

Live Graph Model Documentation with Graph Gists

Who the hell is this guy?

- Michael Hunger
- Developer Advocate Neo Technology
- Love People and Graphs
- @mesirii | michael@neotechnology.com

What will he talk about?

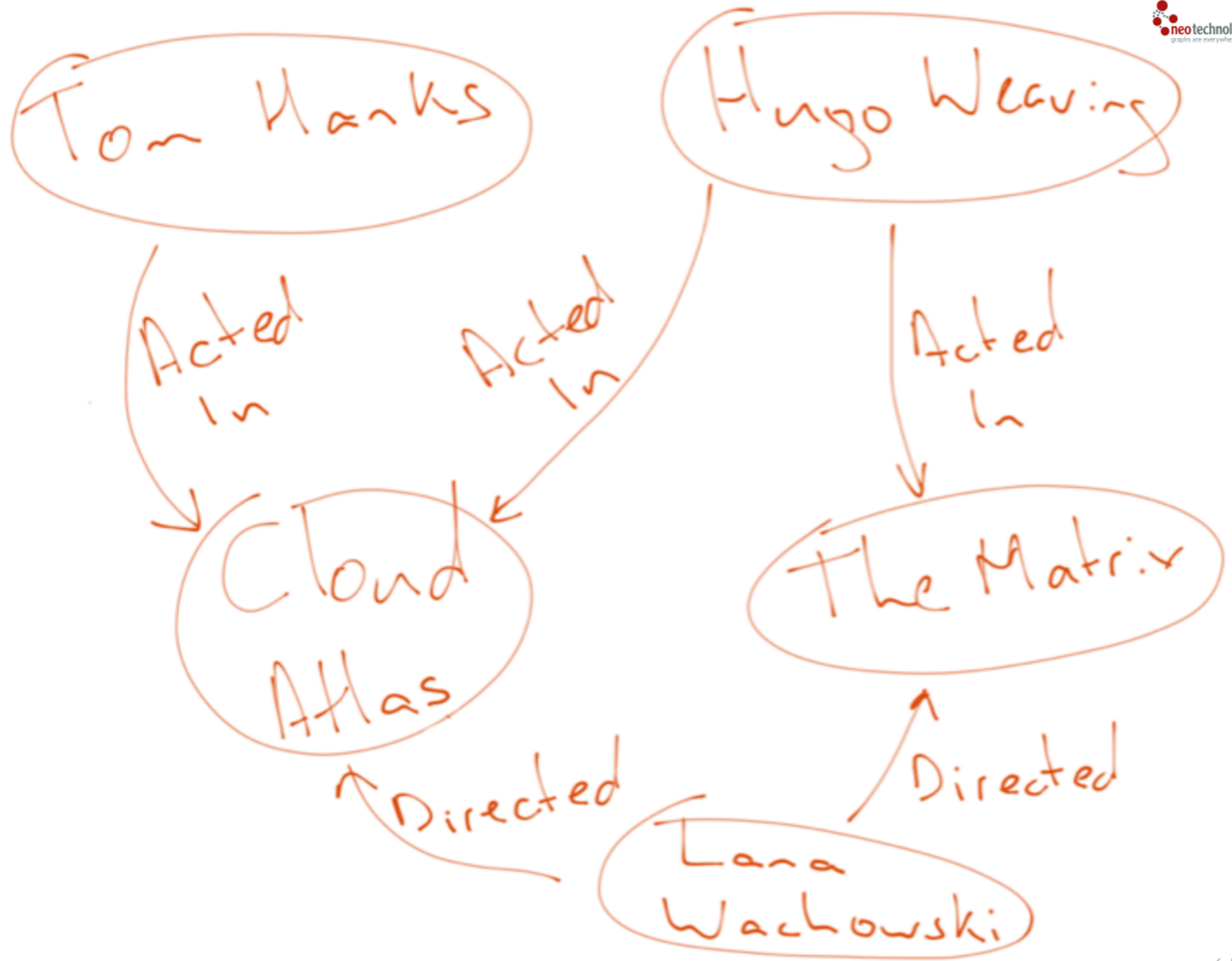
- (Graphs)-[:LOVE]→(Whiteboards)
- (Graphs)-[:HATE]→(Text-Files)
- Ascii-Art Rocks
- Really? The GraphGist idea
- Power Combination of Cool Tools
- How does it work?
- Our Community Rocks - The GraphGist Challenge

