



Grupo de Investigación en
Reutilización y Orientación a
Objeto

Dynamism in Refactoring Construction and Evolution

A Solution Based on XML and Reflection



Authors:

Raúl Marticorena
Yania Crespo

rmartico@ubu.es
yania@infor.uva.es

ICSOFT
2008

3rd International Conference on Software and Data Technologies

5 - 8 July, 2008

Porto, Portugal

Outline



- Introduction
- MOON as Metamodel
- Running Refactorings
- Refactoring Construction and Evolution
- Current Work
- Conclusions and Future Work

Introduction



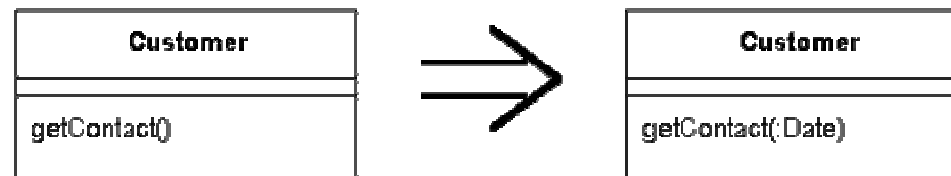
■ Introduction

- MOON as Metamodel
- Running Refactorings
- Refactoring Construction and Evolution
- Current Work
- Conclusions and Future Work

3 of 16

■ Refactoring [Fowler, 2000]

- *"Process of changing a software system in such a way that it does not alter external behavior of the code yet improve its internal structure"*
- *Example: Add Parameter* (275)



■ Open Research Trends

- Define new refactorings
- Identify code defects (*Bad Code Smells*)
- **Apply refactorings**
- **Tool support with certain language independence**
- etc.

Introduction



■ Introduction

- MOON as Metamodel
- Running Refactorings
- Refactoring Construction and Evolution
- Current Work
- Conclusions and Future Work

4 of 16

■ Language Representation

- Abstract Syntax Tree (AST)
- Database
- Logical predicates
- XML
- **Metamodel**
 - Language independence

■ Goals



Aided refactoring construction



Refactoring evolution with low effort



Running refactorings with certain language independence

- Reuse elements

MOON as Metamodel



- Introduction
- **MOON as Metamodel**
- Running Refactorings
- Refactoring Construction and Evolution
- Current Work
- Conclusions and Future Work

5 of 16

■ MOON [Crespo 2000]

■ Minimal Object-Oriented Notation

- abstractions for refactoring
- 50 classes

■ Storing:

- Classes
- Relationships
- Variants on the type system
- Entities
 - Concepts in source code with type
 - *self reference, super reference, local variable, method formal argument, class attribute and function result*
- Expressions
- Instructions
 - *creation, assignment, call and compound instructions*

