Getting Groovy with Graphs

Who the hell is this guy?

- *Stefan Armbruster*
- *Field Engineer @ Neo Technology*
- *passionate hacker*
- *volunteer firefighter*
- *@darthvader42 | stefan.armbruster@neotechnology.com*

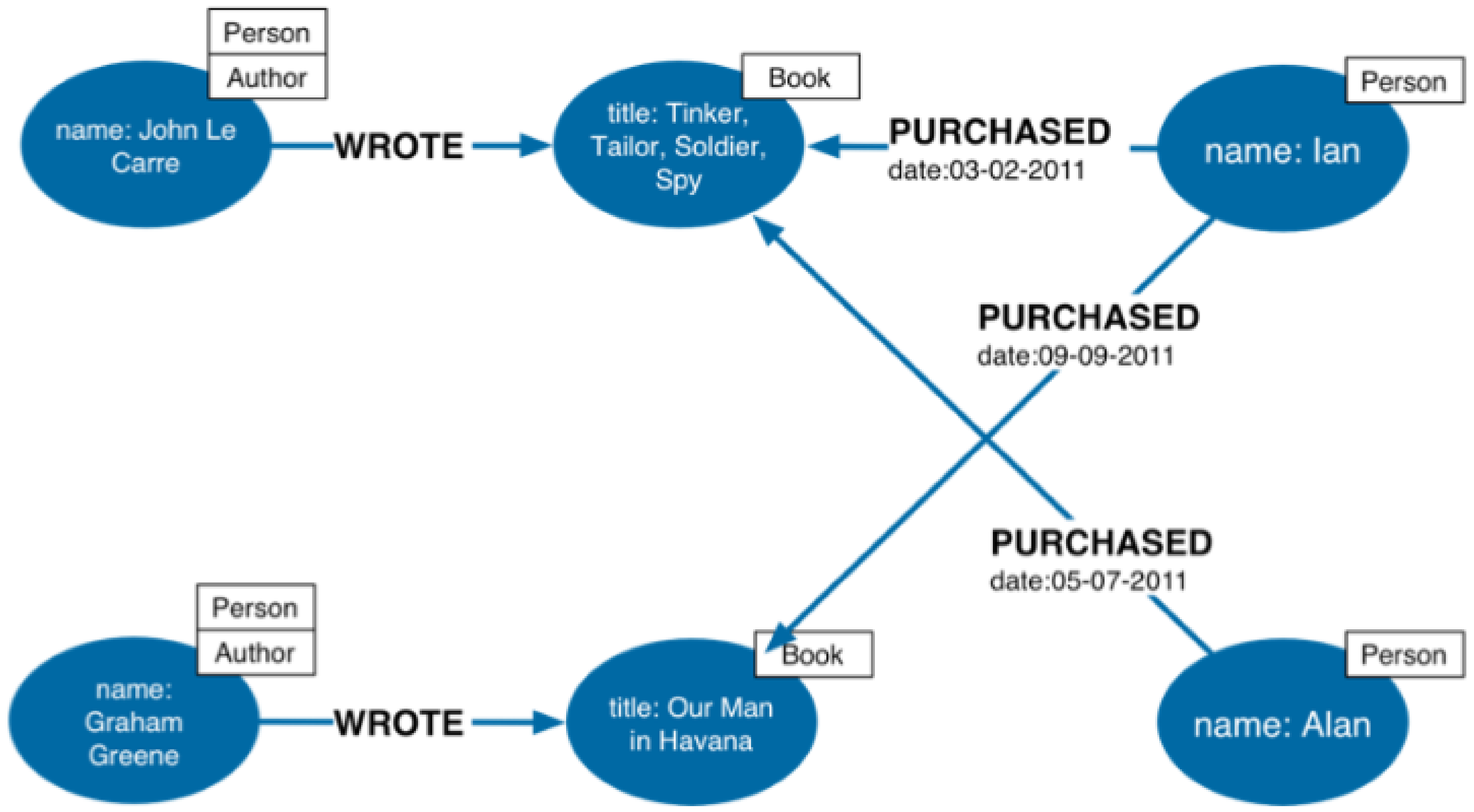
What will he talk about?

