

Metamorphic Testing with WS-BPEL

Student Short Talk Session - POPL'13

M. Azahara Camacho Magriñán



UCASE Software Engineering Research Group
Department of Computer Science
University of Cádiz

25th January 2013

UCASE



Website:

<http://www.uca.es/ucase>

- University of Cádiz (Spain)
- Members:
 - Director: Inmaculada Medina Bulo
 - Researchers: 9
 - Student Collaborators: 14
- Research lines:
 - Evolutionary mutation
 - Context-aware service adaption
 - Model-driving testing
 - Complex event procesing
 - Formal software verification
 - Dynamic invariant generation (with SPI&FM)

1 Basic concepts

2 Example: MetaSearch

3 Conclusions

1 Basic concepts

2 Example: MetaSearch

3 Conclusions

Metamorphic testing

Software testing technique that determines if a program has failed by the comparison between the actual output and the expected output. The main tools to apply this technique are **Metamorphic relations**.

Metamorphic testing

Software testing technique that determines if a program has failed by the comparison between the actual output and the expected output. The main tools to apply this technique are **Metamorphic relations**.

Metamorphic relations

Logic relations / properties that must be met between the inputs and outputs of different test cases that are applied with these test cases to check that are correct.

Web Services - Business Process Execution Language (WS-BPEL)

Language for...



Standardized by...

OASIS

Based language...

XML

Characterized by...

- Structured Programming
- XPath 1.0
- Event and fault handlers
- Concurrency

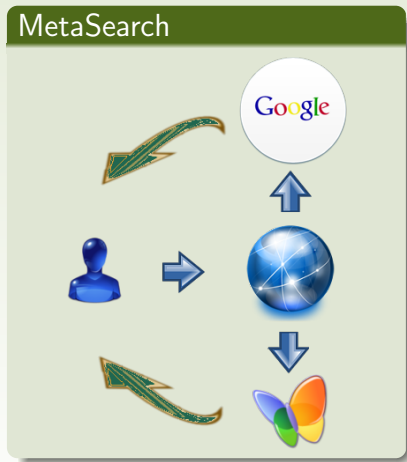
Example: MetaSearch

1 Basic concepts

2 Example: MetaSearch

3 Conclusions

Example: MetaSearch



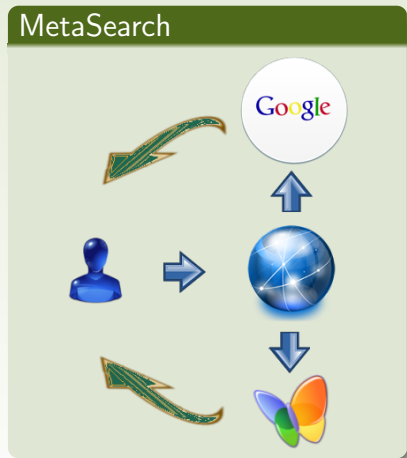
Information about the user:

- Language (Lan)
- Country (Cou)

Information about the search:

- Culture (Cul)

Example: MetaSearch



Information about the user:

- Language (Lan)
- Country (Cou)

Information about the search:

- Culture (Cul)

Relation

```
if !Lan or !Cou
    Cul = "en-US"
else
    Cul = concat(Lan-Cou)
```

Original fragment

```
<bpel:condition>  
  (($inputVariable.payload/  
    client:country != '' )  
    and  
    ($inputVariable.payload/  
      client:language != ''))  
</bpel:condition>
```

Implemented fragment

```
<bpel:condition>  
  (($inputVariable.payload/  
    client:country != '' )  
    or  
    ($inputVariable.payload/  
      client:language != ''))  
</bpel:condition>
```

Example: MetaSearch

Original fragment

```
<bpel:condition>
  (($inputVariable.payload/
    client:country != ''))
  and
  ($inputVariable.payload/
    client:language != '')
</bpel:condition>
```

Implemented fragment

```
<bpel:condition>
  (($inputVariable.payload/
    client:country != ''))
  or
  ($inputVariable.payload/
    client:language != '')
</bpel:condition>
```

Metamorphic relation

$$Cul1 \neq \text{"en - US"} \wedge Lan2 = \text{""} \Rightarrow Cul2 \neq Cul1$$

Conclusions

1 Basic concepts

2 Example: MetaSearch

3 Conclusions

Conclusions

- WS-BPEL is a tedious programming language, but represents very well web services.
- No software is completely correct.
- Metamorphic testing provides a new and innovative method to test software.
- Metamorphic relations requires a huge knowledge of the problem.
- This method allows us to improve correct software detecting new errors that they were not detected before.

Thank you for your attention



UCASE Software Engineering Research Group
Department of Computer Science
University of Cádiz