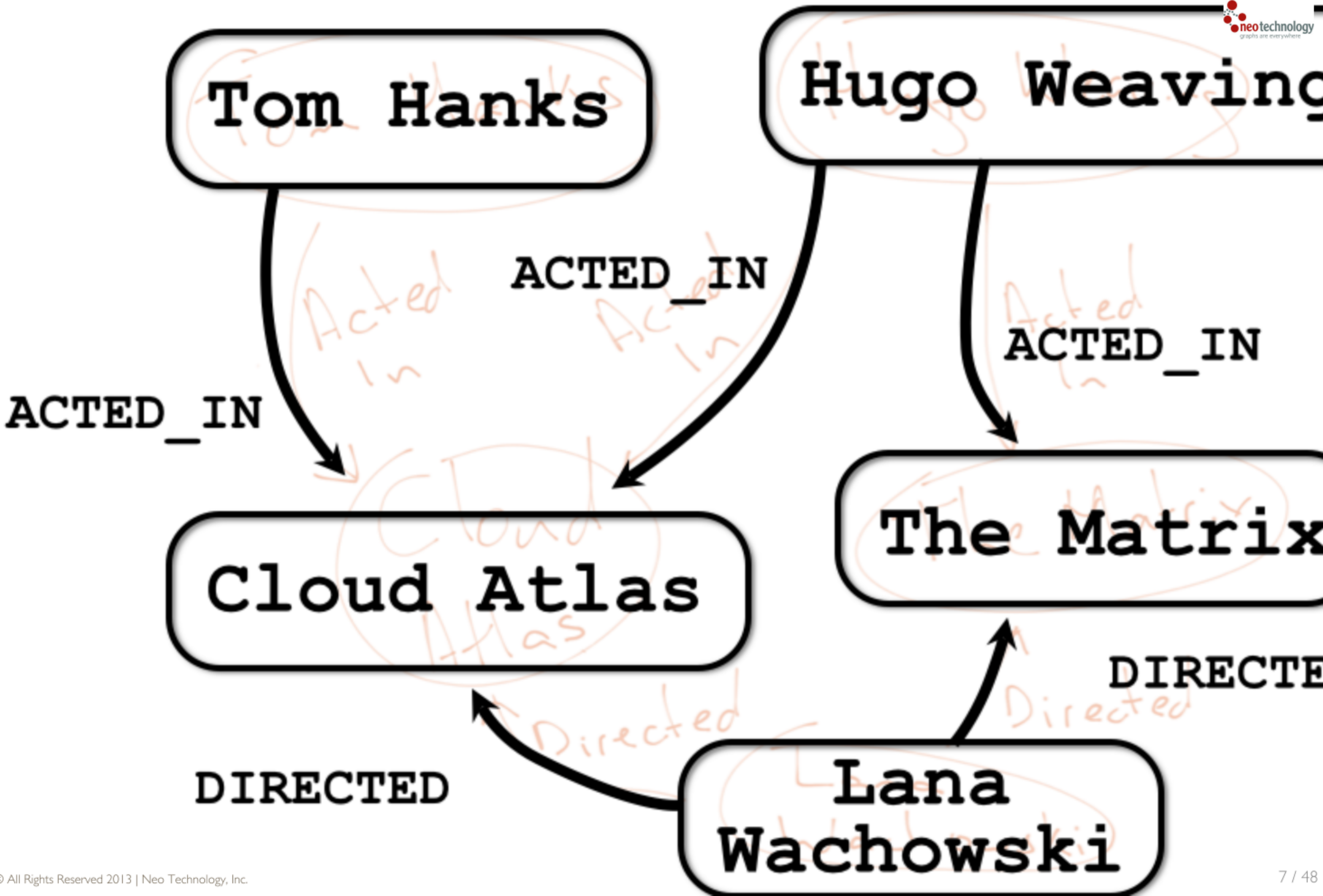
Question for you!

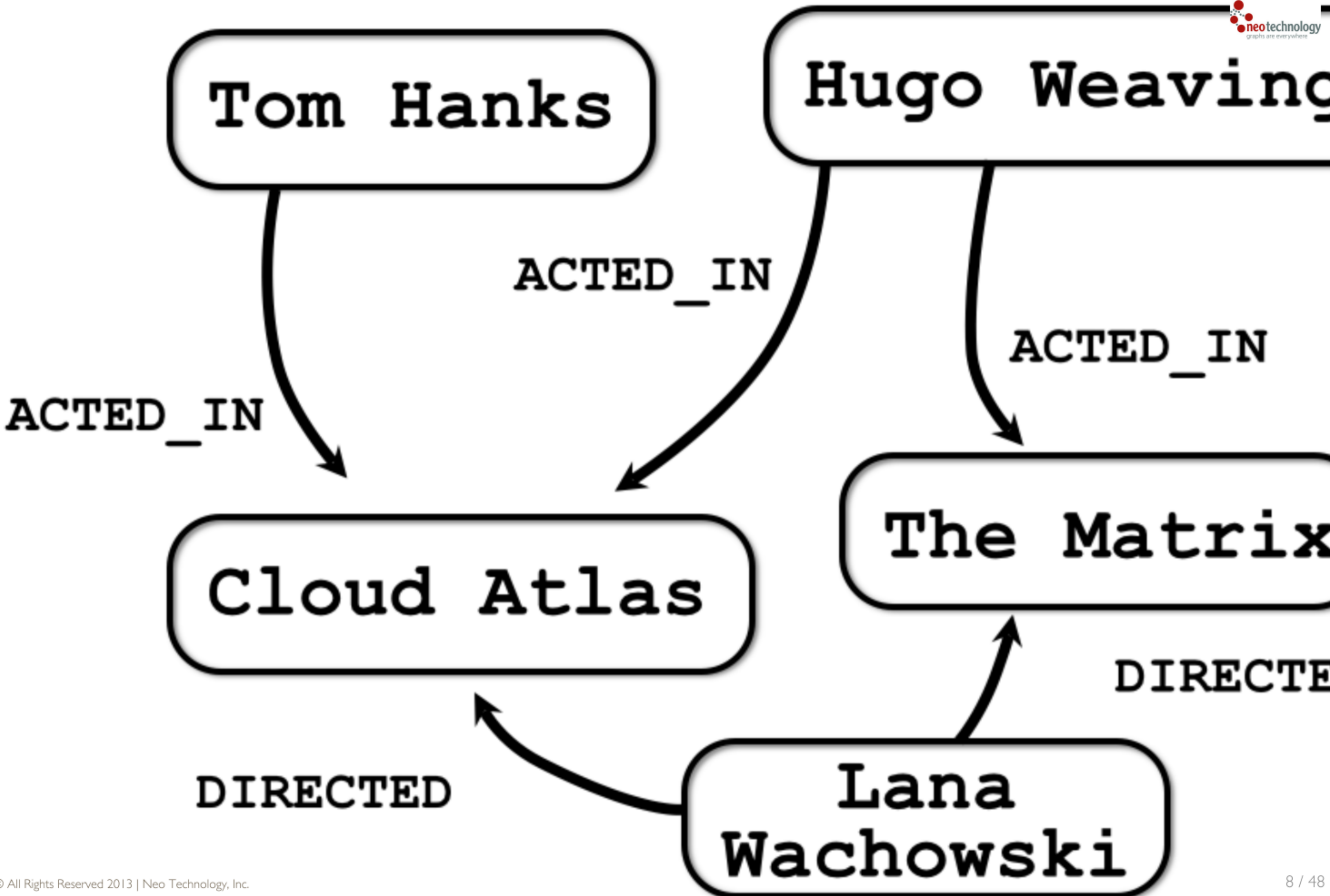
- How to discuss/communicate your awesome graph domain model?
- What is an easy way to present it in a modern, interactive format?
- Do you draw the model by hand?
- Do you need a running database?
- Or just a browser?