- *What is this Neo4j Graphdatabase thing?*
- *Cypher and http-builer*
- *Cypher over JDBC with GSQL*
- *easy upgrades*
- *neo4j-shell's gsh*
- *unmanaged extensions*
- *neo4j grails plugin*
- *neo4j with ratpack*

What is a Graph Database ?

- *labeled Nodes*
- *directed, typed Relationships*
- *arbitrary Properties on each*

Property Graph Model

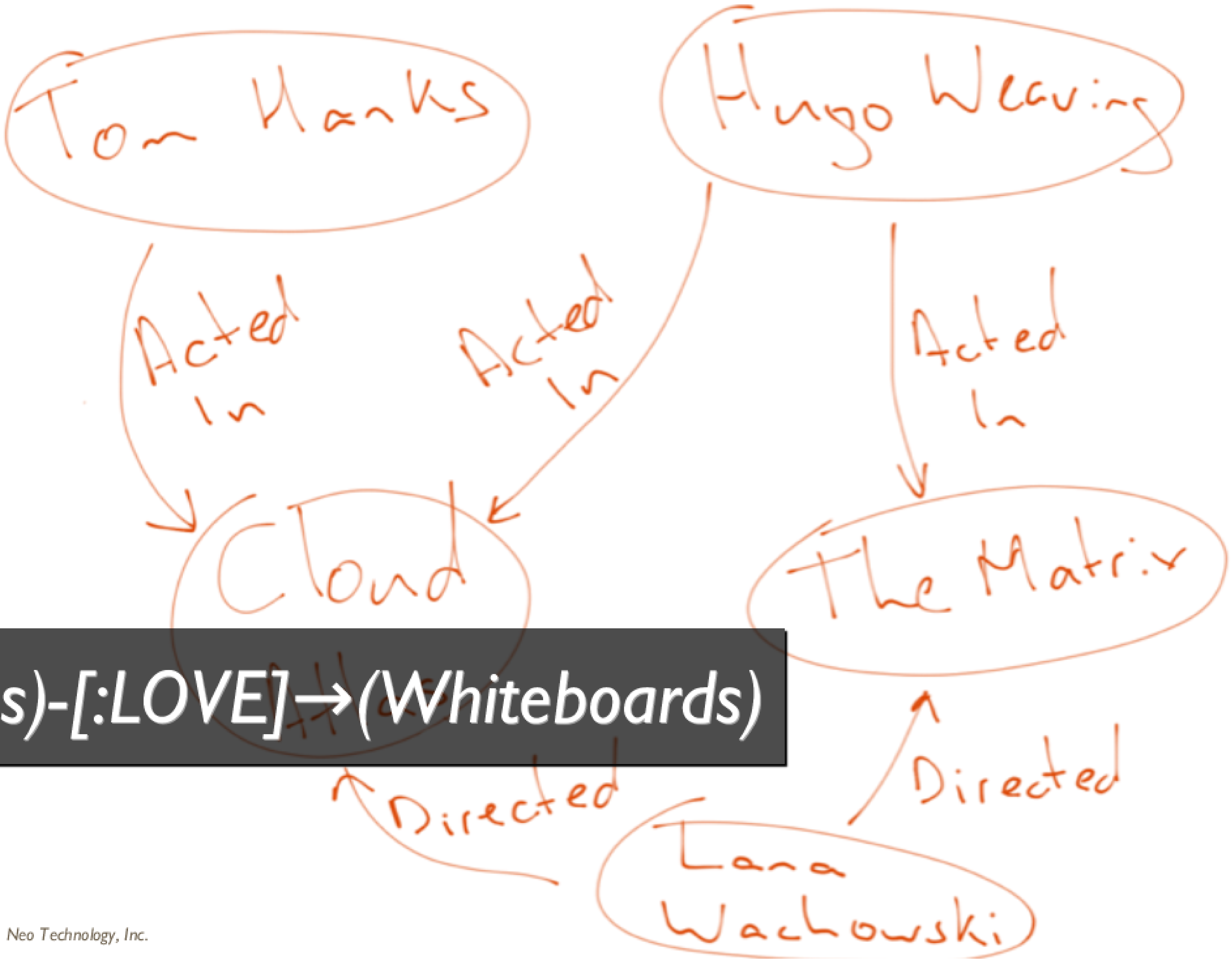


What makes it special ?

