

Wicket and Aida/Web

Brief Comparison

Jaroslav Havlín
ČVUT FEL
Subject X36ASS

Introduction

- The aim of this project is to design a simple reservation system for a small restaurant and to implement it twice, using two different architectural approaches.
- The result is a comparison of these approaches for this particular application.
- Chosen frameworks for this task are:
 - Apache Wicket for Java
 - Aida/Web for Smalltalk

Common Features

- Object-oriented programming languages
- Component-based web frameworks
- Integrated transparent AJAX support

Differences

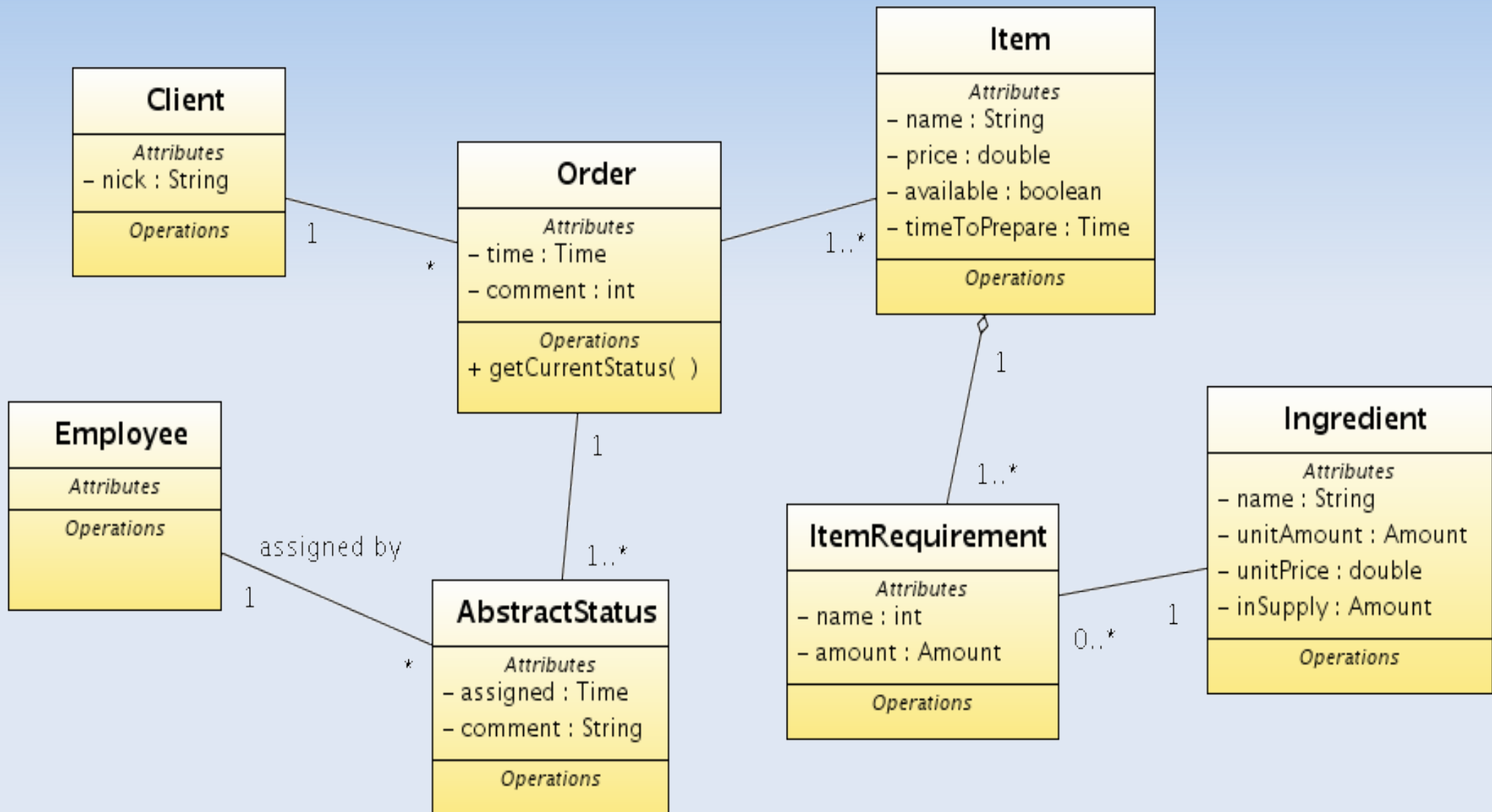
- Java + Apache Wicket
 - Statically typed language
 - HTML-based templates; CSS, scripts and images are stored in separate files



- Smalltalk + Aida
 - Dynamic Language
 - No templates, all is Smalltalk



Simplified model



User Interface

eJídlo

www.eJidlo.cz



Vítejte!

