = graphDb.beginTx();
try {
   Node n = graphDb.createNode();
   //…
   tx.success();
} finally {
   tx.finish()
}
```

Транзакции

могут быть вложенными

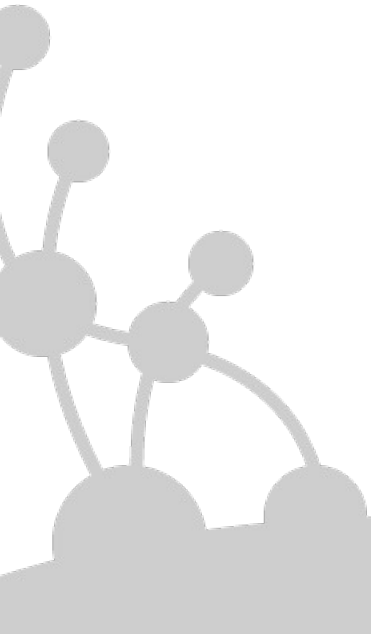


neotechnology
value in relationships

Транзакции

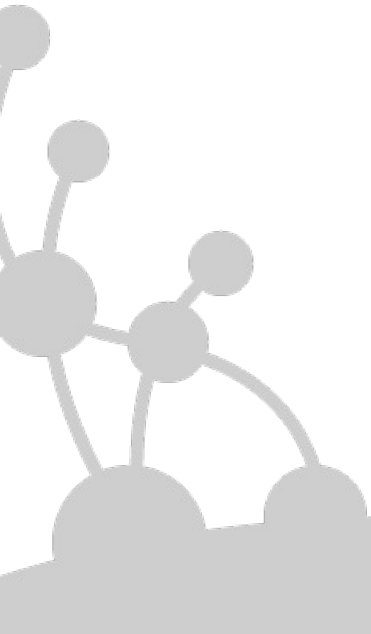
