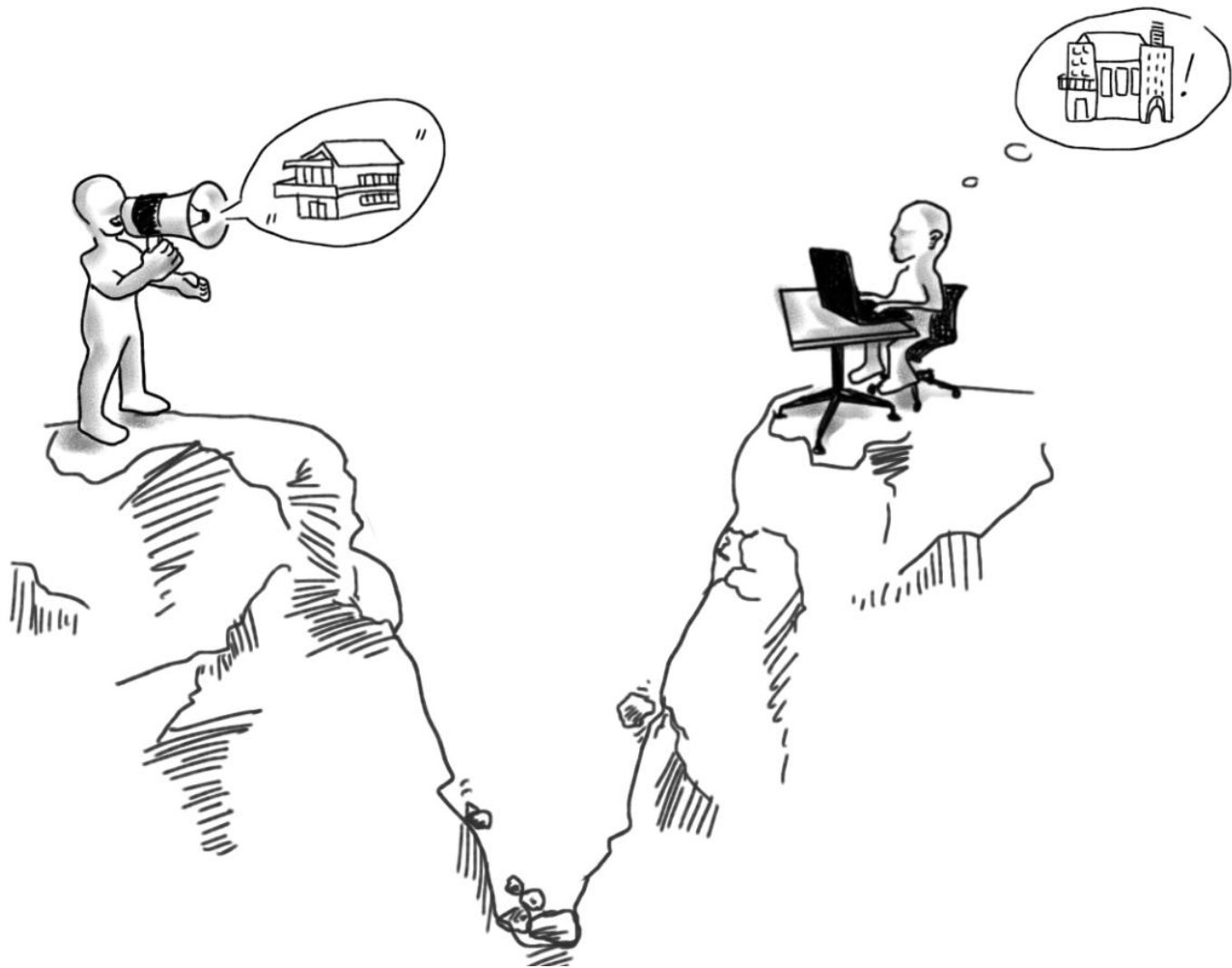


Mind The Gap!

Architecture vs. Code

Java User Group Saxony
Dec 2015

Oliver B. Fischer - E-Post Development GmbH
Dirk Mahler - buschmais GbR



Was ist Software Architektur?

Eine strukturierte oder hierarchische Anordnung der Systemkomponenten sowie die Beschreibung Ihrer Beziehungen.

Helmut Balzert

Was ist Software Architektur?

The fundamental organisation of a system embodied in its components, their relationships to each other, and the environment, and the principles guiding its design and evolution.

IEEE 1471-2000

Was ist Software Architektur?