1. 12. 2008

Aktuální nabídka

- [Svíčková >>](#)
- [Hovězí pečeně >>](#)
- [Zelný salát >>](#)
- [Krupičná kaše >>](#)
- [Sekyrková polévka >>](#)

Zobrazit [komplentí seznam](#)

Hledání

[Rozšířené hledání](#)

Přihlášení

[Nový uživatel](#) | [Zapomenuté heslo](#)

Návod

[Jak objednávat](#)

[Reklamace](#)

[Vaše náměty](#)

Smalltalk

- Dynamic language
 - No spending time declaring types (+)
 - Great features for debugging (+)
 - No need to cast variables to correct types (+)
 - It is not necessary to wrap code into methods, there are blocks for anonymous pieces of code (+)
 - Some errors can't be found at compile-time (-)
- Specific user interface (+/-)
- Many dialects (-)

Aida

- Quite new framework
- No templates, only Smalltalk code – pure, saves time in many cases (+)
- Not completely finished documentation (-), but very kind community (++)
- Convention over Configuration paradigm (+)
 - Naming conventions for presentation objects, views and actions, no extra configuration needed.

Smalltalk + Aida (example)

```
viewMain
```

```
|e|
```

```
e := WebElement new.
```

```
e addTextH1: 'Registracni formular'.
```

```
e addTextH2: 'Aktualni nabidka'.
```

```
self observee items do: [ :item |  
    e add: ((WebElement newDiv)  
        addText: item name;  
        class: 'item'; yourself) ].
```

```
^ self pageFrameWith: e title: 'eJidlo'.
```

Java

- Statically-linked language, robust code
- Well-known, stable, established, well-documented (+, +, +, +)
- Sometimes quite "talky" (more code to write) (-)
- Excellent IDE support
 - Refactoring (+)
 - Static code analysis (+)
- More complicated syntax (-)



Wicket

- Web Framework based on templates
 - Natural division of labour for programmers and for designers (+)
 - It is possible to use standard tools for creating HTML templates (+)
- Template is special (X)HTML file with custom tags and properties
- Easy creation of new components, both simple and complicated, by nesting and inheritance
- Many free components and extensions available for free



Wicket Template (Example)

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html>
  <head>
    <title>Example</title>
  </head>
  <body style="margin:1em;">
    <form wicket:id="formComp">
      <div class="cellLabel" wicket:id="welcomeText"></div>
      <input type="text" wicket:id="name" />
      <input type="submit" wicket:id="submitButton" />
    </form>
  </body>
</html>
```

Wicket Class (Example)

```
public class HomePage extends WebPage {  
  
    String name = "";  
  
    public HomePage(final PageParameters parameters) {  
        Form f = new Form("formComp",  
                           new CompoundPropertyModel(this));  
        f.add(new Label("welcomeText", "This is an example"));  
        f.add(new TextField("name"));  
        f.add(new Button("submitButton"));  
        add(f);  
    }  
}
```

Comparison

■ Aida

- **Rapid development +**
- Readable code
- Code only
- Easy to extend
- **Dynamic platform +**
- Many dialects
- **More fun +**

■ Wicket

- Robust platform
- **Great documentation +**
- Templates
- **Many extensions +**
- Application server
- **Standardized +**
- Sophisticated syntax

Conclusion

- Aida is suitable for Smalltalk fans.
- It is so easy to become a Smalltalk fan :-)
- Java is suitable for (almost) all others.
- As there is a big variety of Java libraries for many problems (database drivers, image manipulation, artificial intelligence) and there is support for scalability, more complicated projects and projects that depend on legacy systems will probably choose a Java as a platform.
- For small and middle-sized projects, Aida promises faster development and more enjoyable programming.

Links

- havlij6@fel.cvut.cz
- <http://www.aidaweb.si/>
- <http://wicket.apache.org/>