оптимальный размер

до 50К изменений

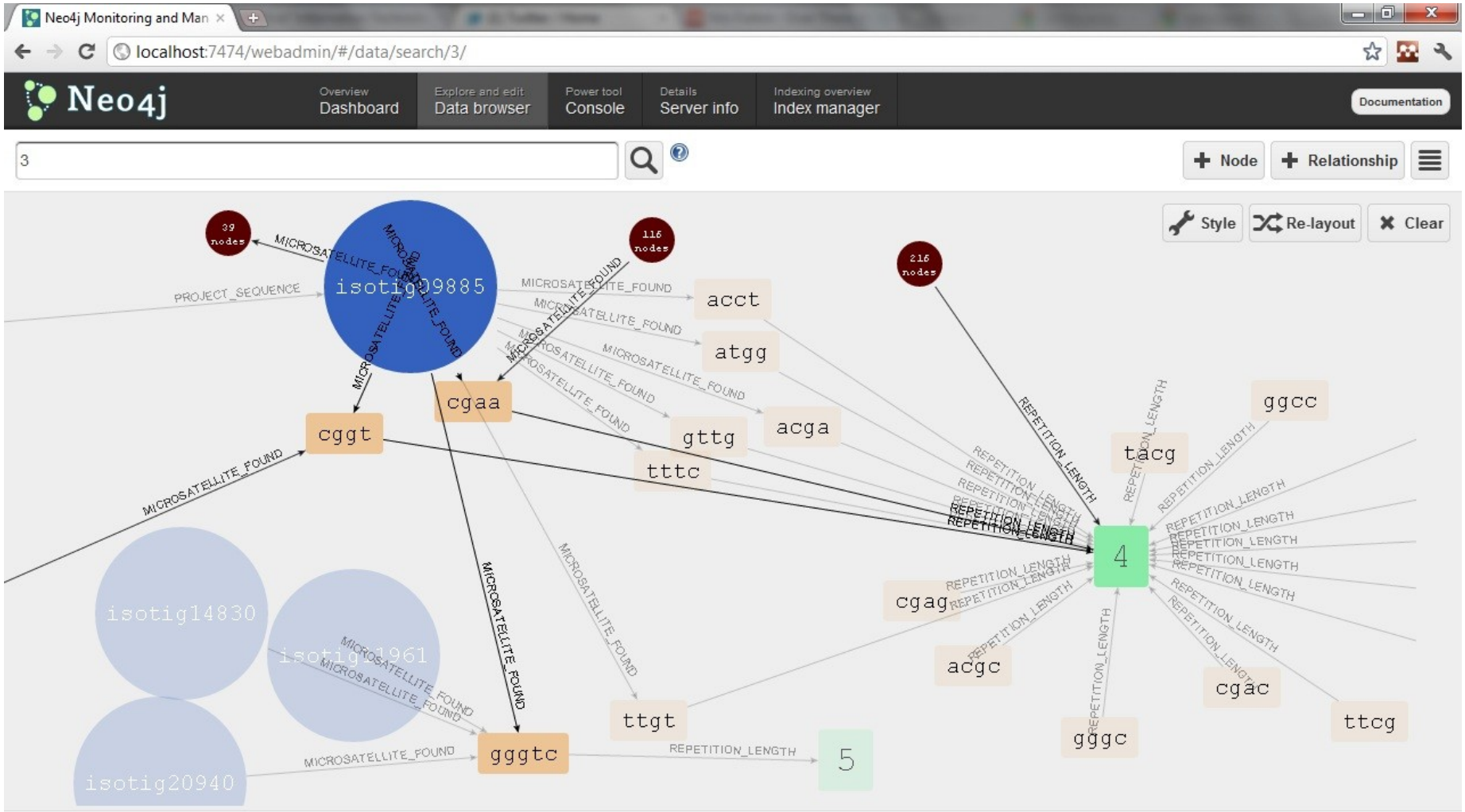


neo technology
value in relationships

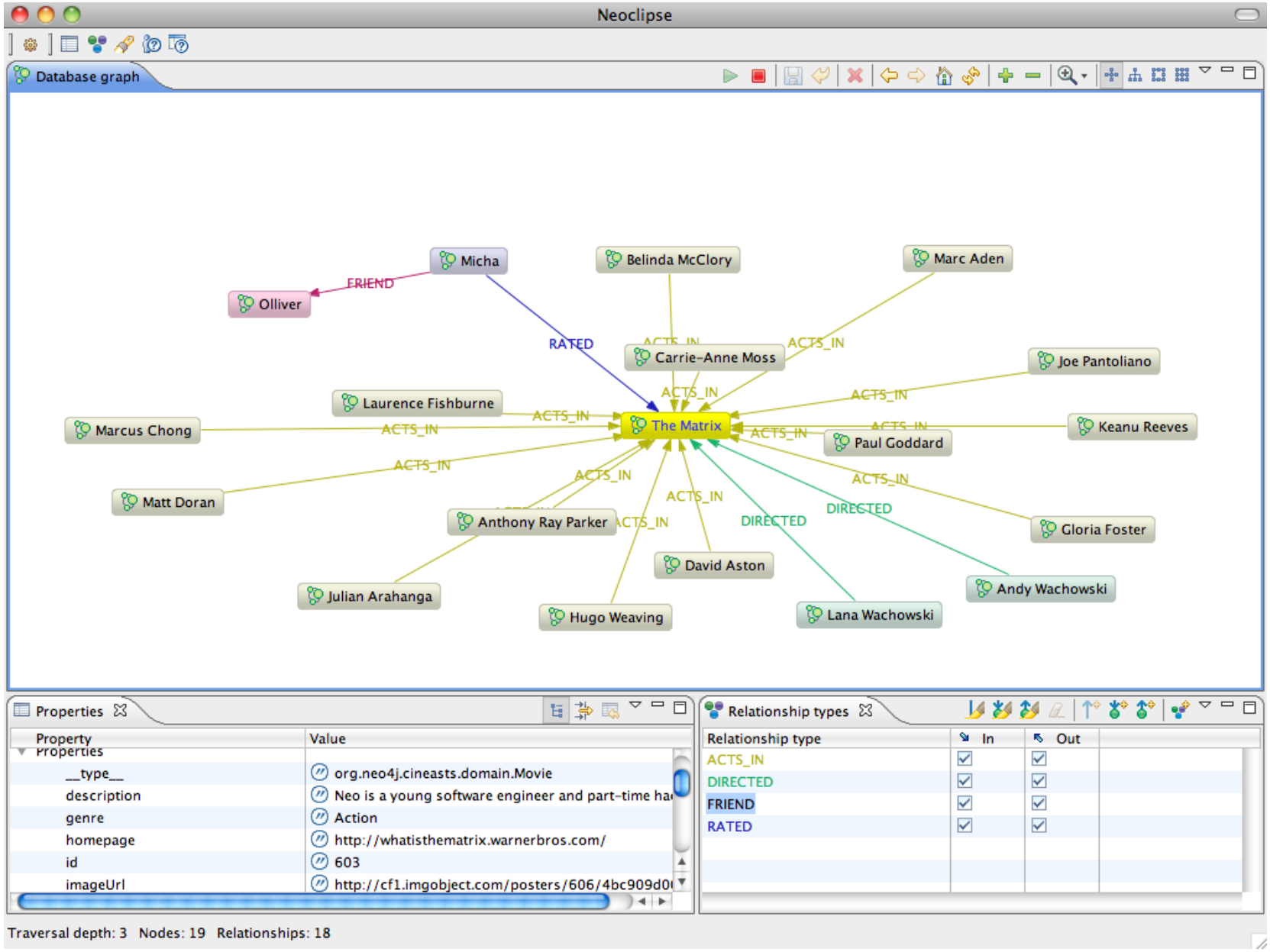
Инструменты



Neo4j Webadmin

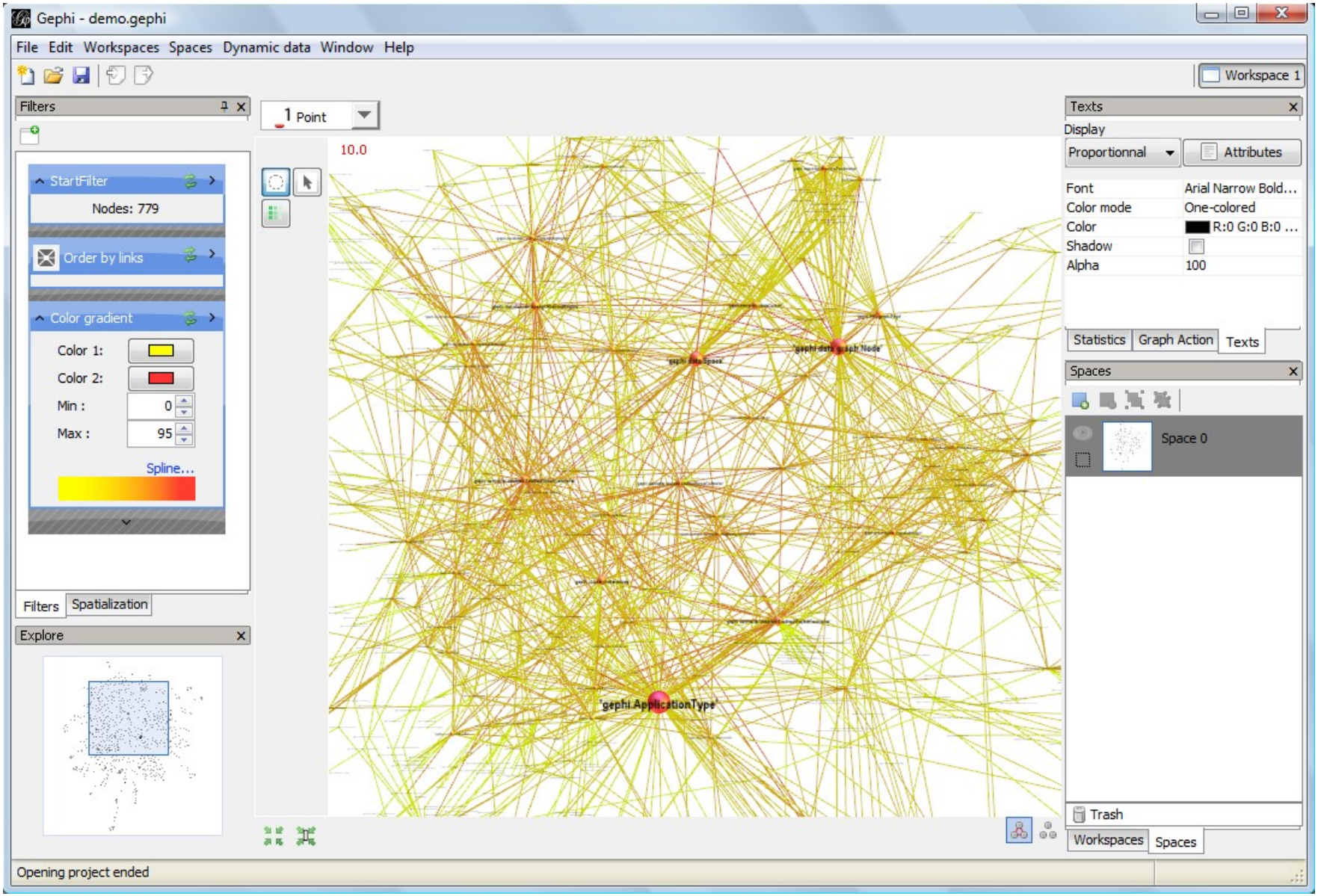


Neoclipse


Neoclipse
Database graph
Micha
Belinda McClory
Marc Aden