SEI @ CMI: John Carter

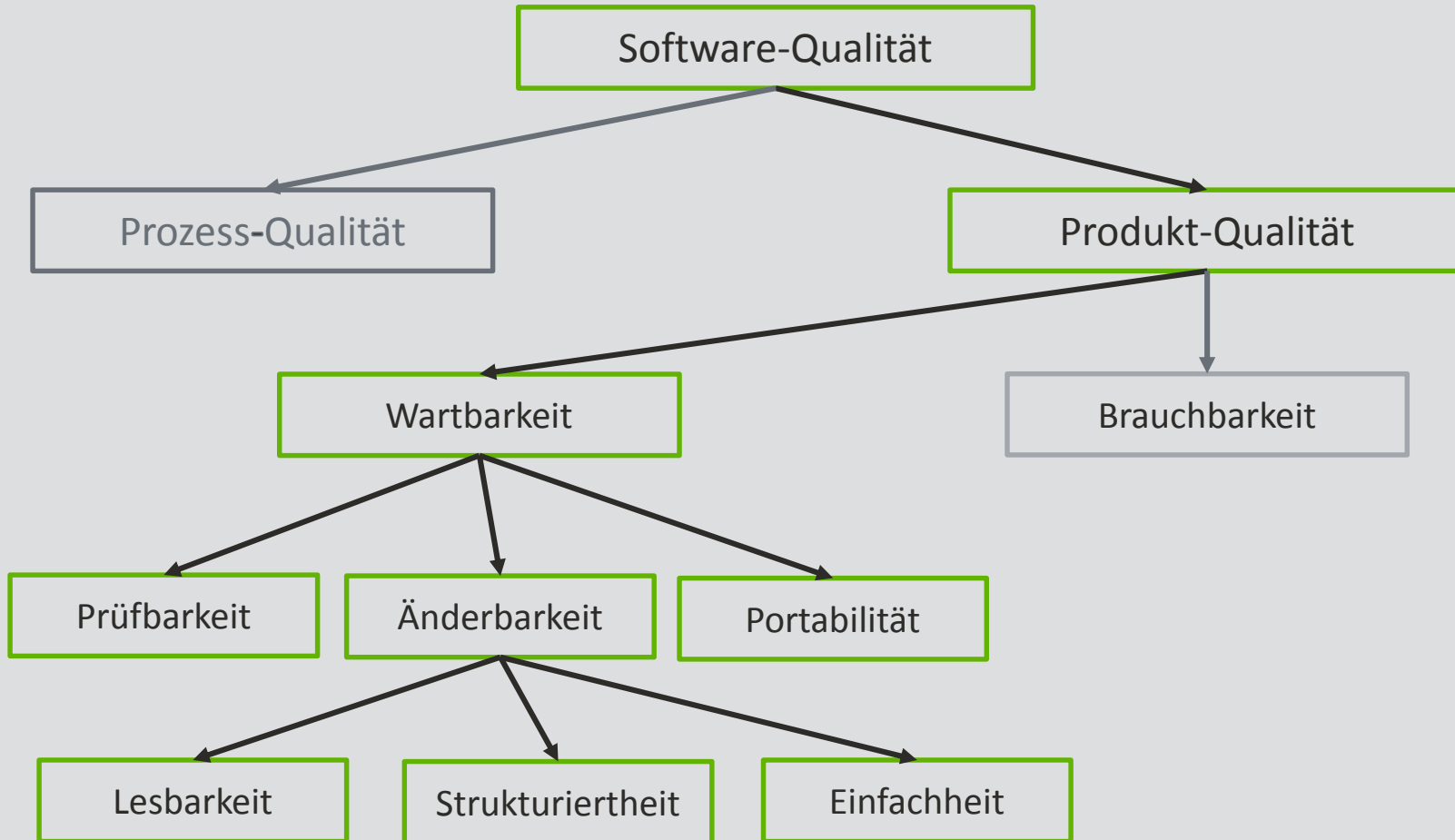
...from the plethora of definitions the vary concept of Software Architecture is fuzzy to the point of utter uselessness...

A lot of folks mean:

That peculiar consistency and homogeneity of Design that would give arise to the ease of use and reuse, which is wished for by the developer who calls himself „The Software Architect“.

...[which] implodes immediatly on the appointment of a second Software Architect...

Ziele von Software Architektur



Ziele von Software Architektur

Durch Software Architektur wollen wir erhalten:

- Wartbare Software
- Verständliche Software
- Anpassbare bzw. Erweiterbare Software
- Effiziente Software

Wann Software Architektur entsteht

Software Architektur entsteht sobald sich zwei Personen über einen Aspekt eines Dings oder das Zusammenspiel zweier Dinge abstimmen.

Wann Software Architektur entsteht

Software Architektur entsteht sobald sich zwei Personen [über einen Aspekt eines Dings oder das Zusammenspiel zweier Dinge] **abstimmen.**

Software Architektur ist Kommunikation

Software Architektur ist [auch] Kommunikation über Strukturen, Beziehungen und Prinzipien...

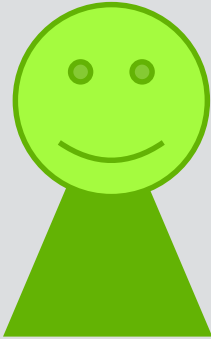
...um besser und leichter arbeiten zu können.

Software Architektur ist Kommunikation

Software Architektur muß Verständnis schaffen,
nicht nur für das **Wie**, sondern auch für das
Warum!

Software Architektur ist Kommunikation

Drei Rollen-Model



Domainexperte



Entwickler



User

Software Architektur ist Kommunikation

Einfachster Fall



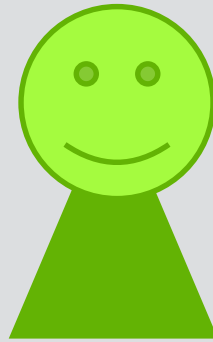
- Domainexperte
- Entwickler
- User

Software Architektur ist Kommunikation

Es wird schwieriger...