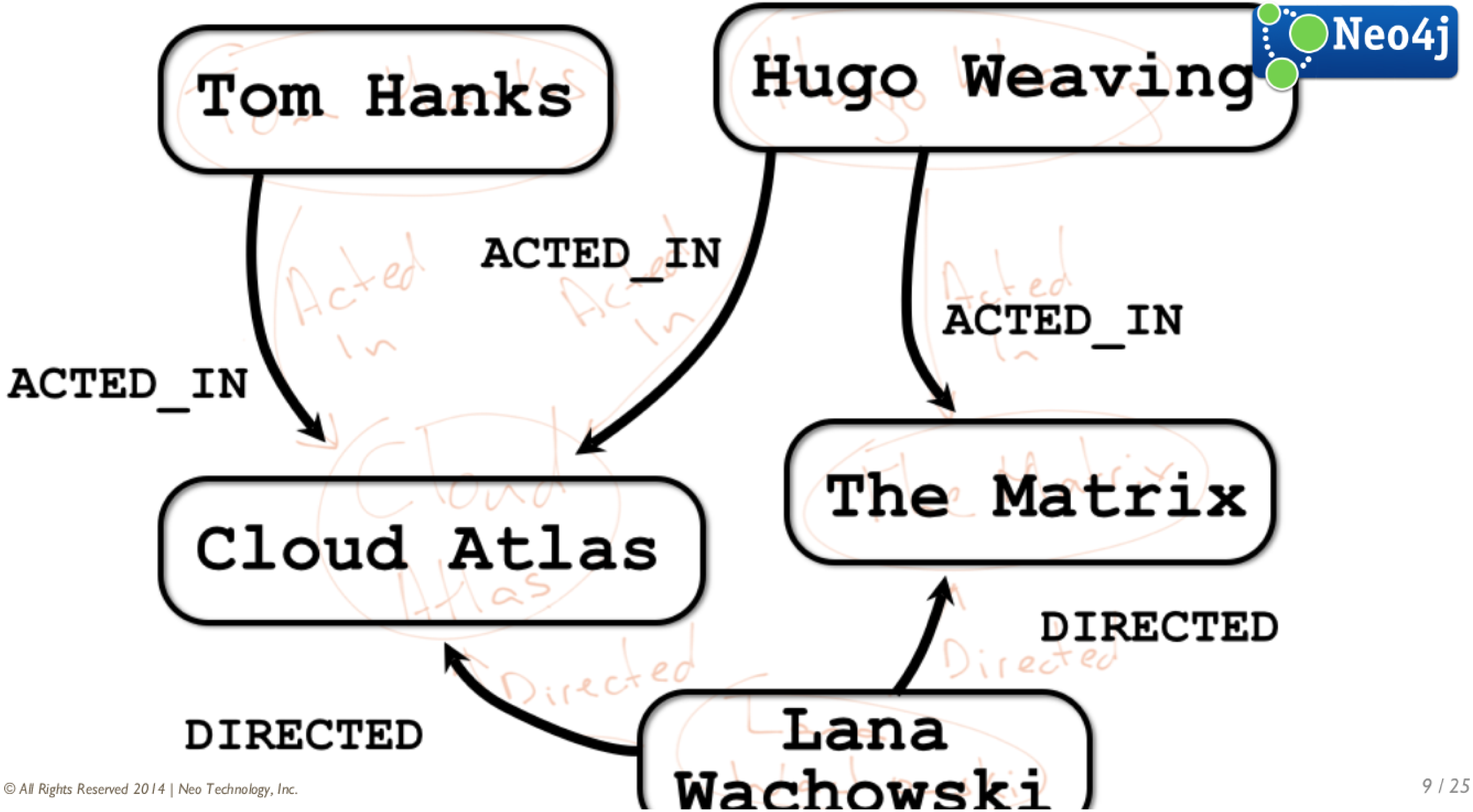
- *close to the object model*
- *prematerialize relationships*
- *traversals in linear time*
- *sparse, heterogenous data + schema free*
- *local queries - explore the neighbourhood*
- *whiteboard-friendly*