Olliver
FRIEND
RATED
Carrie-Anne Moss
ACTS_IN
Joe Pantoliano
Laurence Fishburne
Marcus Chong
The Matrix
Keanu Reeves
Paul Goddard
Matt Doran
Anthony Ray Parker
DIRECTED
Gloria Foster
David Aston
Andy Wachowski
Julian Arahanga
Hugo Weaving
Lana Wachowski
Properties
Property
Value
__type__
org.neo4j.cineasts.domain.Movie
description
Neo is a young software engineer and part-time ha
genre
Action
homepage
http://whatisthematrix.warnerbros.com/
id
603
imageUrl
Relationship types
Relationship type
In
Out
ACTS_IN
DIRECTED
FRIEND
RATED
Traversal depth: 3 Nodes: 19 Relationships: 18

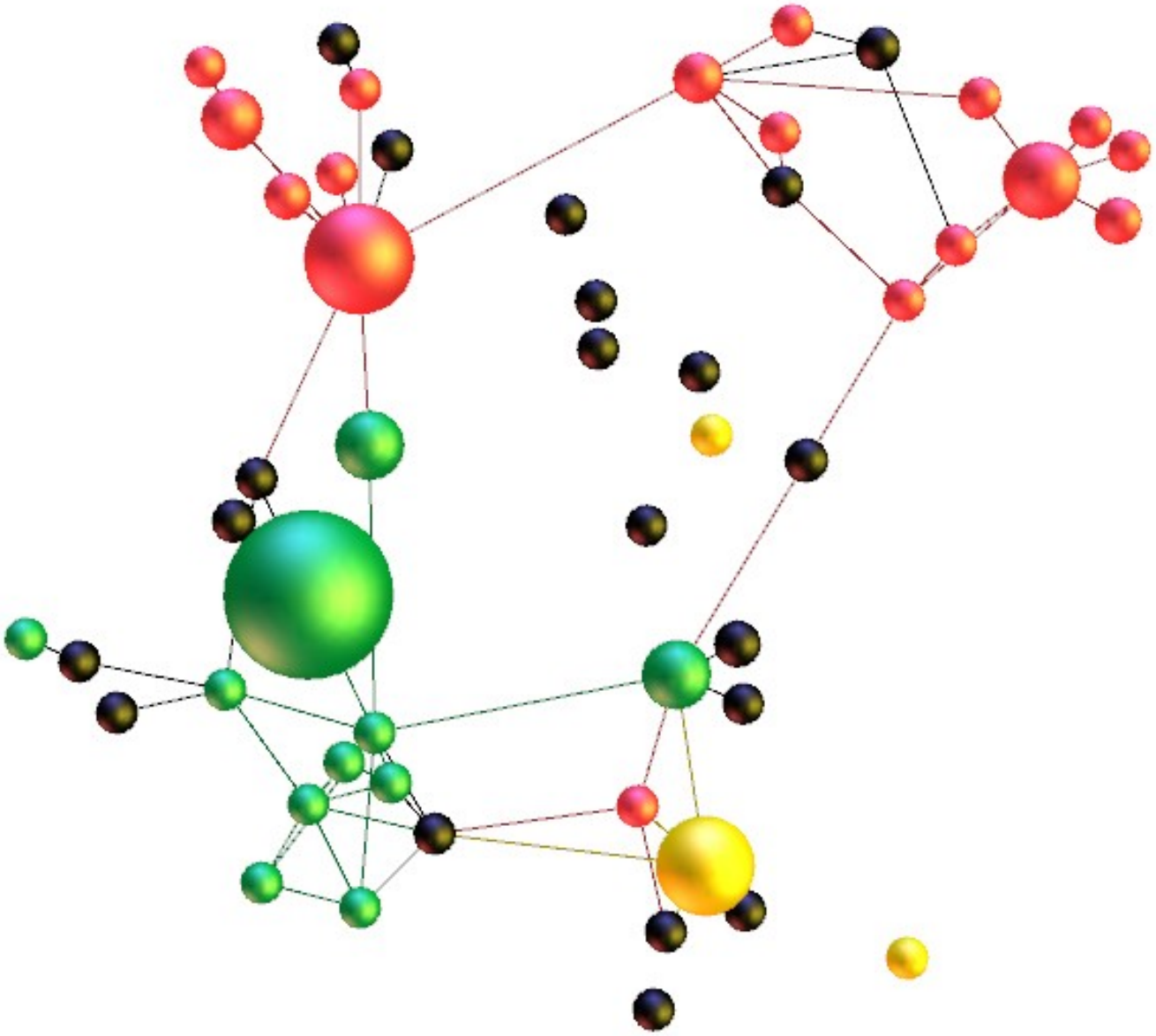
neo technology
value in relationships

Gephi



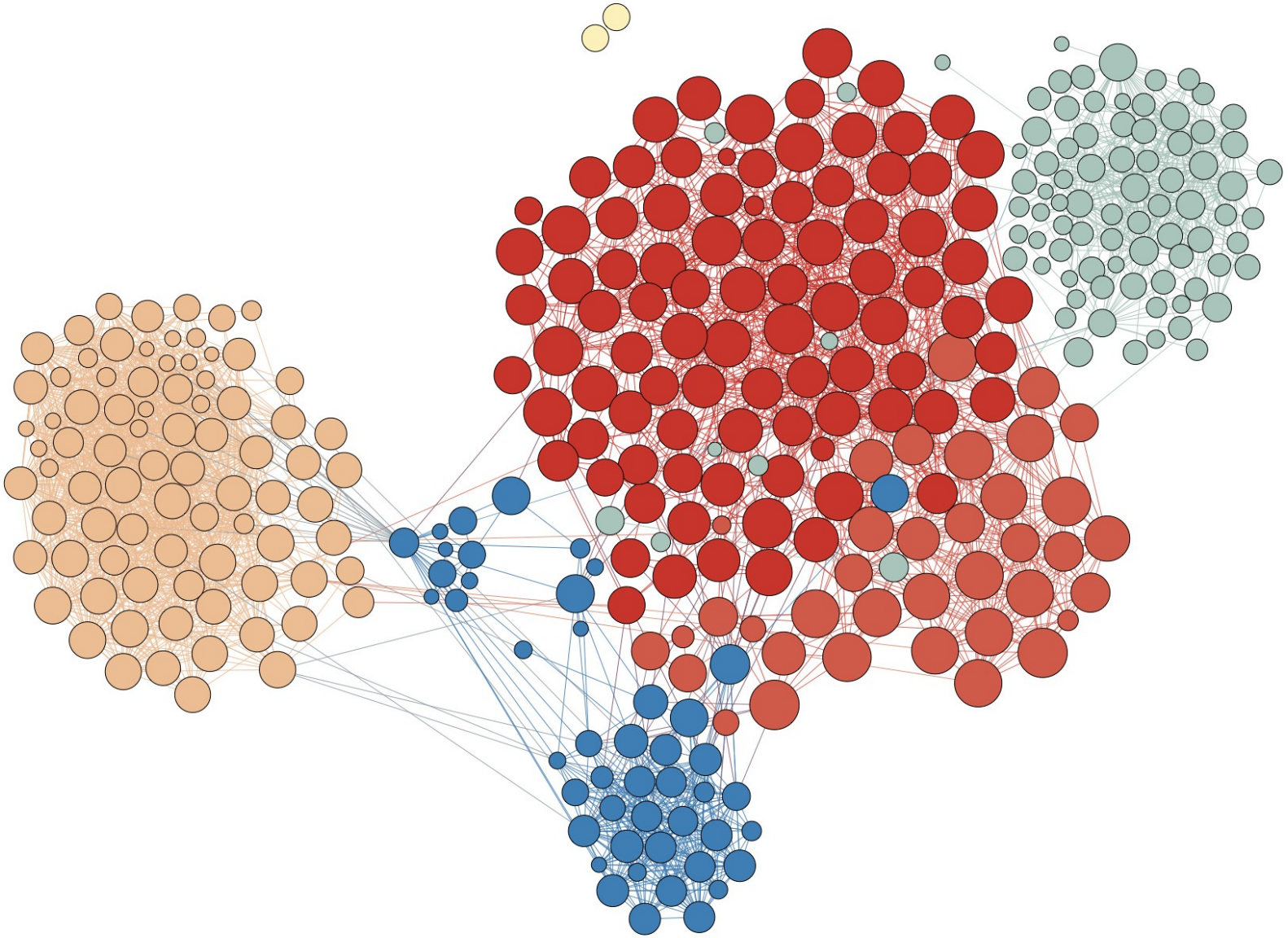
neotechnology
value in relationships

Gephi



neo technology
value in relationships

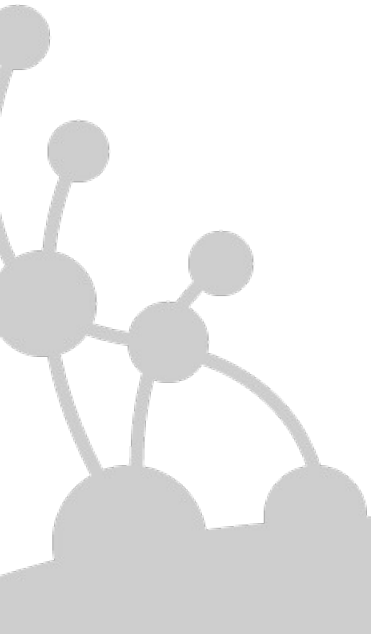
Gephi



neotechnology
value in relationships