(Graphs)-[:LOVE]→(Whiteboards)

- The picture vs. 1000 words
- creative thinking doodles
- modeling is incremental
- ongoing discussion, refinement
- whiteboard, -walls, -tables







Person

name: Tom Hanks
nationality: USA
won: Oscar, Emmy

Person

name: Hugo Weaving
nationality: Australia
won: MTV Movie Award

ACTED_IN
role: Zachry

ACTED_IN
role: Bill Smoke

ACTED_IN
role: Agent S

Movie

title: Cloud Atlas
genre: drama, sci-fi

Movie

title: The Matrix
genre: sci-fi

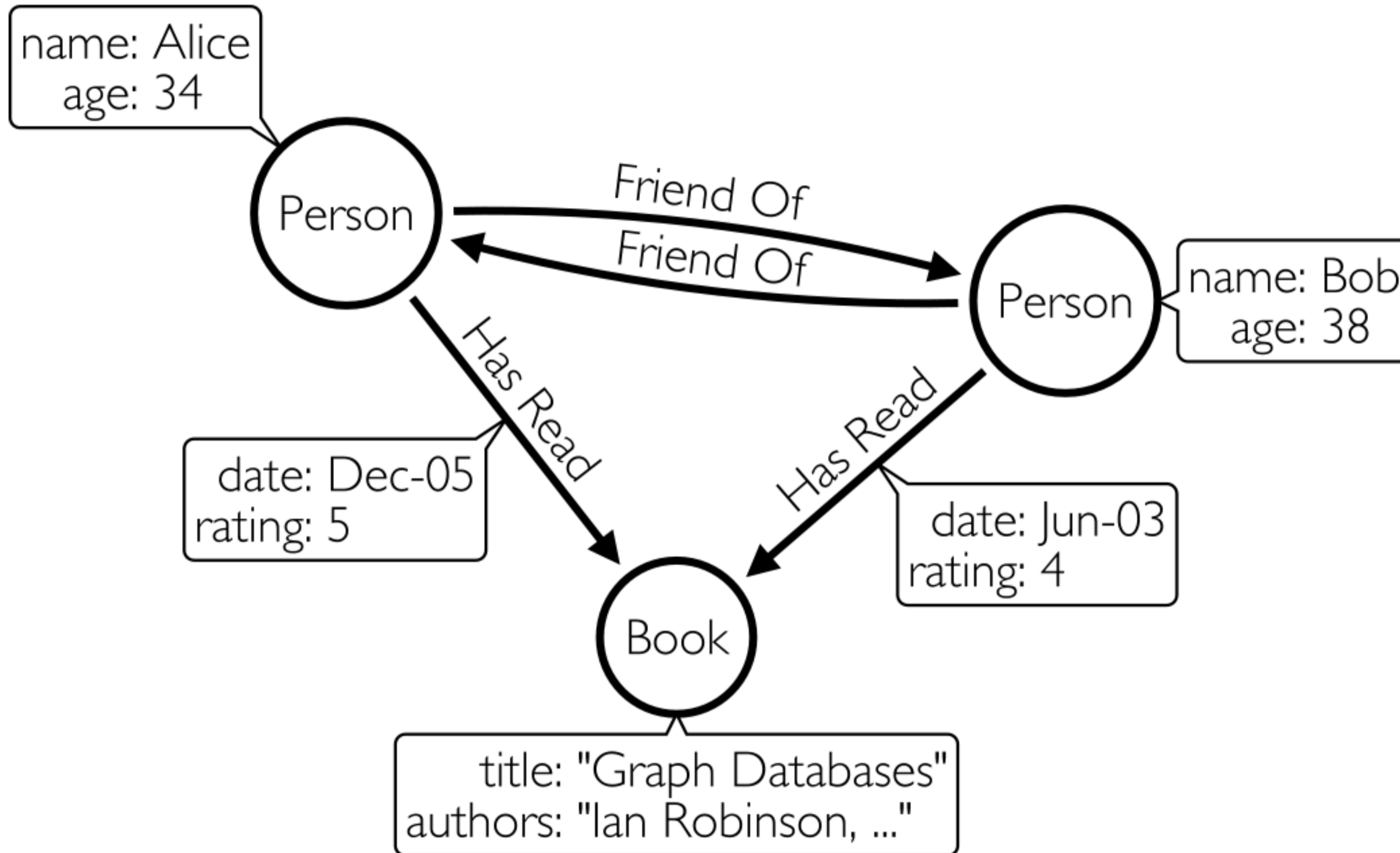
DIRECTED

Person

name: Lana Wachowski
nationality: USA
won: Razzie, Hugo

DIRECTED

Graph Model



(Graphs)-[:HATE]→(Text-Files)

- It's the 1000 words
- Single dimensional
- No graphics
- Verbose

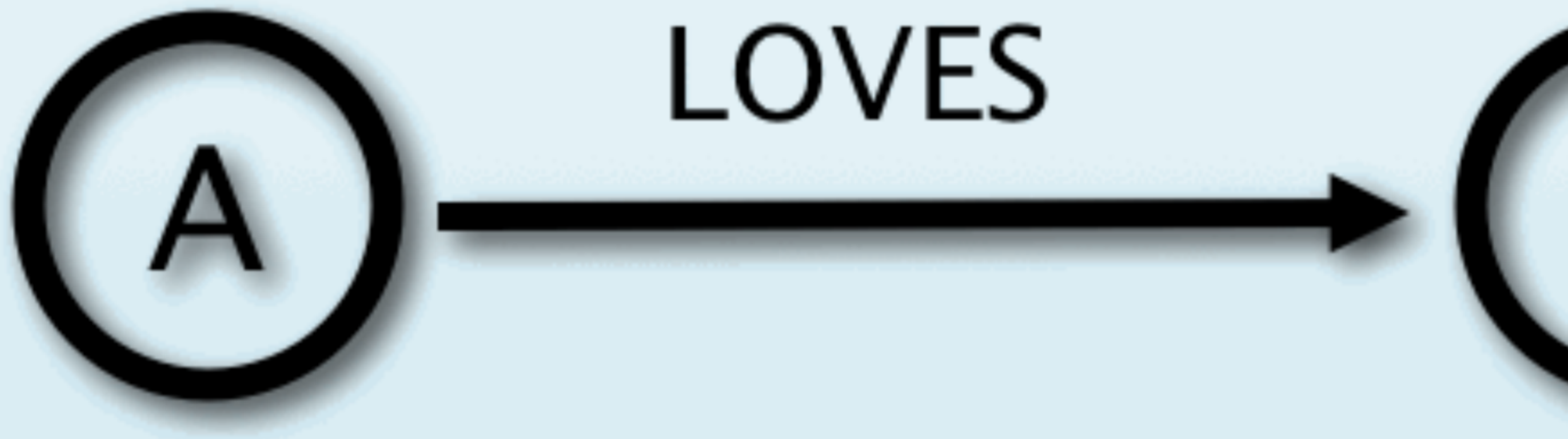
But Ascii-Art Rocks

- Turn text into pictures
- The Power of Symbols
- Graph Patterns Made easy
- Hacker and Mudder Friendly
- Diffs, VCS

