

Yes We Scan! Software Analysis Using jQAssistant And Neo4j

**JavaOne Latin America
São Paulo 2016**

Dirk Mahler

AGENDA

- Black Boxes Called Artifacts
- jQAssistant
- Software As A Graph
- Let's Explore Libraries!

Yes We Scan!
Software Analysis Using jQAssistant And Neo4j

Black Boxes Called Artifacts

■ Artifact

- Result of a build/integration processes
- Black Box – What does it hide?

acme.war

```
/index.xhtml  
/WEB-INF/lib/model.jar
```

Person.java

```
@Entity  
public class Person {  
    private String name;  
}
```

pom.xml

```
<dependency>  
    <groupId>org.eclipse.persistence</groupId>  
    <artifactId>persistence-api</artifactId>  
    <version>2.1.0</version>  
</dependency>
```

model.jar

```
/com/acme/model/Person.class  
/META-INF/persistence.xml
```

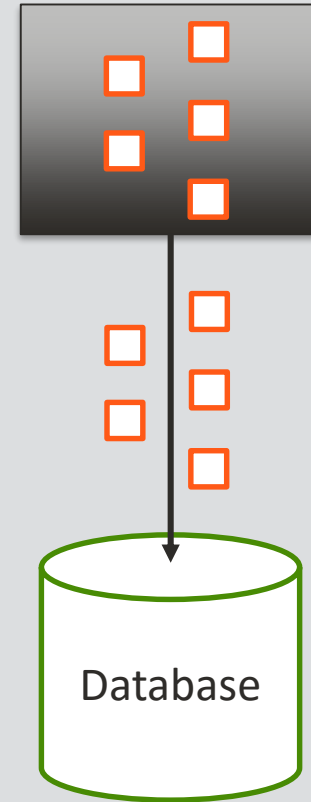
person.sql

```
create table PERSON (id number, name varchar(255))
```

- Let's open it!

■ The Idea

- Scan software structures
- Store in a database
- Execute queries
- Create reports