MOON as Metamodel

■ Introduction

■ **MOON as Metamodel**

■ Running Refactorings

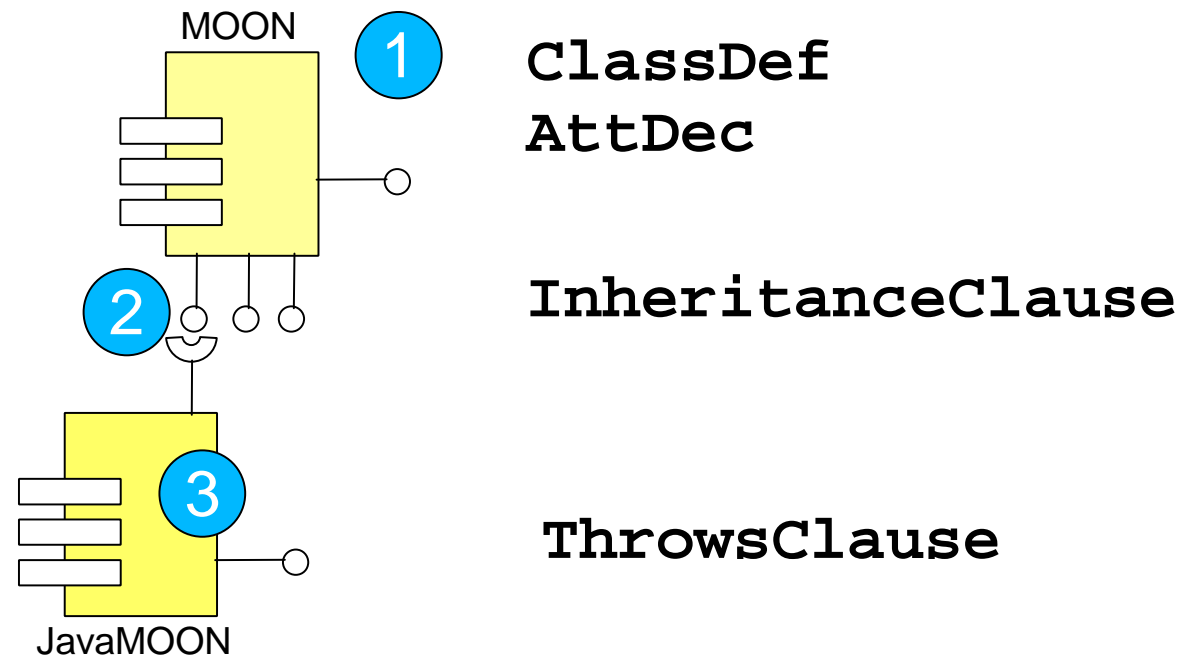
■ Refactoring Construction and Evolution

■ Current Work

■ Conclusions and Future Work

6 of 16

- ① General concepts: defined and implemented on MOON
- ② Extensible:
 - Defined on MOON
 - Implemented on concrete language (framework instantiation)
- ③ Particular: defined and implemented on a concrete language