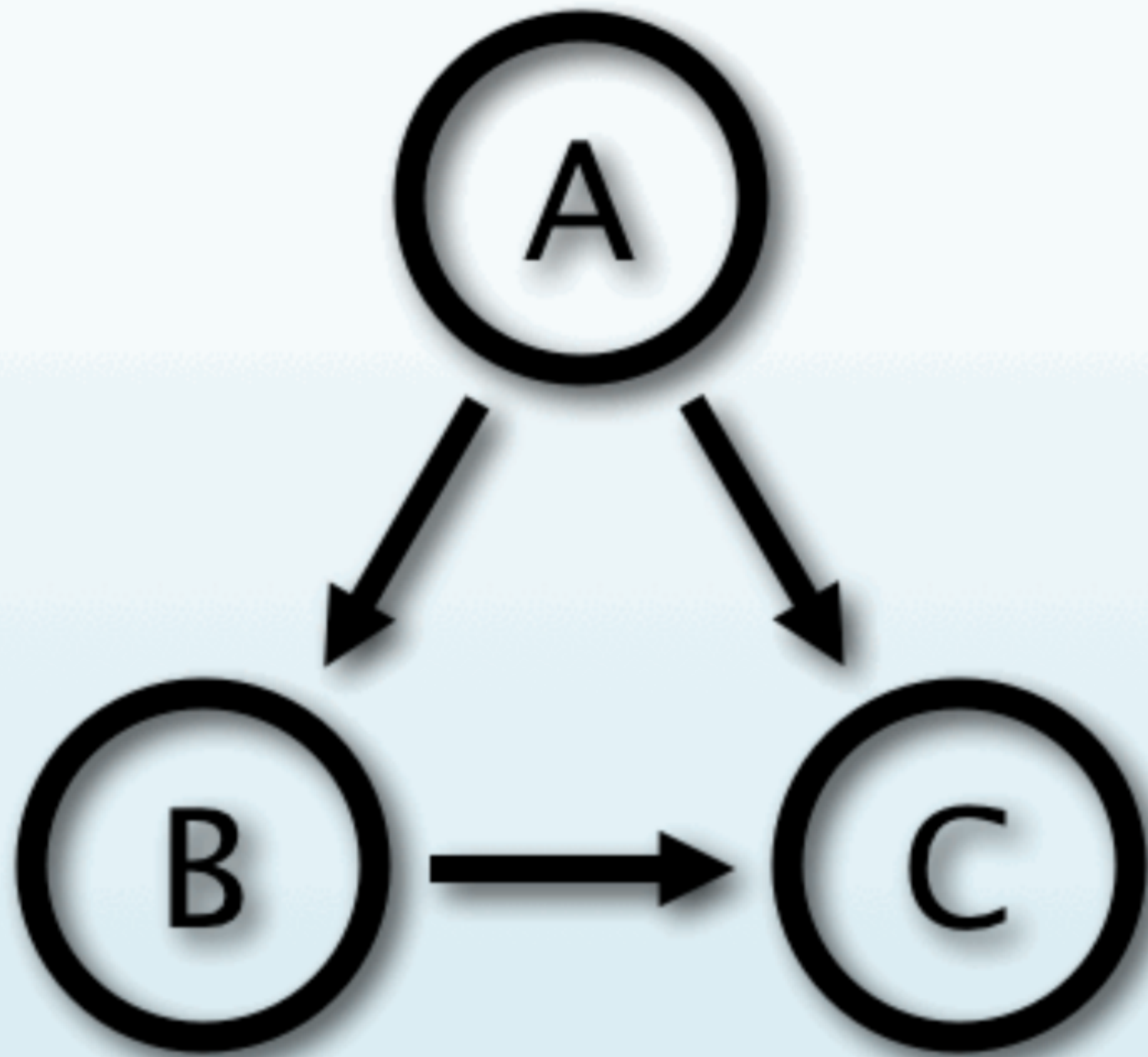
Cypher

(Cypher)-[:USES]→(Ascii-Art)

- Declarative Graph Query Language
- Graph Pattern Matching
- Humane, Readable
- Expressive
- Read and Write Graphs
- Tabular Results



A - [:LOVES] ->



A --> B --> C, A -->

A --> B --> C <--

Cypher Query

Setup

```

1 CREATE (neo:Company {name:"Neo"})-[:IN]->(:Country {tld:"SE"})
2 CREATE (:Person {name:"Peter",age:40})-[:WORKS_AT]->(neo)
3 CREATE (:Person {name:"Kenny",age:23})-[:WORKS_AT]->(neo)
4 CREATE (:Person {name:"Kenny",age:23})-[:WORKS_AT]->(neo)
5 CREATE (structr:Company {name:"Structr"})-[:IN]->(:Country {tld:"DE"})
6 CREATE (:Person {name:"Axel",age:39})-[:WORKS_AT]->(structr)
7 CREATE (:Person {name:"Christian",age:25})-[:WORKS_AT]->(structr)
  
```

Query

```

1 // Count Hipsters by Country
2 MATCH (p:Person)-[:WORKS_AT]->(:Company)-[:IN]->(c:Country)
3 WHERE p.age < 28
4 RETURN c.tld, count(p) as hipsters
  
```

How do you demo it?

- We built our own sandbox

The Neo4j Console

The screenshot shows the Neo4j Console interface. At the top, there are buttons for 'Clear DB', 'Help', 'Share', 'Toggle Viz', and 'Options'. Below these is a Cypher query:

```
MATCH (p:Person)-[:WORKS_AT]->(c:Company)-[:IN]->(c:Country)
WHERE p.age < 28
RETURN c.tld, count(p) AS hipsters
```

To the right of the query is a graph visualization showing nodes for 'Company Neo', 'Person Kenny', and 'Person Peter' connected by 'WORKS_AT' and 'IN' relationships.

Below the query is a table with the following data:

c.tld	hipsters
SE	2
DE	1

Below the table, it says 'Query took 52 ms and returned 2 rows.' with a 'Result Details' button.

In the center, there is instructional text: 'You can modify and query this graph by entering statements in the input field at the bottom. For some syntax help hit the Help button. If you want to share your graph, just do it with Share'. There are 'Help' and 'Share' buttons.

Below the text is another graph visualization showing nodes for 'Person Christian', 'Person Axel', 'Company Structr', and 'Country DE' connected by 'WORKS_AT' and 'IN' relationships.