Yes We Scan!
Software Analysis Using jQAssistant And Neo4j

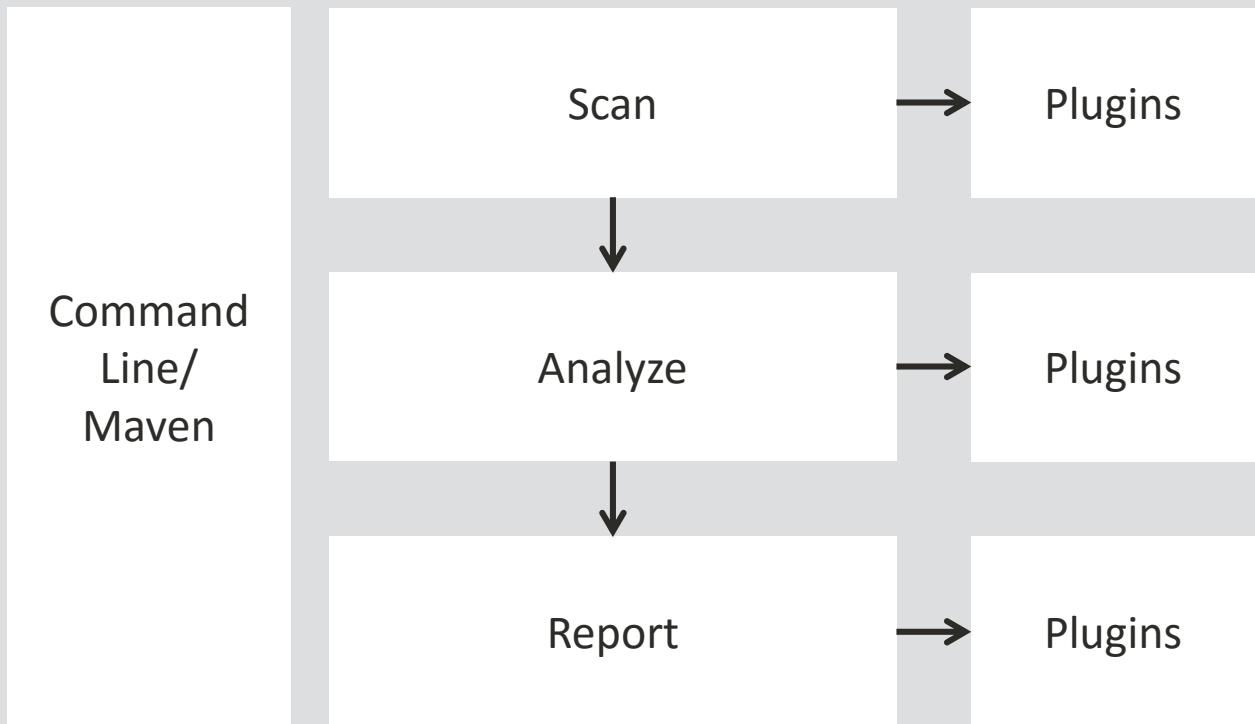
jQAssistant

jQAAssistant

<http://jQAAssistant.org>

- Open Source: GPLv3
- Current release: 1.1.3
 - initiated: 03/2013
 - first stable release: 04/2015
- Neo4j Community Edition
 - <http://neo4j.org>
 - embedded, no installation necessary





■ Getting Started – Command Line

- `unzip jqassistant.distribution-1.x.x.zip`
- `cd jqassistant.distribution-1.x.x/bin`

- `jqassistant.sh scan -f model.jar`
- `jqassistant.sh scan -f acme.war`
- `jqassistant.sh scan -f acme.ear`
- `jqassistant.sh scan -u http://somewhere.com/acme.ear`
- `jqassistant.sh scan -u`
`maven:repository::http://host/releases`
- `jqassistant.sh scan -u`
`rdbms:schema::jdbc:oracle:thin:user/secret@host:1521:sid`

- `jqassistant.sh server`
→ <http://localhost:7474>

■ Getting Started – Maven Project

```
<build>
  <plugins>
    <plugin>
      <groupId>com.buschmais.jqassistant.scm</groupId>
      <artifactId>jqassistant-maven-plugin</artifactId>
      <version>1.x.x</version>
    </plugin>
  </plugins>
</build>
```

- `mvn install jqassistant:scan`
- `mvn jqassistant:server`
- <http://localhost:7474>

■ Available scanner plugins

ZIP

JAR, WAR, EAR

GZ

*.class

*.properties

*.yaml

MANIFEST.MF

/META-INF/services/*

web.xml

application.xml

beans.xml

pom.xml

surefire-reports.xml

*.xsd

Liquibase

CheckStyle

FindBugs

JaCoCo

Git

■ Plugin API is public

M2 Repository

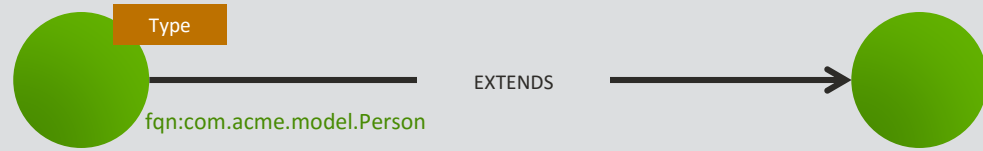
RDBMS Schema

Yes We Scan!
Software Analysis Using jQAssistant And Neo4j

Software As A Graph

- All we need is...

