SEMINAR: SOFTWARE AUS KOMPONENTEN

Generierung einer Spring Roo-Anwendung aus UML

Stefan Faulhaber

25. Januar 2016

Universität Leipzig

Agenda

- Motivation
  - Aufgabenstellung
  - Softwaregeneration
- Umsetzung
  - Vorgehensweise
  - Demo
- Ausblick
  - Erweiterbarkeit

MOTIVATION

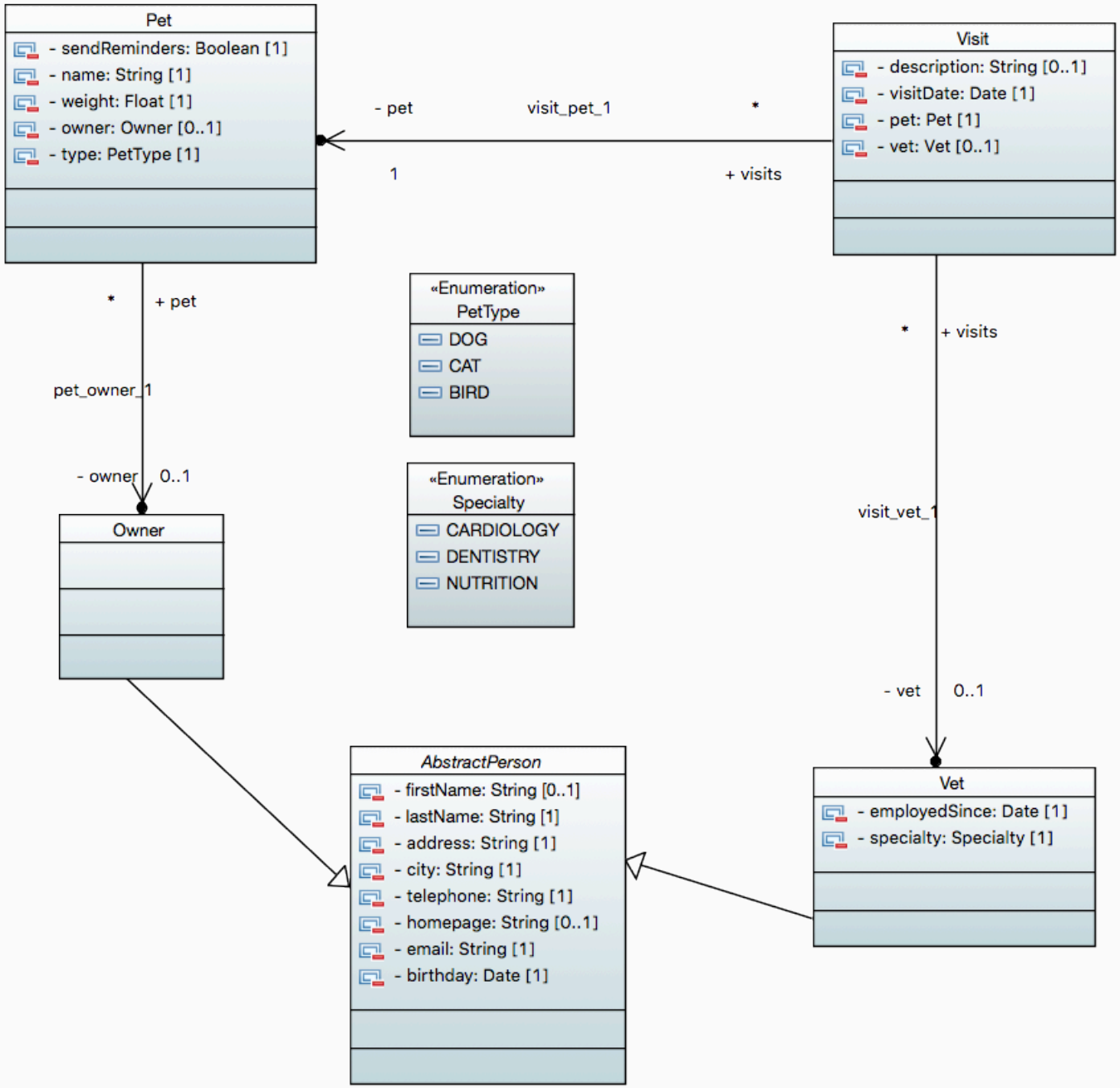
Aufgabenstellung

„Entwickeln Sie einen Softwaregenerator, der aus einem UML-Modell Scripte für Spring Roo erzeugt. Es ist möglich, auf einer existierenden Implementierung aufzubauen.“

Weiteres:

- UML-Modellierung mit Eclipse Papyrus
- Beispiel Diagramm: Pet Clinic
- Xpand und Xtend als Generatorsprachen
- Spring Roo in Version 1.3.2