At the bottom, there is an input field with the same Cypher query as above and a 'Run' button.

Console - Sandbox, a useful Tool

How does it work?

- Tiny Webapp on Heroku
- In-Memory throwaway Neo4j instances
- Simple Console UI with Graph Viz and Table Results
- One Click sharing
- Repl, SandBox, Bug-Reporter, Modeling Questions

Console - Sandbox, a useful Tool

How does it work?

- Tiny Webapp on Heroku
- In-Memory throwaway Neo4j instances
- Simple Console UI with Graph Viz and Table Results
- One Click sharing
- Repl, SandBox, Bug-Reporter, Modeling Questions

People love it

Graph Setup:

```

CREATE (westeros { name: "Westeros" })
CREATE (targaryen { house:"Targaryen" }),(stark { house:"Stark" }),(lannister { house:"Lannister" }),(baratheon { house:"Baratheon" })
FOREACH (house IN [stark,lannister,baratheon,targaryen,tully]| CREATE house-[:HOUSE]->westeros)
CREATE (danaerys { name:"Danaerys" }), danaerys-[:OF_HOUSE]->targaryen,(drogo { name:"Khal Drogo" }), danaerys-[:MARRIED_TO]->drogo
(:OF_HOUSE)->baratheon,(rickard { name:"Rickard" }), rickard-[:OF_HOUSE]->stark,(ned { name:"Eddard" }), ned-[:CHILD_OF]->rickard,(catelyn
), jon-[:CHILD_OF]->ned
FOREACH (child IN ["Robb", "Bran", "Arya", "Sansa", "Rickon"]| CREATE UNIQUE ned<-[:CHILD_OF]-({ name:child })-[:CHILD_OF]->catelyn)
FOREACH (child IN ["Cersei", "Jamie", "Tyrion"]| CREATE UNIQUE tywin<-[:CHILD_OF]-({ name:child }))
FOREACH (brother IN ["Robert", "Renly", "Stannis"]| CREATE UNIQUE steffon<-[:CHILD_OF]-({ name:brother }))
FOREACH (child IN ["Joffrey", "Myrcella", "Tommen"]| CREATE UNIQUE tywin<-[:CHILD_OF]-({ name:child })-[:CHILD_OF]-({ name:tywin }))
CREATE UNIQUE steffon<-[:CHILD_OF]-({ name:robert })-[:MARRIED_TO]-({ name:cersei })-[:CHILD_OF]->tywin
CREATE UNIQUE ned<-[:CHILD_OF]-({ name:sansa })-[:PROMISED_TO]->({ name:joffrey })-[:CHILD_OF]->cersei
CREATE UNIQUE ned<-[:CHILD_OF]-({ name:sansa })-[:MARRIED_TO]->({ name:tyrion })-[:CHILD_OF]->steffon
  
```

Query:

```

MATCH westeros<-[:HOUSE]-house<-[:OF_HOUSE]-ancestor, family=ancestor<-[:CHILD_OF*0..]-last
RETURN house.house, collect(DISTINCT last.name)
  
```

house.house	collect(distinct last.name)
Tully	[Catelyn, Robb, Bran, Arya, Sansa, Rickon]
Baratheon	[Steffon, Robert, Renly, Stannis, Tyrion]
Stark	[Rickard, Eddard, Jon, Robb, Bran, Arya, Sansa, Rickon]
Targaryen	[Danaerys]
Lannister	[Tywin, Cersei, Joffrey, Myrcella, Tommen, Jamie, Tyrion]

Query took 17 ms and returned 5 rows.

[Result Details](#)

Winter is Coming

```

MATCH westeros<-[:HOUSE]-house<-[:OF_HOUSE]-ancestor, family=ancestor<-[:CHILD_OF*0..]-last
RETURN house.house, collect(DISTINCT last.name)
  
```


AsciiDoc

AsciiDoc - the better Markup Language

- Full Toolchain for book generation
- lots of options
- still easy to read text files
- generates html, pdf, text
- The Neo4j Manual is pure AsciiDoc goodness

AsciiDoc Example (source)

```

1 == Basic AsciiDoc formatting
2
3 [width="50%",cols="1m,1a"]
4 |===
5 | \_Italic_ | _Italic_
6 | \*Bold* | *Bold*
7 | \`Monospace` | `Monospace`
8 | `http://www.neo4j.org/` | http://www.neo4j.org/
9 | `http://www.neo4j.org/[neo4j.org]` | http://www.neo4j.org/[neo4j.org]
10 | `link:../?5956246[Link to a GraphGist]` | link:../?5956246[Link to a GraphGist]
11 |===
12
13 Headings:
14
15 = Heading 1
16 == Heading 2
17 === Heading 3
18
19 Images:
20
21 image::http://assets.neo4j.org/img/still/cineasts.gif[]
  
```

Basic AsciiDoc formatting

<code>_Italic_</code>	<i>Italic</i>
<code>*Bold*</code>	Bold
<code>`Monospace`</code>	<code>Monospace</code>
<code>http://www.neo4j.org/</code>	http://www.neo4j.org/
<code>http://www.neo4j.org/[neo4j.org]</code>	neo4j.org
<code>link:./?5956246[Link to a GraphGist]</code>	Link to a GraphGist