- Domainexperte



- Entwickler
- User

Software Architektur ist Kommunikation

Es wird schwieriger...



■ Domainexperte



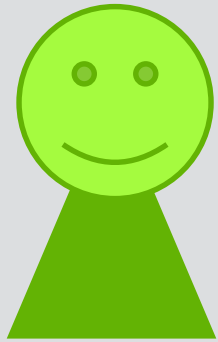
■ Entwickler



■ User

Software Architektur ist Kommunikation

Das Team wird größer



■ Domainexperte



■ Entwickler



■ User

Software Architektur ist Kommunikation

Es verteilt sich...



Team in Berlin



Team in München

Software Architektur ist Kommunikation

Es verteilt sich geographisch und kulturell



Ich spreche
Deutsch...



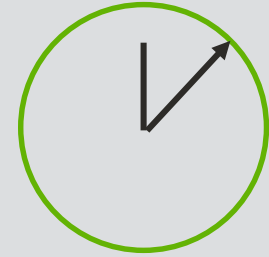
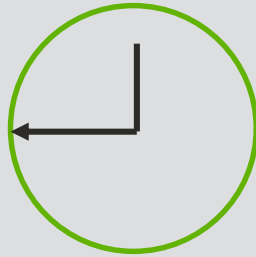
Я говорю
по-русский



I speak English

Software Architektur ist Kommunikation

Und dann auch zeitlich...



Altlasten & Technische Schulden

Veraltete Libraries....

Neue Lösungsansätze

Divergierende Makroarchitektur

Wir brauchen Regeln



Hast Du mal Zeit?

Ich habe da eine Idee.

...

Ja, so machen wir das!



Ja, was willst Du denn?

Laß mal hören...

...

- §1. Jedes System...
- §2. Jede Methode...
- §3. Auf jedem Server...
- §4. ...

Wir brauchen Regeln

Regeln entstehen meist informell

§1. Jedes System...

§2. Jede Methode...

§3. Auf jedem Server...

§4. ...

Wir brauchen Regeln

Informelle Regeln haben Grenzen...

- Größe des Teams erschwert Kommunikation
- Entstehung von Inselwissen
- Fluktuation
- Charaktertypen

Lösungsansatz

Überführung von informellen Regeln
in formale Regeln.



A rectangular box with a green border, containing 15 horizontal green lines, serving as a template for writing.

Lösungsansatz



„Wir schreiben das auf!“

„Wo?“

„Im Wiki!“

Zitat „Wiki, der Ort wo
Informationen zum Sterben
hingehen.“



Dokumentation

!=

Kommunikation

Lösungsansatz

Helfen Reviews?



Helfen Tools?

Mind The Gap!
Architecture vs. Code

jqAssistant

jQAAssistant

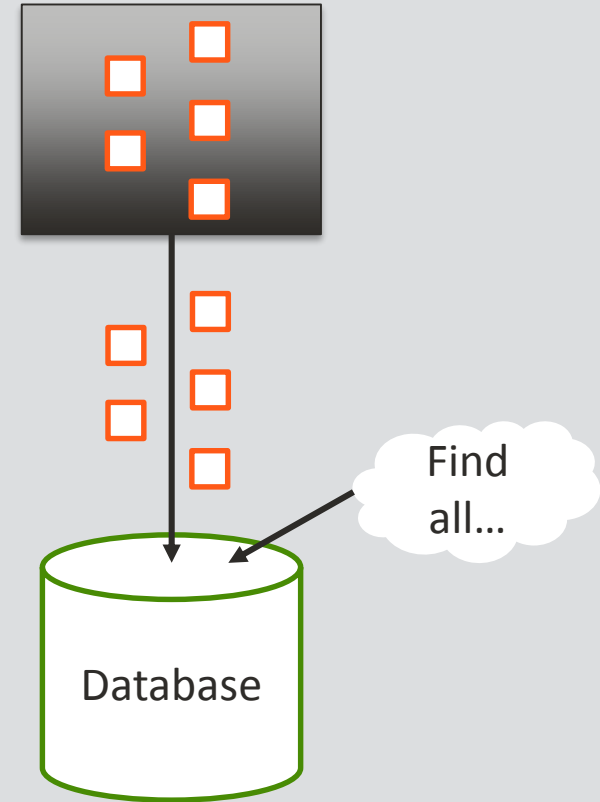
<http://jQAAssistant.org>

- Open Source
 - GPL v3
- Current release: 1.1.0
 - initiated: 03/2013
 - first stable release: 04/2015
- Neo4j Community Edition embedded
 - no installation necessary