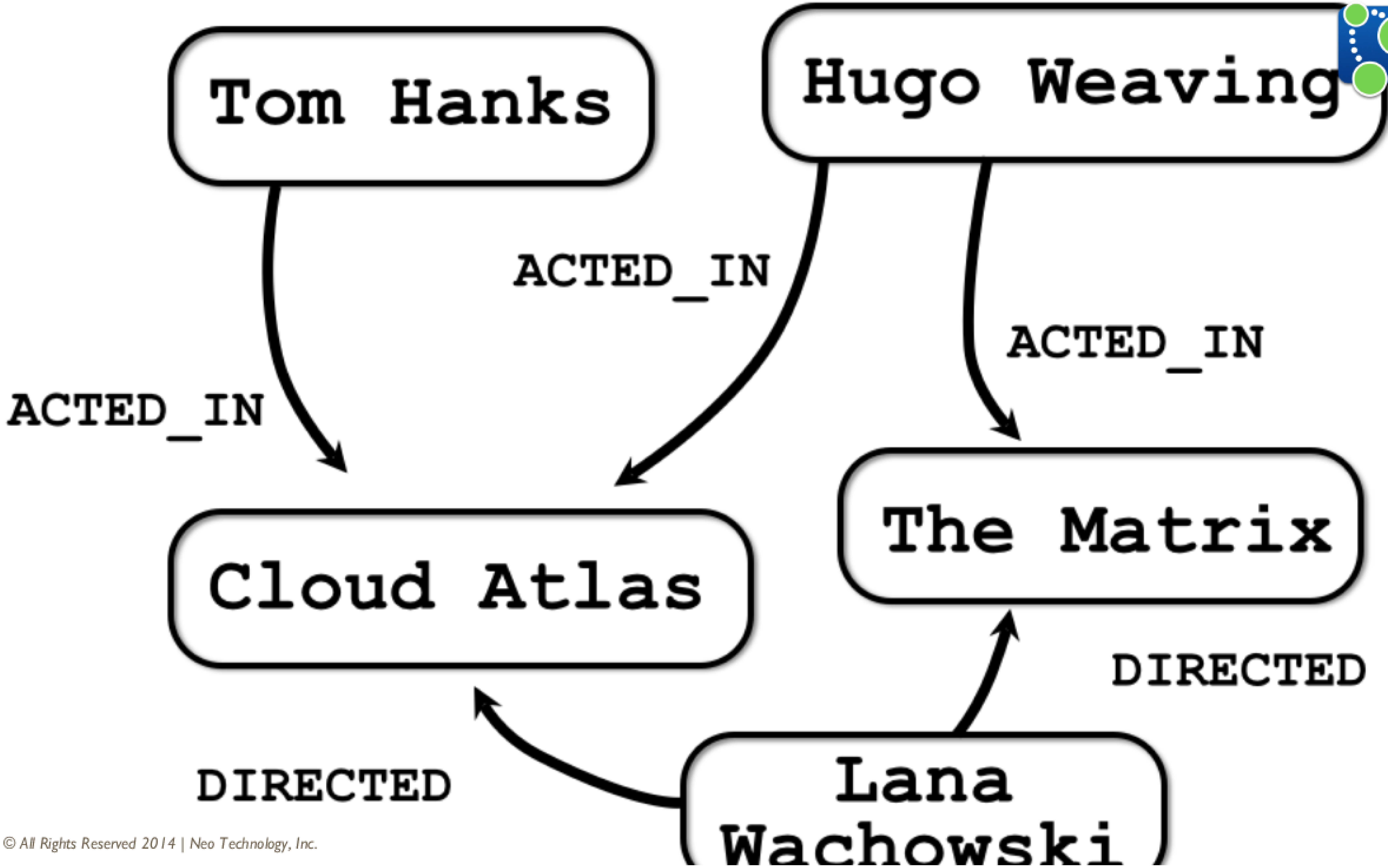
Where should I use it ?

