

Beetlz

Software Model Consistency Checker for Eclipse

Eva Darulová

Supervisor: Dr. Joseph Kiniry

University College Dublin

14th May 2009



Overview

- Software Modelling
- JML & BON
- Examples
- Implementation
- So what?
- Demo
- Questions



Software Modelling

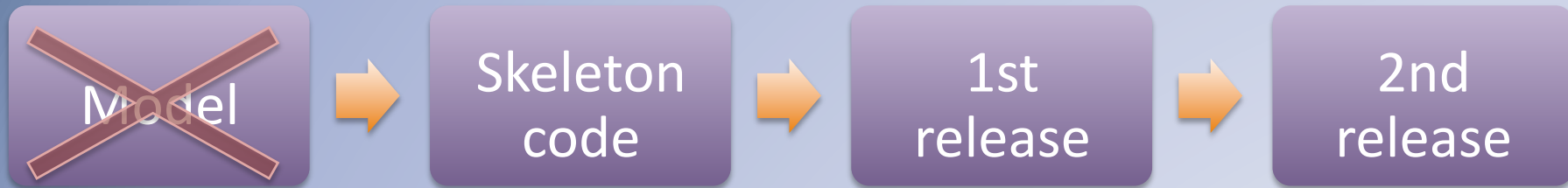
- abstract view of software
- implementation language independent
- different forms:
 - high-level or very detailed
 - graphical or textual
 - formal or informal

→ precise specification of the system
→ better understanding



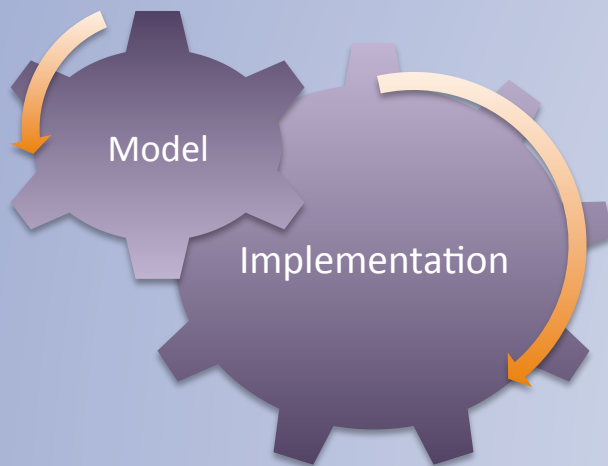
Consistency

- usually...



... model gets forgotten once coding starts

- better...



→ requires consistency checking
→ requires tool support

/*@

JML

- a.k.a. Java Modelling Language
- tool support available
- tailored for Java
 - familiarity for Java users
- supports Design by Contract

```
public class Keeper extends Personnel {  
    public /*@ non_null */ String name;  
    public Set < Animal > my_animals;  
  
    //@requires my_animals.contains(animal);  
    //@ensures !animal.isHungry;  
    public boolean feedAnimal (Animal animal) { ... }  
}
```

precondition

postcondition

BON

- a.k.a. Business Object Notation
- design process & notation
- graphical & textual
- object-oriented
- seamless development
- reversibility
- Design by Contract

→ simple and complete

```
class KEEPER
inherit PERSONNEL
feature
  name: STRING
  animals: SET[ANIMAL]
  ensure
    animals.length > 0;
  end
end
```



Example

```
class KEEPER
```



```
public class Keeper
```

```
class KEEPER  
inherit PERSONNEL
```



```
public class Keeper  
    implements ZooPersonnel
```

```
name: STRING
```



```
/*@non_null@*/ String name;
```

```
Require  
    0 < weight;
```



```
//@requires weight > 0;
```

```
Result = a + b * c;
```



```
\result = (a + b) * c;
```



Challenges

- naming conventions
- implementation details
 - private methods/constructors
 - getter & setter
 - Java standard library classes
- partial translations
- impossible translations
 - static in Java
 - restricted visibility in BON



Implementation

- command-line
- Eclipse plugin
 - popular programming platform
 - integrates different functionalities
 - quick and convenient use of tool
 - results presented in for user familiar fashion
- skeleton code generation
- internationalisation



So what?

Software modelling is an effective way of developing software.

A good modelling language is needed.

Tool support is needed.

This project extends current list of consistency checkers.



Demo

|*@

Questions?

|*@