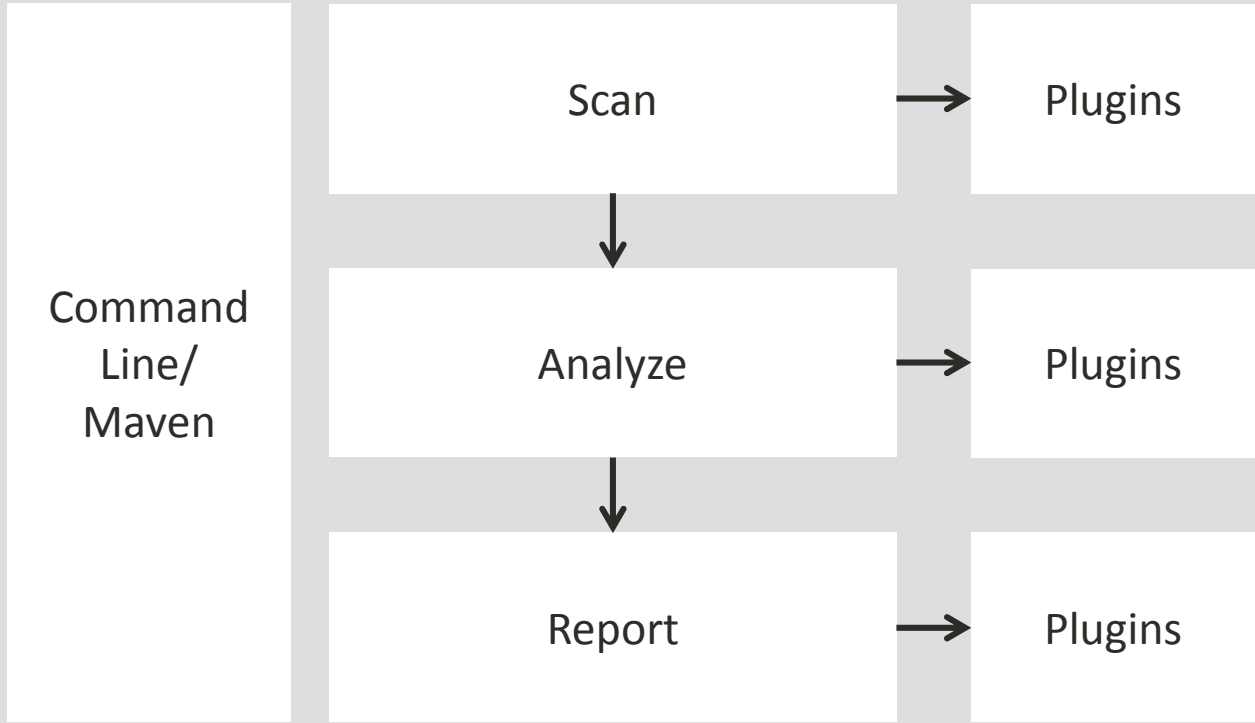


■ The Idea

- Scan software structures
- Store in a database
- Execute queries
 - Add higher level concepts
 - Find constraint violations
- Create Reports



→ Provide feedback to developers within build process



■ Available scanner plugins

ZIP

JAR, WAR, EAR

GZ

Liquibase

*.class

*.properties

*.yaml

CheckStyle

FindBugs

MANIFEST.MF

/META-INF/services/*

JaCoCo

Git

web.xml

application.xml

beans.xml

pom.xml

surefire-reports.xml

*.xsd

M2 Repository

RDBMS Schema

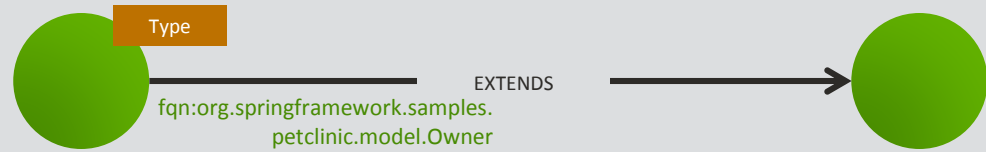
■ Plugin API is public

Mind The Gap!
Architecture vs. Code

Software As A Graph

- All we need is...

- Nodes
- Labels
- Properties
- Relationships

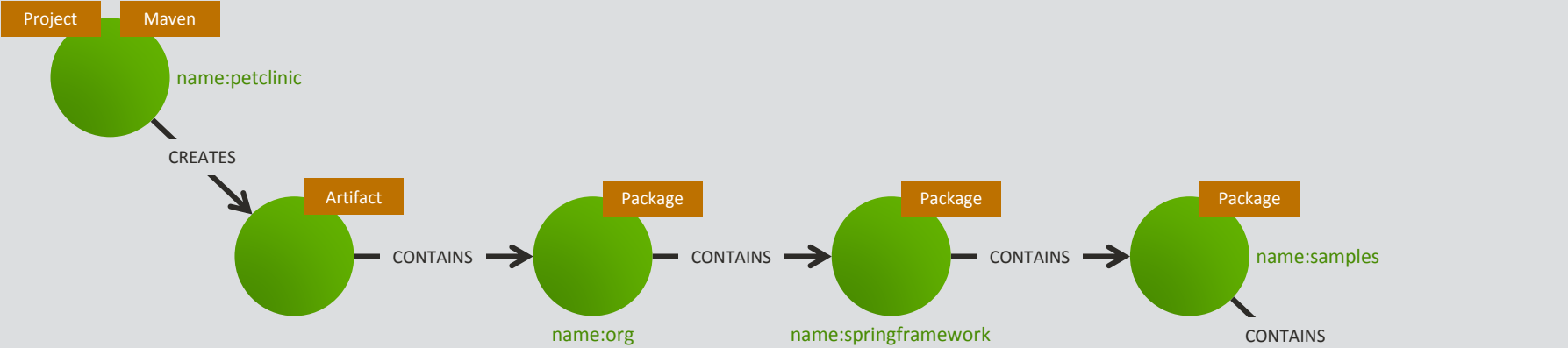


- Modeling is just...