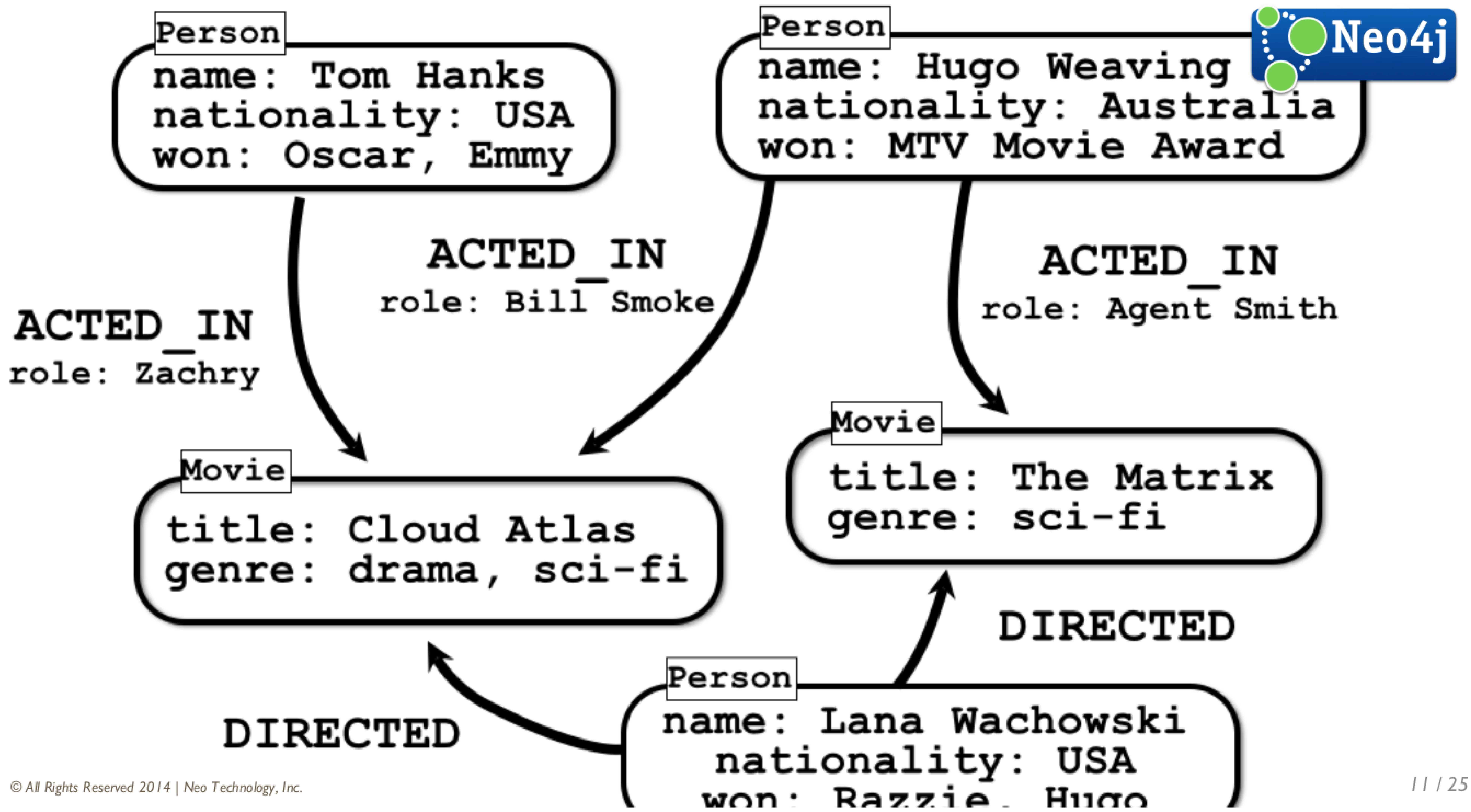
- *Impact Analysis (Network, Software)*
- *Routing / Logistics*
- *Recommendation, Dating, Job-Search*
- *Sciene (Metadata, Drug Research)*
- *Masterdata, Hierarchy-Mgmt*
- *Fraud-Detection, Market-Analysis*
- *Social, …. and many more*



(Graphs)-[:LOVE]→(Whiteboards)