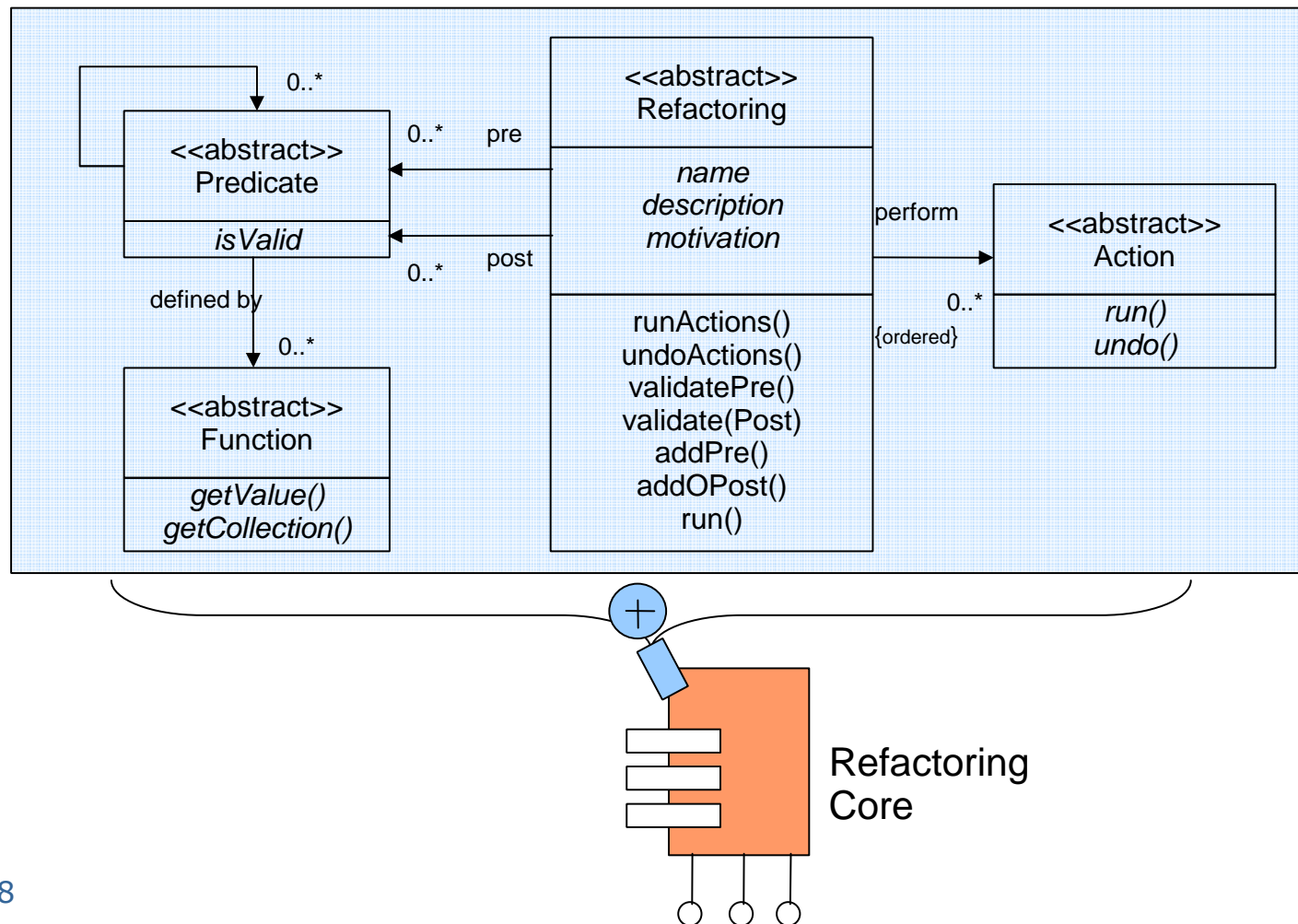
Running Refactorings

- Introduction
- MOON as Metamodel
- **Running Refactorings**
- Refactoring Construction and Evolution
- Current Work
- Conclusions and Future Work