AsciiDoc Example (rendered source)

Headings:

```
= Heading 1
= Heading 2
```

Basic AsciiDoc formatting

`_Italic_`

Italic

`*Bold*`

Bold

``Monospace``

`Monospace`

`http://www.neo4j.org/`

<http://www.neo4j.org/>

`http://www.neo4j.org/[neo4j.org]`

[neo4j.org](http://www.neo4j.org/)

`link:../?5956246[Link to a GraphGist]`

[Link to a GraphGist](#)

Headings:

```
= Heading 1
== Heading 2
=== Heading 3
```

AsciiDoc Example (rendered)

Images:

```
image::http://assets.neo4j.org/img/still/cineasts.gif[]
```

AsciiDoctor - the better AsciiDoc

- Reimplementation in Ruby (also jRuby)
- **Much** faster
- lots of extensions
- support for deck.js (you see it)
- Cross-Compiled to Javascript (Opal.js)

Focus, Michael

What was the question again?

How to present a live, graph model?

What is a gist?

- A useful snippet of information
- Easy to share, fork and change
- Nicely rendered and presented

What is a GraphGist?

an AsciiDoc file with:

- a graph domain model (cypher)
- describing text and pictures
- some example queries checked against the model
- interactively executable (Cypher)
- a Neo4j Console for further exploration

Full Source

```

1 = The Game of Thrones in Neo4j
2
3 image::http://maxdemarzidotcom.files.wordpress.com/2013/06/neoiscoming.jpg?w=580[]
4
5 == The setup
6
7 //hide
8 [source,cypher]
9 ----
10 CREATE
11   (_0 { name:"Westeros" }),
12   (_1 { house:"Tully" }),
13   ...
14   (_13 { name:"Danaerys" }),
15   ...
16   (_28 { name:"Tyrion" }),
17   _1-[:HOUSE]->_0,
18   _13-[:MARRIED_TO]->_12,
19   _14-[:CHILD_OF]->_8,
20   ...
21   _27-[:CHILD_OF]->_19, _28-[:CHILD_OF]->_10
22 ----
23
24 //graph
25
26 == Find all children of all houses
27
28 [source, cypher]
29 ----
30 MATCH (westeros)-[:HOUSE]-(house)-[:OF_HOUSE]-(ancestor), family=(ancestor)-[:CHILD_OF*0..]- (last)
31 WHERE westeros.name='Westeros'
32 RETURN house.house, collect(DISTINCT last.name)
33 ----
34
35 //table
36
37 == Find all the children of parents that are siblings
38
39 [source,cypher]
40 ----
41 MATCH (kid)-[:CHILD_OF]->(parent1)-[:CHILD_OF]->(ancestor)-[:CHILD_OF]-(parent2)-[:CHILD_OF]-(kid)
42 RETURN DISTINCT kid.name as name
43 ----
44
45 //table