- Nodes
- Labels
- Properties
- Relationships

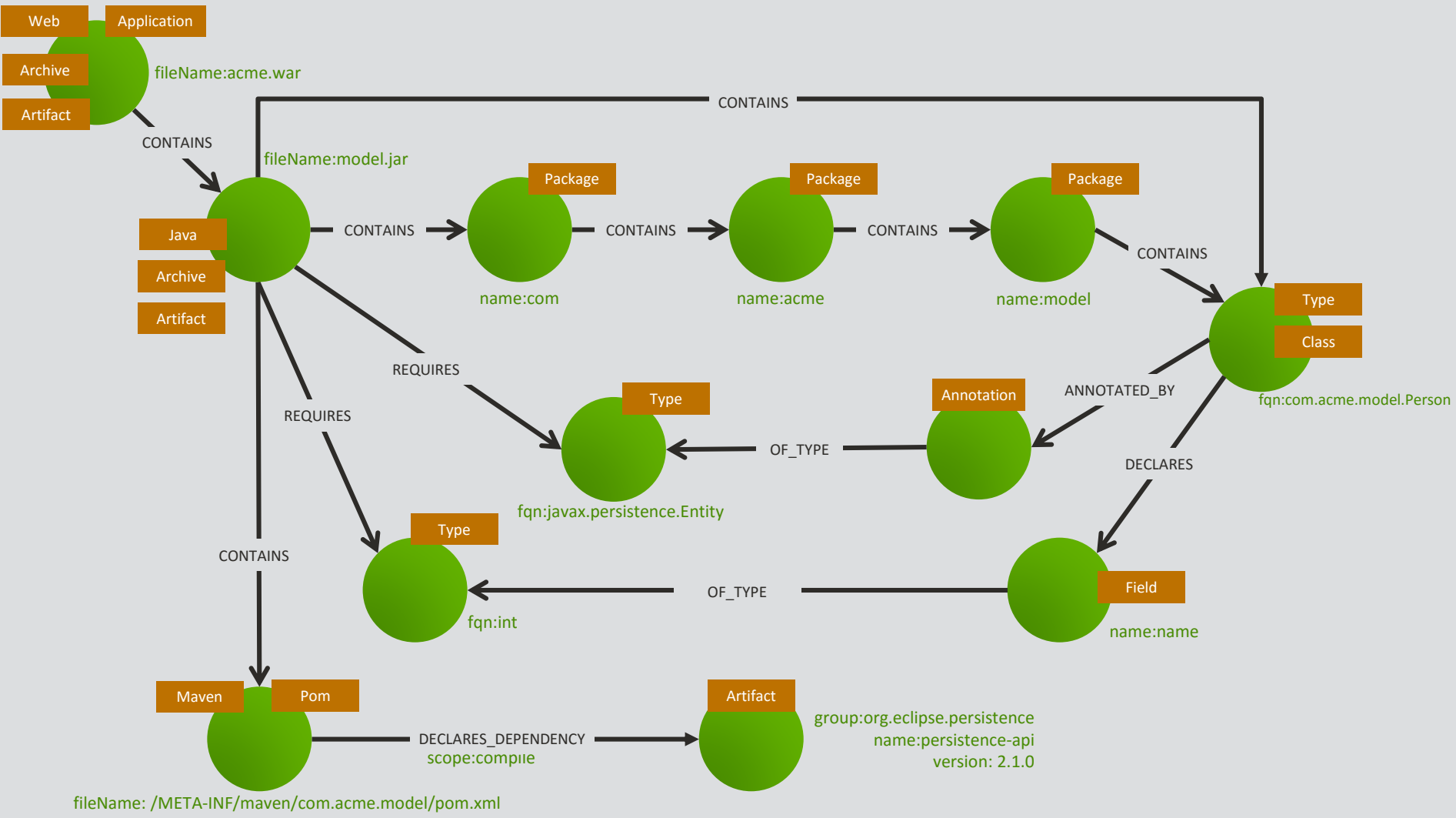


- Modeling is just...

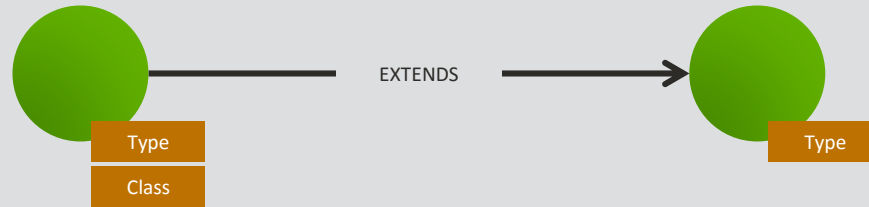
- Taking a pen
- Drawing the structures on a whiteboard (i.e. the database)

- We don't need...

- Foreign keys
- Tables and schemas
- Knowledge in graph theory

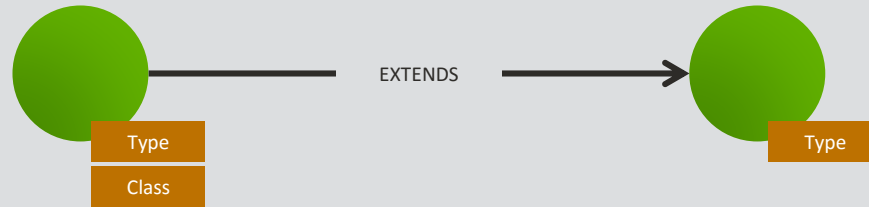


- Explore an application using queries
 - Which class extends from another class?



- Let's convert this to ASCII art...
 - () as nodes
 - -[]-> as directed relationships

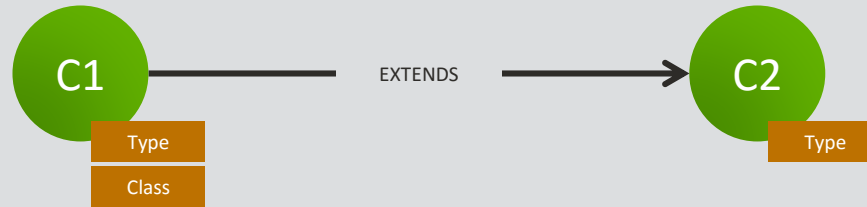
- Explore an application using queries
 - Which class extends from another class?