7 of 16

■ Framework engine

■ Actions & Queries



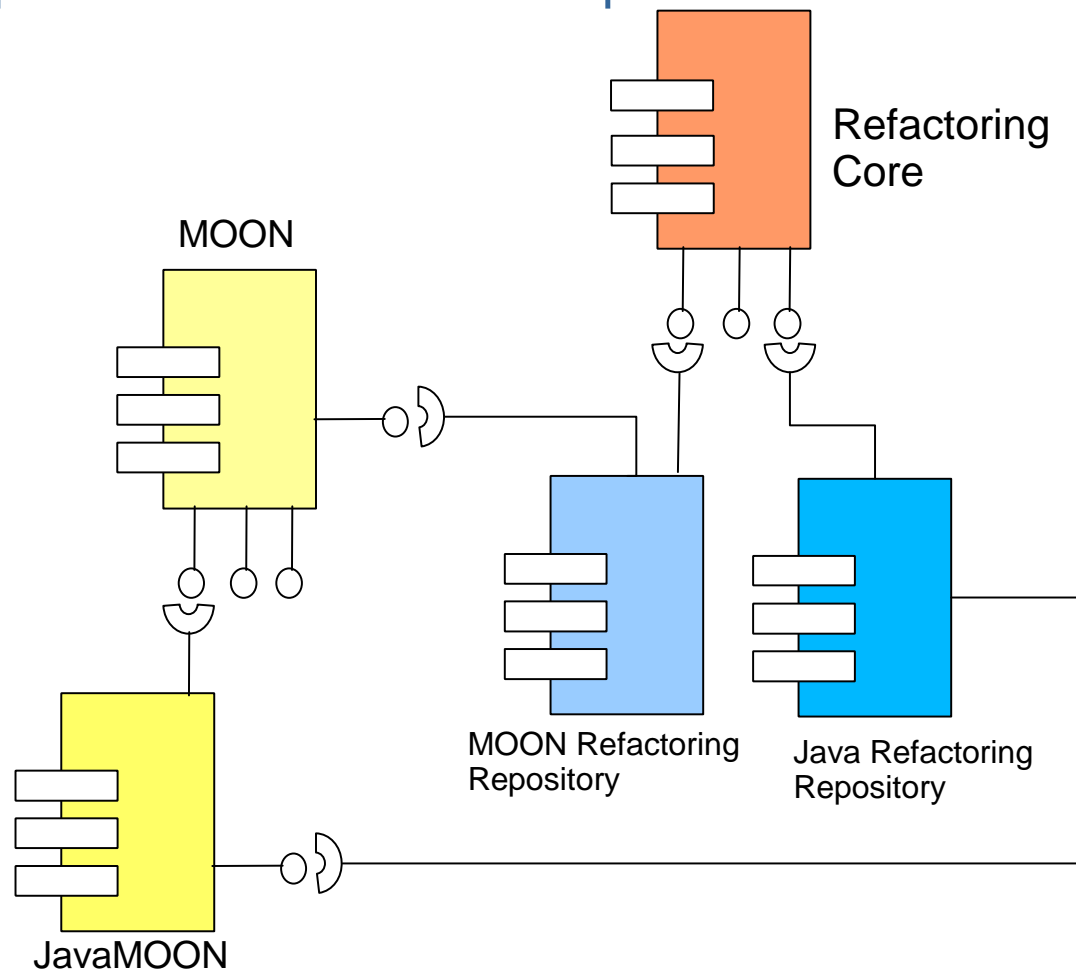
Running Refactorings

- Introduction
- MOON as Metamodel
- **Running Refactorings**
- Refactoring Construction and Evolution
- Current Work
- Conclusions and Future Work