```

Source: Setup

```

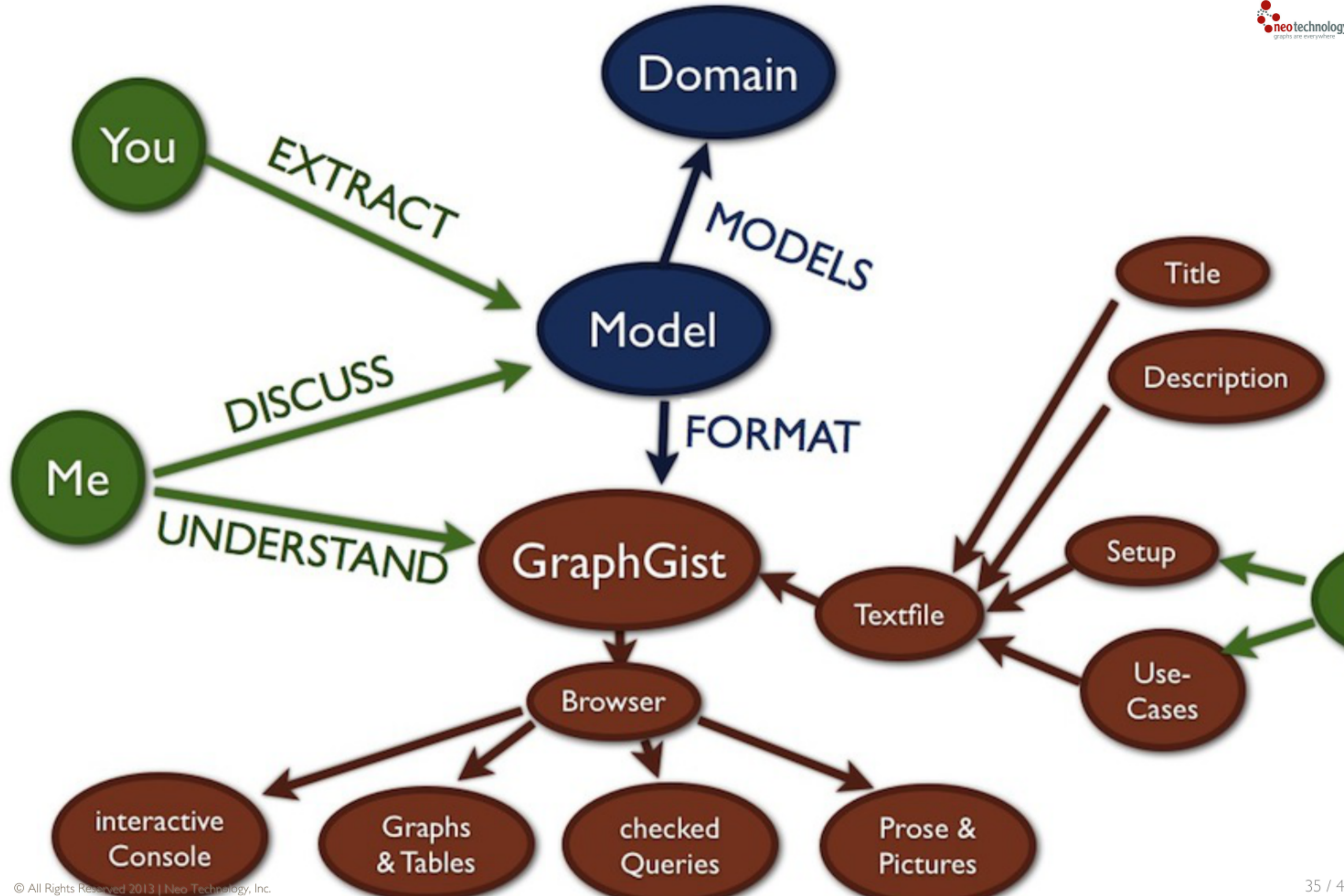
1 = The Game of Thrones in Neo4j
2
3 image::http://maxdemarzidotcom.files.wordpress.com/2013/06/neoiscoming.jpg?w=580[]
4
5 == The setup
6
7 //hide
8 [source,cypher]
9 ----
10 CREATE
11 (_0 { name:"Westeros" }),
12 (_1 { house:"Tully" }),
13 ...
14 (_13 { name:"Danaerys" }),
15 ...
16 (_28 { name:"Tyrion" }),
17 _1-[:HOUSE]->_0,
18 _13-[:MARRIED_TO]->_12,
19 _14-[:CHILD_OF]->_8,
20 ...
21 _27-[:CHILD_OF]->_19, _28-[:CHILD_OF]->_10
22 ----
23
24 //graph
  
```

Source: Use-Case

```

1 == Find all children of all houses
2
3 [source, cypher]
4 ----
5 MATCH (westeros)<-[:HOUSE]-(house)<-[:OF_HOUSE]-(ancestor), family=(ancestor)<-[:CHILD_OF*0..]-(
6 WHERE westeros.name='Westeros'
7 RETURN house.house, collect(DISTINCT last.name)
8 ----
9
10 //table
11
12 //graph

```

Power Tool Combo

Power Combination of Cool Tools

- AsciiDoc(tor) running in Browser with Opal.js
- Cypher
- Neo4j Console
- JavaScript (D3.js, jQuery)
- HTML5 (postMessage)
- GitHub Gists

The Glue? Javascript

How does it work (Rendering)?

1. Load Gist file from GitHub Gist / Url
2. Render AsciiDoc to HTML5 on the fly
3. Write to page / CSS
4. Placeholder replacement

How does it work (Queries) ?

1. Instantiate Console IFrame
2. Find all setup- and use-case queries
3. Send to Console, Check Results
4. Render Results as Table or Graph
5. Reset and Show Console

Next Steps ?

- We have a great tool for model documentation
- Now we need some graph models, content
- Don't make something up!
- Ask your users, your Community
- Create a Challenge (or two)

First Challenge (September)

- Give me anything
- Nice Prizes (Money, Books, T-Shirts, Tickets)
- Impressive 17 submissions in 4 weeks

First Challenge (September)

- Give me anything
- Nice Prizes (Money, Books, T-Shirts, Tickets)
- Impressive 17 submissions in 4 weeks

Winners

1. US Flights & Airports by Nicole White
2. Learning Graph by Johannes Mockenhaupt
3. Chess Games and Positions by Wes Freeman

Second Challenge (Winter Dec+Jan)

- 10 Categories from Education, Transport up to Advanced aka. "Show Off"
- Unbelievable 65 submissions in 8 weeks w/ holidays
- High quality content
- Hard to choose winners

Winter Challenge Winners

- 11 Winners in the categories, just 3 examples
- Organization Learning by @luannem - covering your path through courses and certifications in a learning management system.
- Single Malt Scotch Whisky by @patbaumgartner is my personal favorite, you certainly know why
- Amazon Web Services Global Infrastructure Graph by @aidanjcasey represents all regions, zones, services and instance types as a graph

What's in for you?

- Use them !
- GraphGists are fun
- They help you model, communicate and discuss your domain
- Great for Stackoverflow questions and answers
- Free to use, any AsciiDoc-File-URL will do
- Rendered in your Browser

Can I win something?

- Sure
- Submit your original GraphGist through this form <http://bit.ly/graphgist>
- Get A T-Shirt
- Be famous

Questions ? Thank You!