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

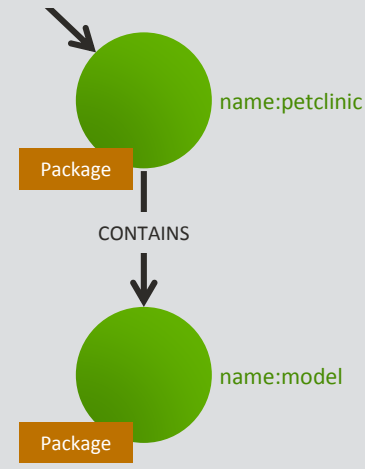
- We don't need...

- Foreign keys
- Tables and schemas
- Deep knowledge in graph theory



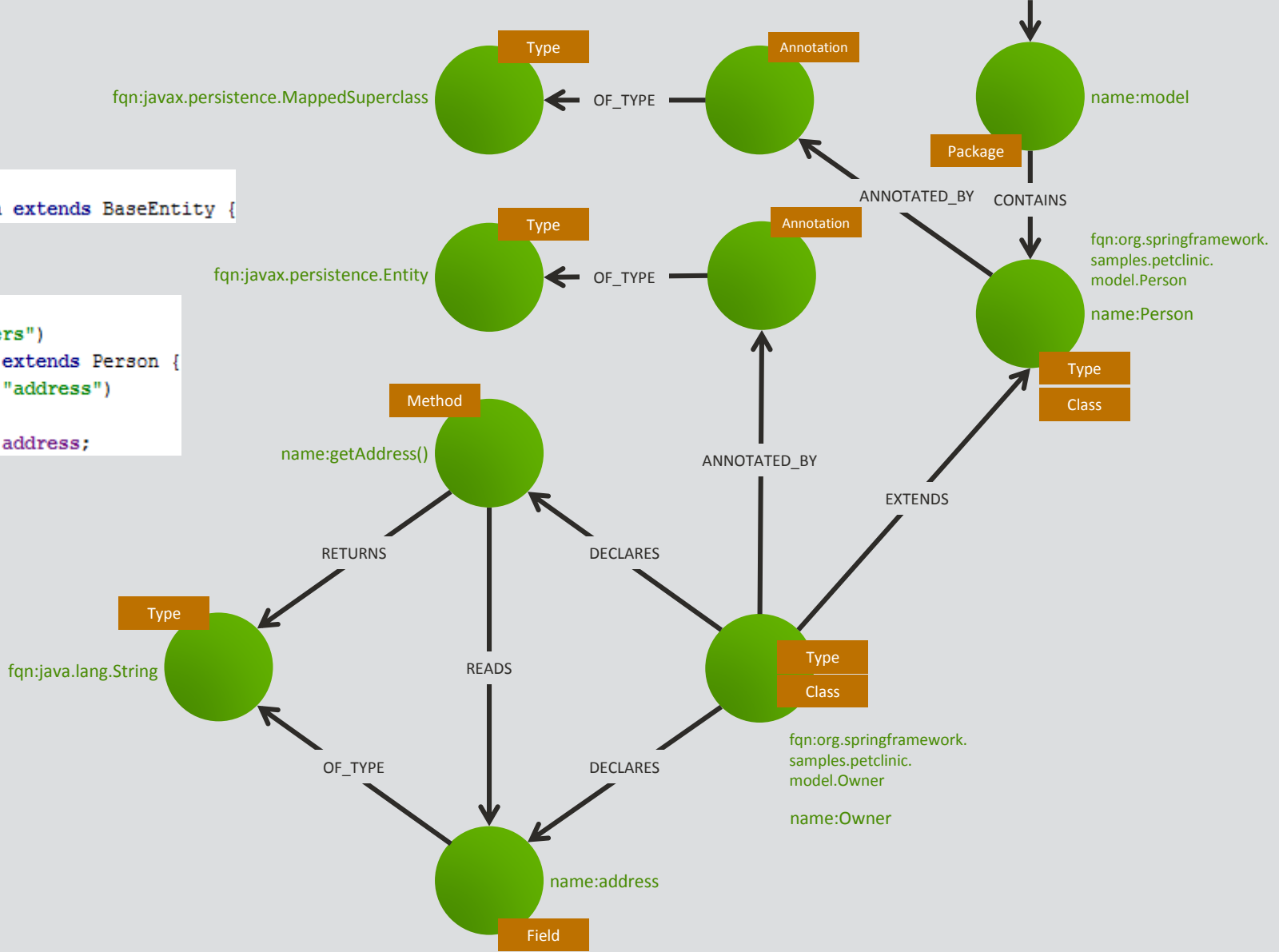
spring-petclinic (C:\Development\projects\spring-petclinic)

- ▶ .idea
- ▶ jqassistant
- ▼ src
 - ▼ main
 - ▼ java
 - ▼ org.springframework.samples.petclinic
 - ▼ model
 - BaseEntity
 - NamedEntity
 - Owner
 - package-info.java
 - Person
 - Pet
 - PetType
 - Specialty
 - Vet
 - Vets
 - Visit