8 of 16

■ Frameworks as solution

- Repositories with actions & queries

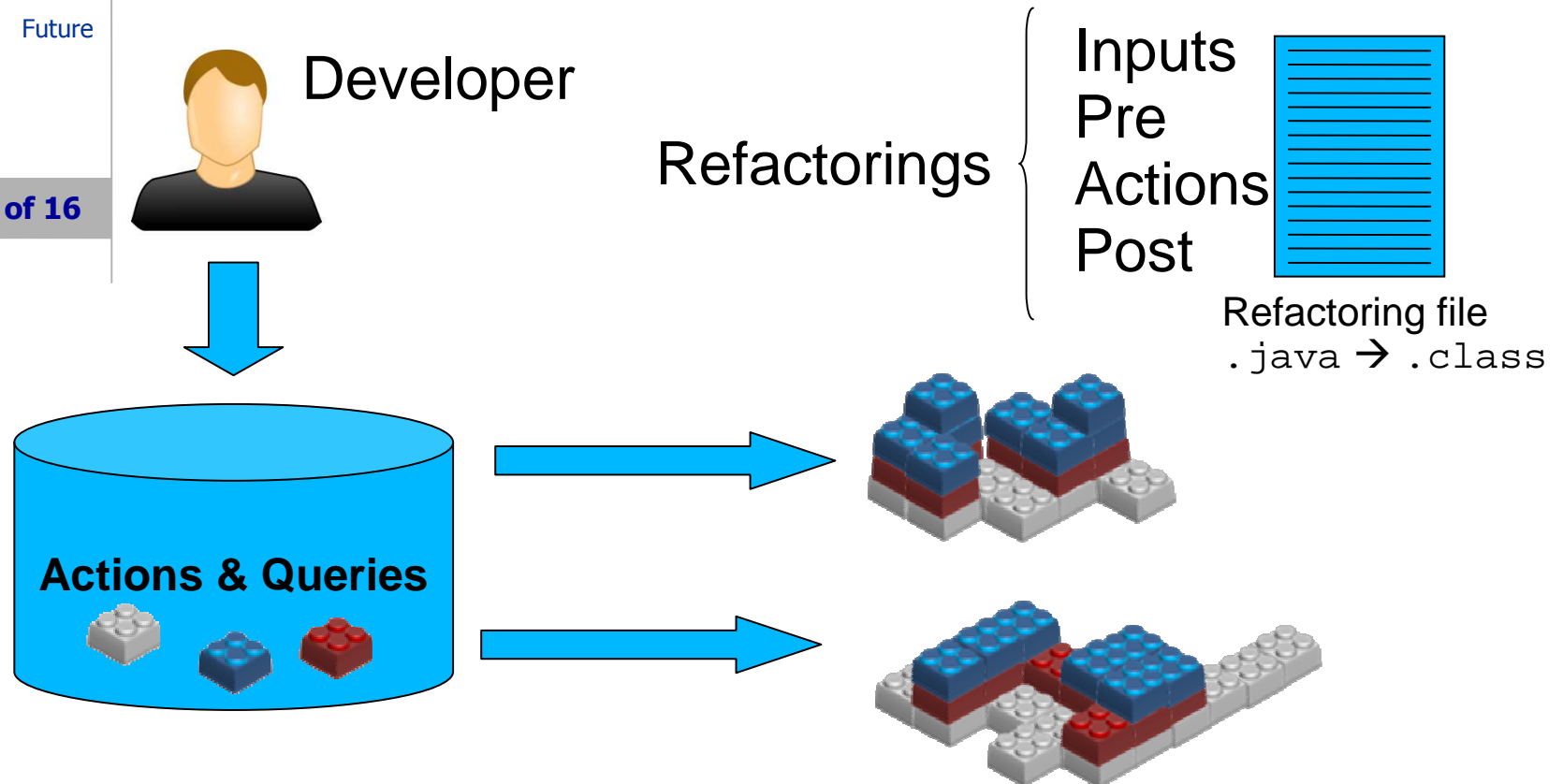


Refactoring Construction and Evolution

- Introduction
- MOON as Metamodel
- Running Refactorings
- **Refactoring Construction and Evolution**
- Current Work
- Conclusions and Future Work

9 of 16

- From hand coded refactorings ...
- To aided construction

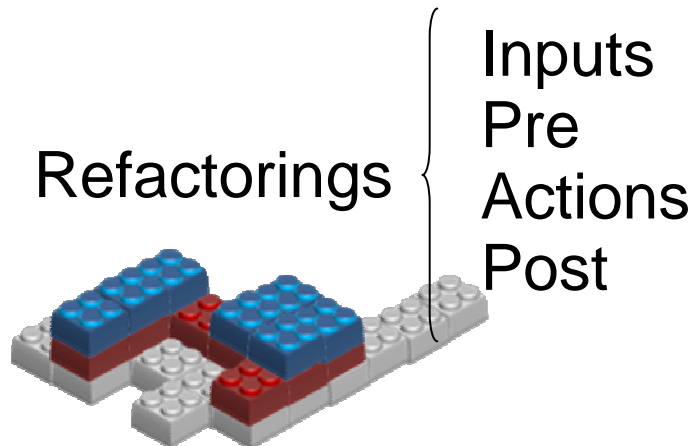


Refactoring Construction and Evolution

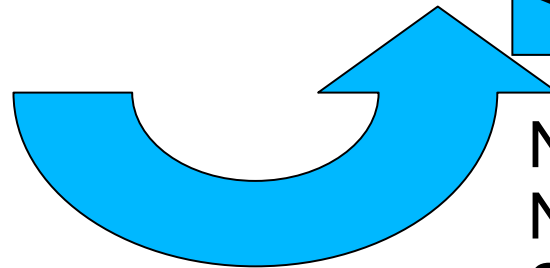
- Introduction
- MOON as Metamodel
- Running Refactorings
- **Refactoring Construction and Evolution**
- Current Work
- Conclusions and Future Work

10 of 16

■ Refactoring as XML file



```
<xml>  
  <refactoring>  
    <input>  
    </input>  
    ...  
  </refactoring>  
  
</xml>
```