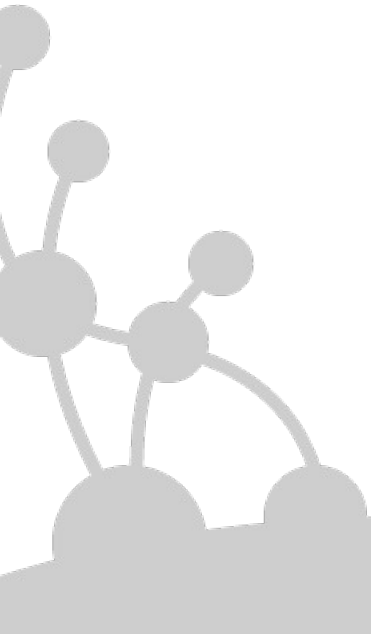
Кластер

CA, AGPL

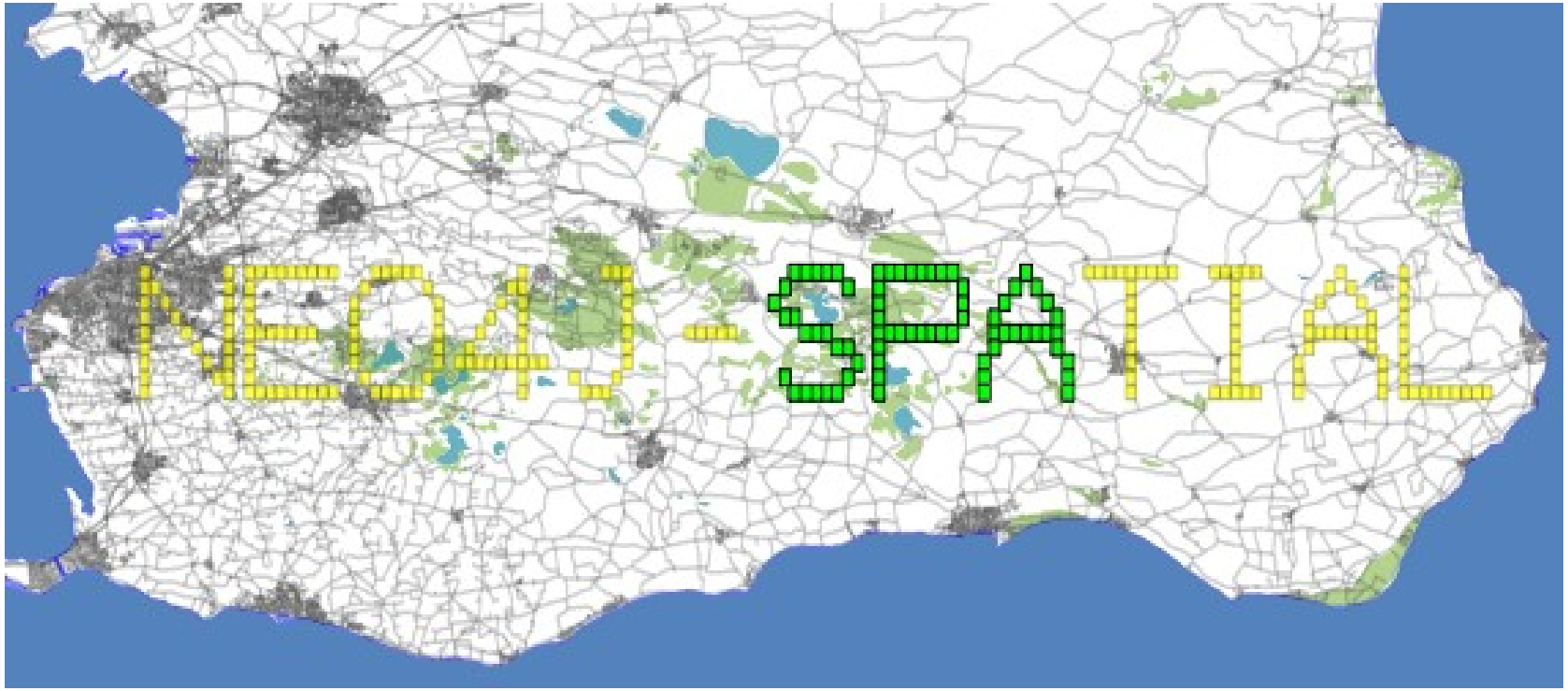


neo technology
value in relationships

Проекты

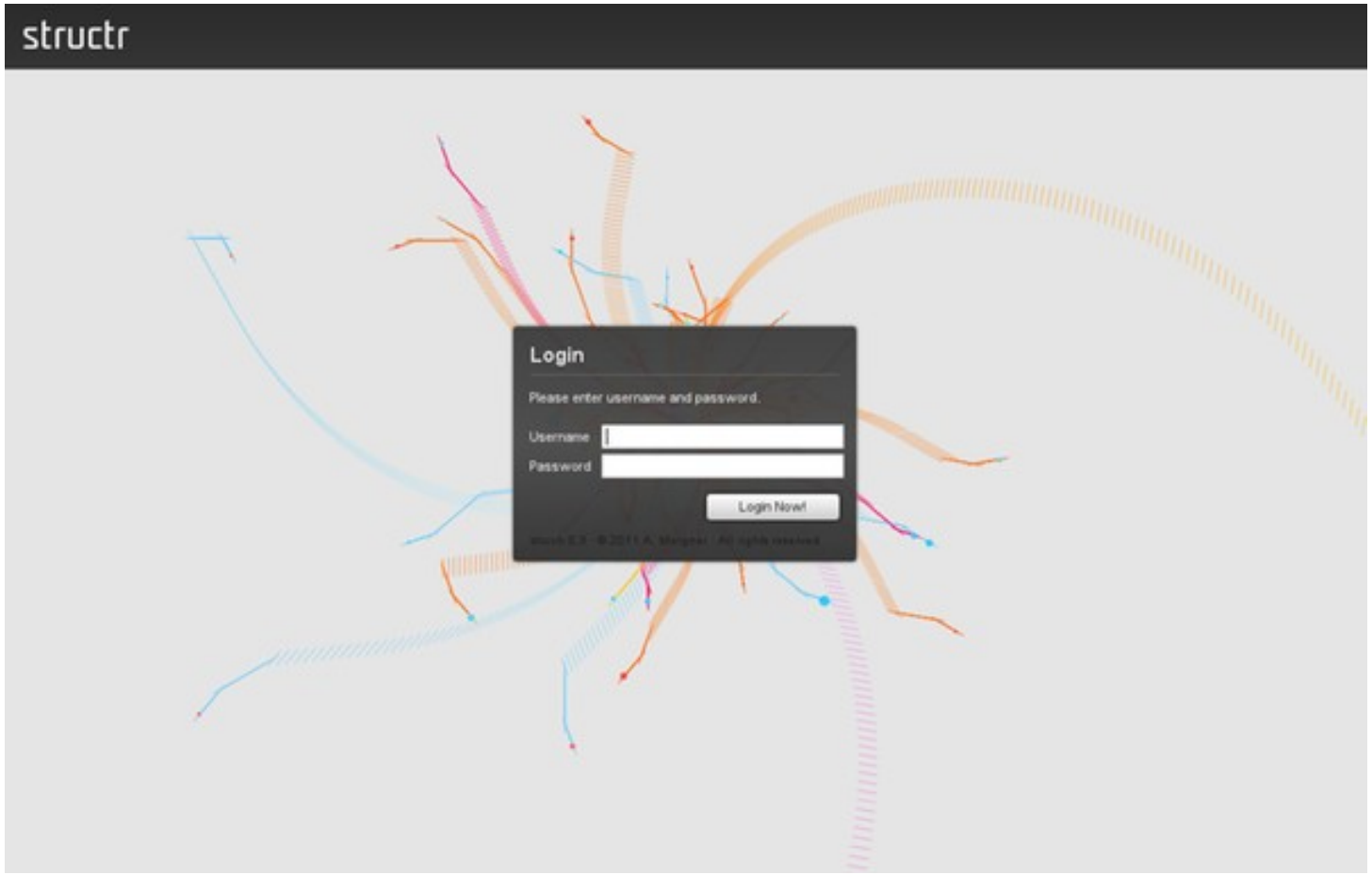


Neo4j Spatial



neotechnology
value in relationships

Structur CMS



neotechnology
value in relationships