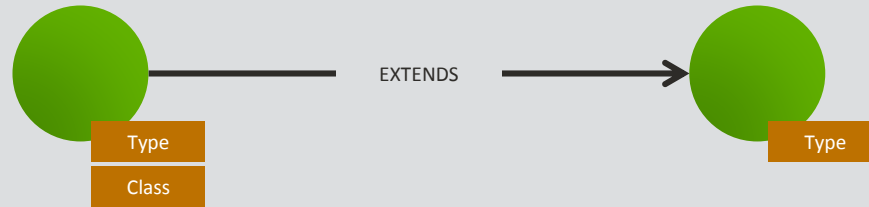


```
@MappedSuperclass
public class Person extends BaseEntity {
```

```
@Entity
@Table(name = "owners")
public class Owner extends Person {
    @Column(name = "address")
    @NotEmpty
    private String address;
```

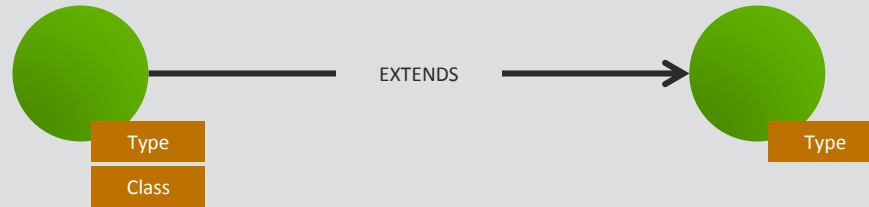


- Explore an application using queries
 - Which class extends from another class?



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

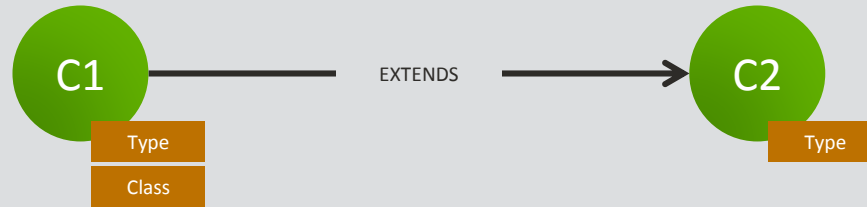
- Explore an application using queries
 - Which class extends from another class?



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

() - [] - > ()

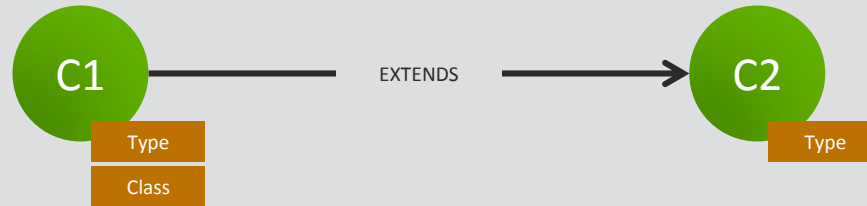
- Explore an application using queries
 - Which class extends from another class?



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

(c1) - [] -> (c2)

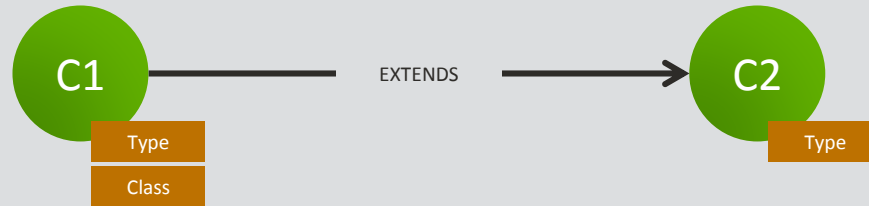
- Explore an application using queries
 - Which class extends from another class?



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

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

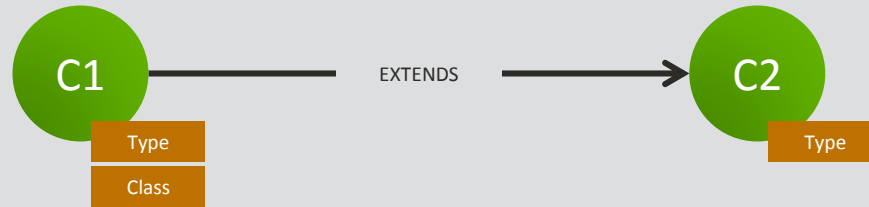
- Explore an application using queries
 - Which class extends from another class?