- Let's convert this to ASCII art...
 - () as nodes
 - -[]-> as directed relationships

() - [] -> ()

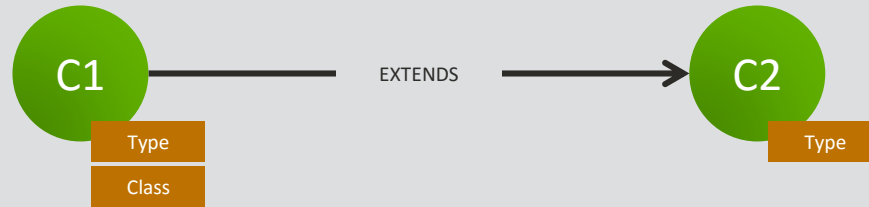
- Explore an application using queries
 - Which class extends from another class?



- Let's convert this to ASCII art...
 - () as nodes
 - -[]-> as directed relationships

(c1) - [] -> (c2)

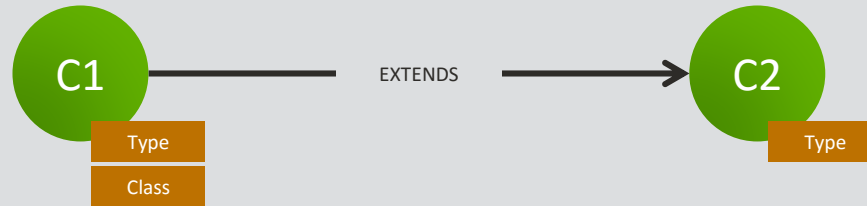
- Explore an application using queries
 - Which class extends from another class?



- Let's convert this to ASCII art...
 - () as nodes
 - -[]-> as directed relationships

`(c1) - [:EXTENDS] -> (c2)`

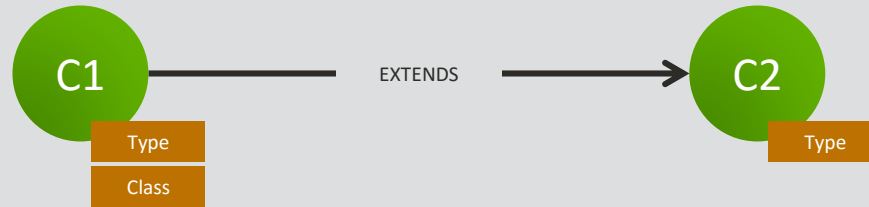
- Explore an application using queries
 - Which class extends from another class?



- Let's convert this to ASCII art...
 - () as nodes
 - -[]-> as directed relationships

`(c1:Class) - [:EXTENDS] -> (c2:Type)`

- Explore an application using queries
 - Which class extends from another class?



- Pattern matching is the core principle of Cypher

MATCH

```
(c1:Class) -[:EXTENDS]->(c2:Type)
```

RETURN

```
c1.fqn, c2.fqn
```

Yes We Scan!
Software Analysis Using jQAssistant And Neo4j

Let's Explore Libraries!

- Examples
 - Artifacts and Maven descriptors
 - API changes between releases

 - Declared deprecations
 - Thrown exceptions
 - Anti-Patterns
 - Structural problems (metrics)
 - Declared members
 - Depth of inheritance hierarchies
 - Cyclomatic complexity
 - Fan In/Fan Out of classes and packages
 - Package Cycles

Yes We Scan!
Software Analysis Using jQAssistant And Neo4j

Live Demo

<http://jqassistant.org/detecting-differences-between-jar-files/>

<https://jqassistant.org/yes-we-scan-exploring-libraries/>

JQA Assistant

- Holistic and extensible view on code structures
 - Graph data model
 - Plugin architecture

- Flexible exploration of structures using queries
 - Expressive query language - Cypher
 - Powerful searches beyond IDE capabilities
 - Individual metrics and visualization

Thank You! – Questions?

Web: jqassistant.org

Twitter: [@jqassistant](https://twitter.com/jqassistant)

Google Group: [jqAssistant](https://groups.google.com/jqassistant)

Stackoverflow Tag: [jqAssistant](https://stackoverflow.com/questions/tagged/jqAssistant)