Tom Hanks
Hugo Weaving
ACTED_IN
ACTED_IN
ACTED_IN
Acted In
Acted In
Acted In
Cloud Atlas
The Matrix
DIRECTED
DIRECTED
Directed
Directed
Lana Wachowski


Tom Hanks
Hugo Weaving
ACTED_IN
ACTED_IN
ACTED_IN
Cloud Atlas
The Matrix
DIRECTED
DIRECTED
Lana Wachowski


Person
name: Tom Hanks
nationality: USA
won: Oscar, Emmy
Person
name: Hugo Weaving
nationality: Australia
won: MTV Movie Award
Neo4j
ACTED_IN
role: Zachry
ACTED_IN
role: Bill Smoke
ACTED_IN
role: Agent Smith
Movie
title: Cloud Atlas
genre: drama, sci-fi
Movie
title: The Matrix
genre: sci-fi
DIRECTED
DIRECTED
Person
name: Lana Wachowski
nationality: USA
won: Razzie, Hugo

http-builder with cypher

```
@Grab(group='org.codehaus.groovy.modules.http-builder', module='http-builder', version='0
@Grab(group='net.sf.json-lib', module='json-lib', version='2.2.3', classifier='jdk15')
import groovyx.net.http.RESTClient
import static groovyx.net.http.ContentType.*

def http = new RESTClient( 'http://localhost:7474/db/data/transaction/commit', JSON)
//http.setProxy('localhost', 8888, 'http')

def response = http.post(body: [statements: [[statement: 'MATCH (n) RETURN n, count(n) LI

assert response.status == 200
println "Location Header: $response.headers.Location"

def data = response.data.results.data[0]

println "result columns: ${response.data.results.columns[0]}"
println "we have ${data.size()} result rows"

data.eachWithIndex { row, index ->
  println "result ${index}: $row.row"
}
```

GSQL with Neo4j JDBC

- *Cypher can be used over JDBC*
- *https://github.com/neo4j-contrib/neo4j-jdbc*
- *Why not using GSQL ?*
- *use backticks for labels and reltypes*
- *no named params*

GSQL with Neo4j JDBC

```
@GrabResolver(name="neo4j", root="http://m2.neo4j.org/content/repositories/releases/")
//@GrabResolver(name="restlet", root="http://maven.restlet.org/")
@GrabConfig(systemClassLoader = true)
@Grab('org.neo4j:neo4j-jdbc:2.0.2')
@Grab('org.neo4j:neo4j-kernel:2.1.2')

import groovy.sql.*

def sql = Sql.newInstance('jdbc:neo4j://localhost:7474/')

println "parameterized cypher statement"
sql.eachRow("match (m:`Movie` {title:{1}}) return m" , ['The Matrix']) {
  println "row $it"
}
```

easy upgrades

- *to upgrade Neo4j datastore*
- *shut down cleanly*
- *use next minor release*
- *set* `allow_store_upgrade=true`
- *start up*
- *Groovy to the rescue!*
- *https://gist.github.com/sarmbruster/3011606*

neo4j-shell's gsh command

- *drop* `groovy-all-<version>.jar` *to* `/lib` *folder*
- *drop a groovy script into neo4j's main directory*
- *use* `gsh --<scriptname> <args*>`
- *limited use: no access to graphdb directly (for now)*

using spock with graphs

- *testing in graph world is even more important*
- *provides 2* `ExternalResource` *for Neo4j:*
    - *Neo4jResource: white box*
    - *Neo4jServerResource: black box*
- *https://github.com/sarmbruster/neo4j-spock-extension*

neo4j-lazybones

- *bootstrap neo4j unmanaged extensions*
- *gradle build*
- *simplistic unmanaged ext + tests*
- *http://dl.bintray.com/sarmbruster/lazybones-templates/unmanaged-extension-template-0.1.zip*

can I haz code?

Demo

neo4j grails plugin

- *as of today: 2.0.0-M02*
- *passing GORM TCK*
- *internally Cypher over JDBC is used*

neo4j grails plugin

Demo

neo4j and ratpack

- *experiment to use ratpack as alternative server*
- *better throuput than classic Neo4j server*

neo4j and ratpack

Demo

Questions ?
Thank You!