- 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
```

Mind The Gap!
Architecture vs. Code

Demo #1

<http://jqassistant.org/get-started/>

Mind The Gap!
Architecture vs. Code

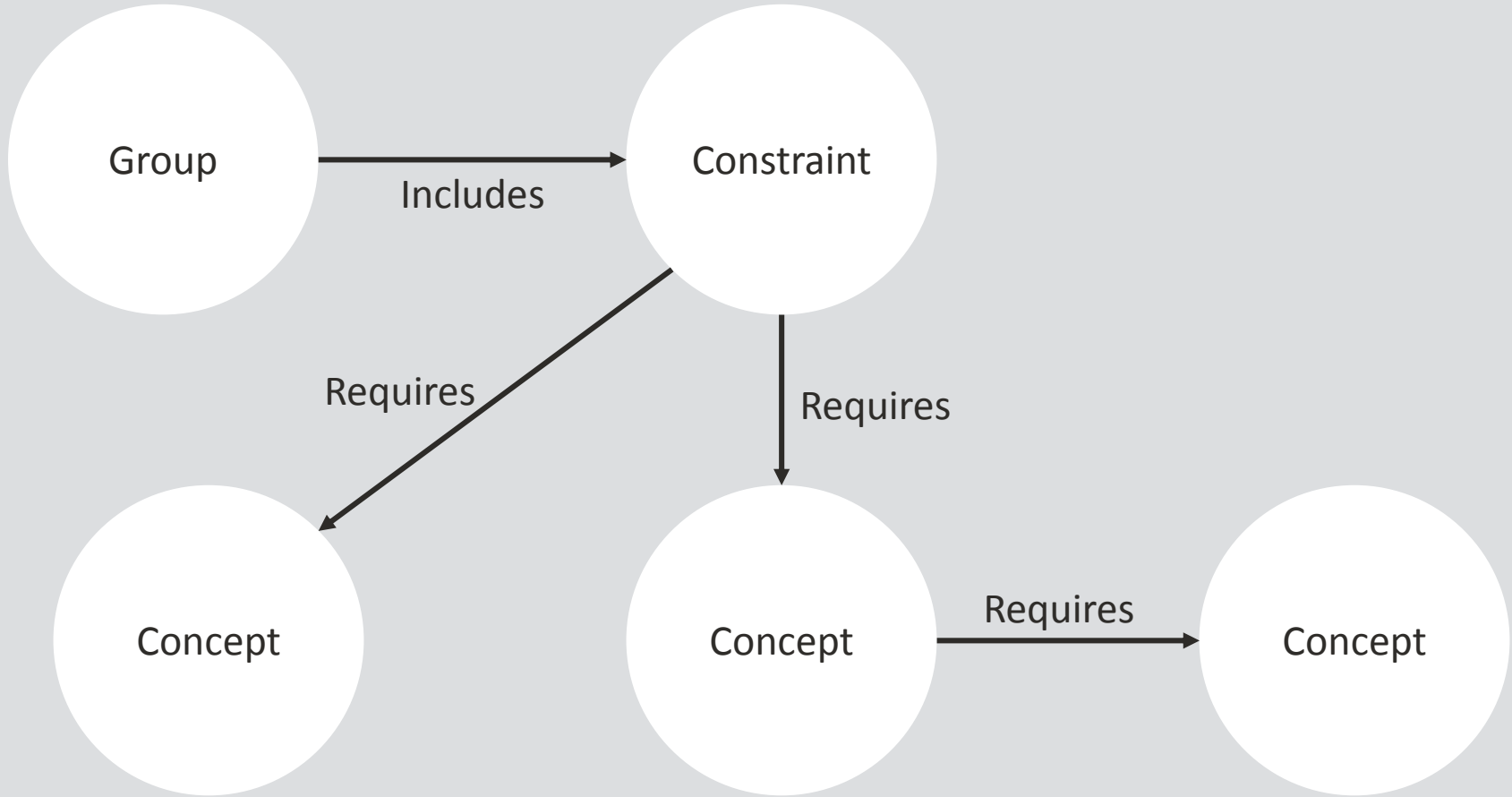
Building The Bridge

- Analyze
 - Execution of rules
 - Defined in AsciiDoc or XML documents

 - Concepts
 - Enrich data model

 - Constraints
 - Detect violations

 - Group
 - Allow different execution profiles



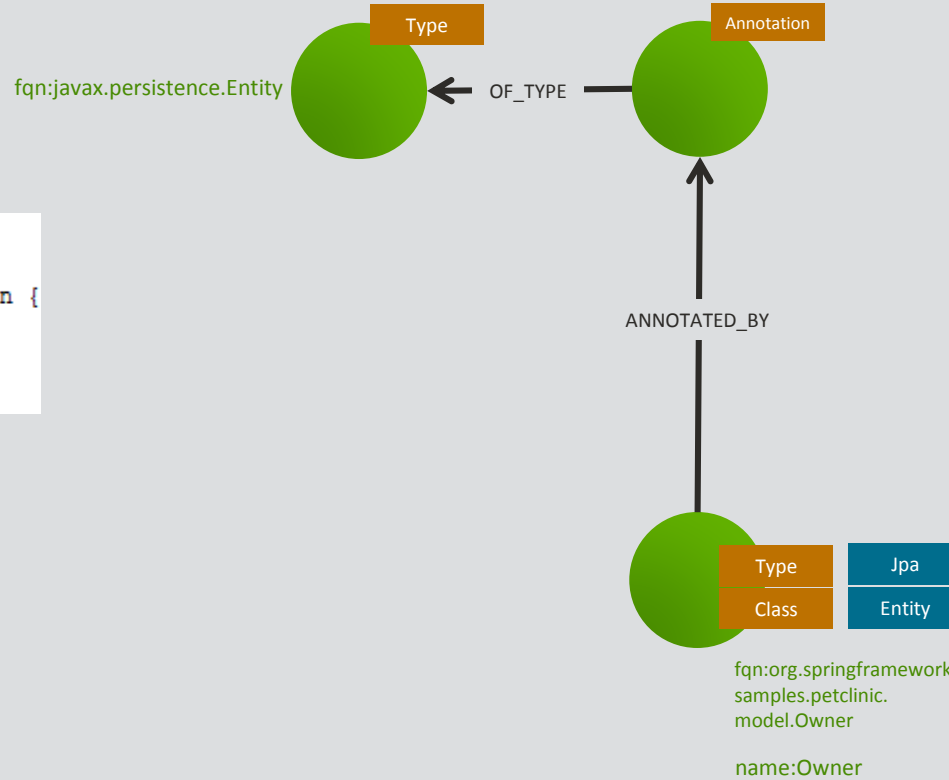
■ Concept

== JPA Entities