Pet Clinic


Pet
- sendReminders: Boolean [1]
- name: String [1]
- weight: Float [1]
- owner: Owner [0..1]
- type: PetType [1]
- pet
visit_pet_1
*
1
+ visits
Visit
- description: String [0..1]
- visitDate: Date [1]
- pet: Pet [1]
- vet: Vet [0..1]
«Enumeration»
PetType
DOG
CAT
BIRD
«Enumeration»
Specialty
CARDIOLOGY
DENTISTRY
NUTRITION
* + pet
pet_owner_1
- owner 0..1
Owner
* + visits
visit_vet_1
- vet 0..1
AbstractPerson
- firstName: String [0..1]
- lastName: String [1]
- address: String [1]
- city: String [1]
- telephone: String [1]
- homepage: String [0..1]
- email: String [1]
- birthday: Date [1]
Vet
- employedSince: Date [1]
- specialty: Specialty [1]

Spring Roo

„*A next-generation rapid application development tool for Java developers.*“

- nutzt unter Anderem das Spring-Framework
- ermöglicht schnelles Prototyping einer Java-Anwendung
- manuelle Nutzung per Roo-Shell

Beispiel:

```
project --topLevelPackage com.helloworld
jpa setup --provider HIBERNATE --database MYSQL
entity jpa --class ~.Main
field string --fieldName text --notNull
```

Xpand Template Language

- geeignet für *Model-to-Text* Transformationen (Alternativen: JET, MTL)
- minimalistische Syntax (Escape-Zeichen: «»)
- Xpand Templates werden meistens innerhalb von MWE-Workflows ausgeführt
- Xpand-Dateiendung: `.xpt`
- nutzt *type-system* und *expression language* von Xtend
- Xtend-Dateiendung: `.ext`

Xpand Beispiel

```
«DEFINE root FOR uml::Model-»
«FILE name + ".txt"-»

// Project Setup
«EXPAND handleFQN FOREACH
ownedElement.typeSelect(uml::Package)-»

// Enumerations
«EXPAND handleEnums FOREACH ownedElement-»

// ...

«ENDFILE-»
«ENDDEFINE-»
```

Vorteile

- geringer overhead
- intuitiver Workflow
  - M2T-Transformation
  - keine Programmierkenntnisse erforderlich
  - vereinfachte Kommunikation der Stakeholder
- schnelles Prototyping
  - viele Iterationen
  - geringes Risiko
  - höhere Softwarequalität
  - Kosten- & Zeitersparnis
  - hilfreiche Ergebnisse (*proof of concept*)
- Erweiterung des Prototypen möglich

UMSETZUNG

Vorgehensweise (Xpand)

In Hauptschleife:

```
«EXPAND handleClasses FOREACH ownedElement-»
```

Hilfsfunktion:

```
«DEFINE handleClasses FOR uml::Package-»
«EXPAND handleClasses FOREACH ownedElement-»
«ENDDEFINE-»

«DEFINE handleClasses FOR uml::Class-»
«createClassJPA(this)»
«ENDDEFINE-»
```

Vorgehensweise (Xtend)

```
// create JPA class
String createClassJPA(uml::Class c):
  “entity jpa “ + classArg(c) + abstractArg(c)
  + extendsArg(c);

// class argument
String classArg(uml::NamedElement x):
  “--class ~.“ + x.name;

// abstract argument
String abstractArg(uml::Class c):
  c.isAbstract ? “ --abstract“ : ““;

// extends argument
String extendsArg(uml::Class c):
  c.generalization.isEmpty ? ““ : “ --extends ~.“
  + (c.generalization.get(0)).general.name;
```

Demo

Ergebnis & Quelltext

AUSBLICK

Erweiterbarkeit

1. Unterstützung weiterer Argumente
   - `--entity`
   - `--package`
   - ...
2. Implementierung weiterer Befehle
   - `finder`
   - `controller`
   - siehe Befehlsreferenz
3. Unterstützung von UML profiles
   - Abbildung weiterer Roo-Befehle in UML möglich
   - mehr Anwendungsfälle für den Generator
4. Evaluierung
   - Praxistauglichkeit
   - Effizienzgewinn

Quellen

Vielen Dank für die Aufmerksamkeit.

Quellen:

- http://projects.spring.io/spring-roo/
- http://docs.spring.io/spring-roo/docs/1.3.2.RELEASE/reference/html/
- http://git.eclipse.org/c/m2t/org.eclipse.xpand.git/plain/doc/org.eclipse.xpand.doc/manual/xpand_reference.pdf
- R. C. Gronback, eclipse Modeling Project - A Domain-Specific Language Toolkit, Addison-Wesley Verlag, 2009