Animotron

neotechnology
value in relationships

Neo4j — это

Встраиваемая Java СуБД

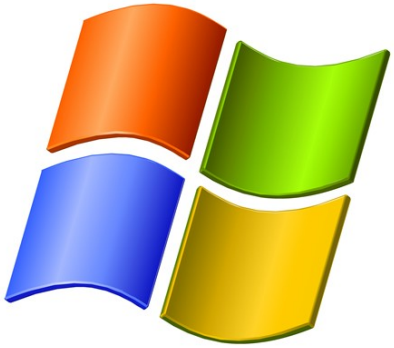
Полностью ACID

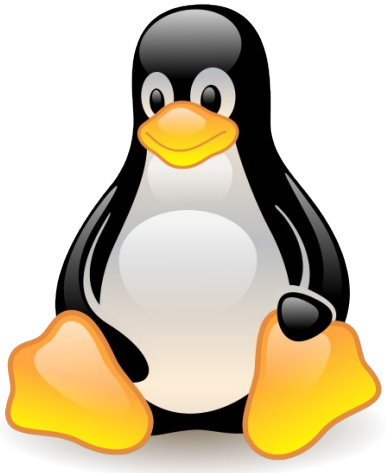
Фреймворк построения индексов

Поддержка 24/7 с 2003 года

Высоко доступная кластеризация (CA)

Большое комьюнити





GPL, AGPL

neo4j.org

neo technology
value in relationships

Спасибо за внимание

Евгений Газдовский

Animotron.org

gazdovsky@gmail.com

twitter.com/gazdovsky

neotechnology
value in relationships

Пожалуйста, поставьте оценку моему докладу.

Ваше мнение очень важно.

Спасибо!