```
[[jpa2:Entity]]
.Labels all types annotated with @javax.persistence.Entity with
Jpa and Entity.
[source,cypher,role=concept]
----
MATCH
  (t:Type)-[:ANNOTATED_BY]->()-[:OF_TYPE]->(a:Type)
WHERE a.fqn ="javax.persistence.Entity"
  SET t:Jpa:Entity
RETURN t AS Entity
----
```


■ Concept

```
@Entity
@Table(name = "owners")
public class Owner extends Person {
    @Column(name = "address")
    @NotEmpty
    private String address;
```



■ Constraint

```
[[model:JpaEntityInModelPackage]]
```

.All JPA entities must be located in packages named "model".

```
[source,cypher,role=constraint,requiresConcepts="jpa2:Entity"]
```

```
----
```

```
MATCH
```

```
  (package:Package) -[:CONTAINS]->(entity:Jpa:Entity)
```

```
WHERE
```

```
  package.name <> "model"
```

```
RETURN
```

```
  entity AS EntityInWrongPackage
```

```
----
```

■ Group

```
[[default]]
```

```
[role=group,includesConstraints="model:JpaEntityInModelPackage"]
```

```
== Naming Rules
```

The following naming rules apply:

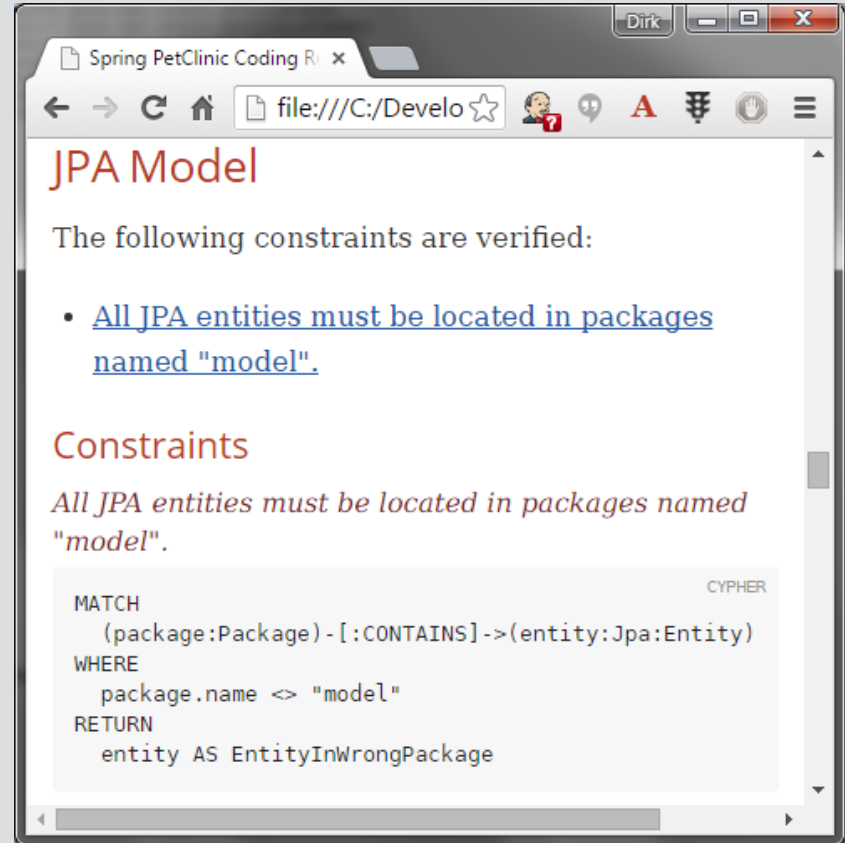
- <<model:JpaEntityInModelPackage>>

■ AsciiDoc

- Wiki syntax
- Rendering to
 - DocBook
 - HTML
 - PDF

■ Executable specification

- Coding rules
- Design & Architecture



Mind The Gap!
Architecture vs. Code

Demo #2

<http://github.com/buschmais/spring-petclinic>

jQAAssistant

- Holistic view on code structure and system architecture
- Exploration of existing structures
- Validation of rules
- Executable architecture and design specification
- Rules using terms from architecture & design language

Thank You! – Questions?

Mail: info@jqassistant.org

Web: jqassistant.org

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