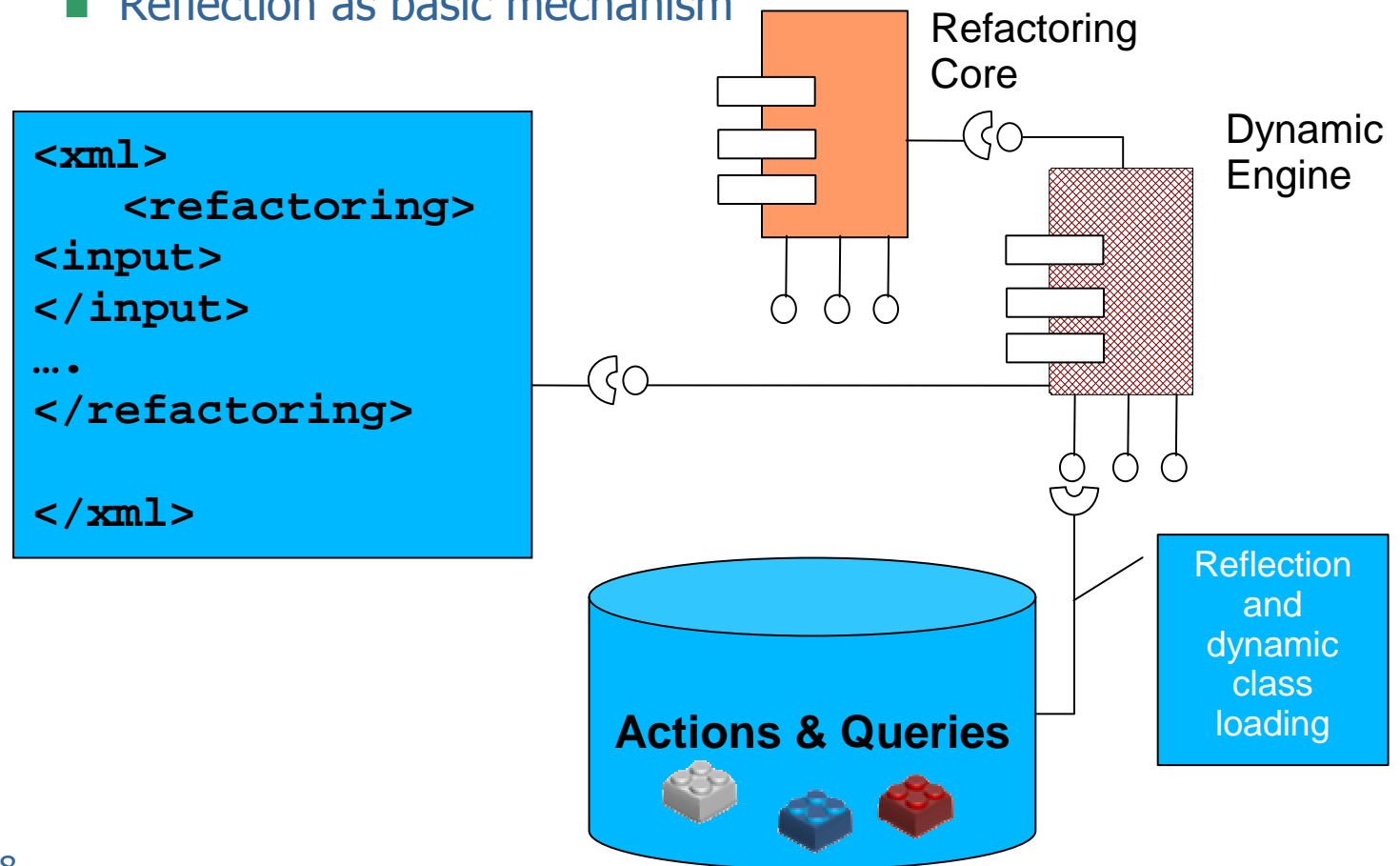
Not hand coded
Not compiled
Saved as XML

Refactoring Construction and Evolution

- Introduction
- MOON as Metamodel
- Running Refactorings
- **Refactoring Construction and Evolution**
- Current Work
- Conclusions and Future Work

11 of 16

- How can we run a refactoring from XML?
- Reusing the refactoring engine
 - Reflection as basic mechanism

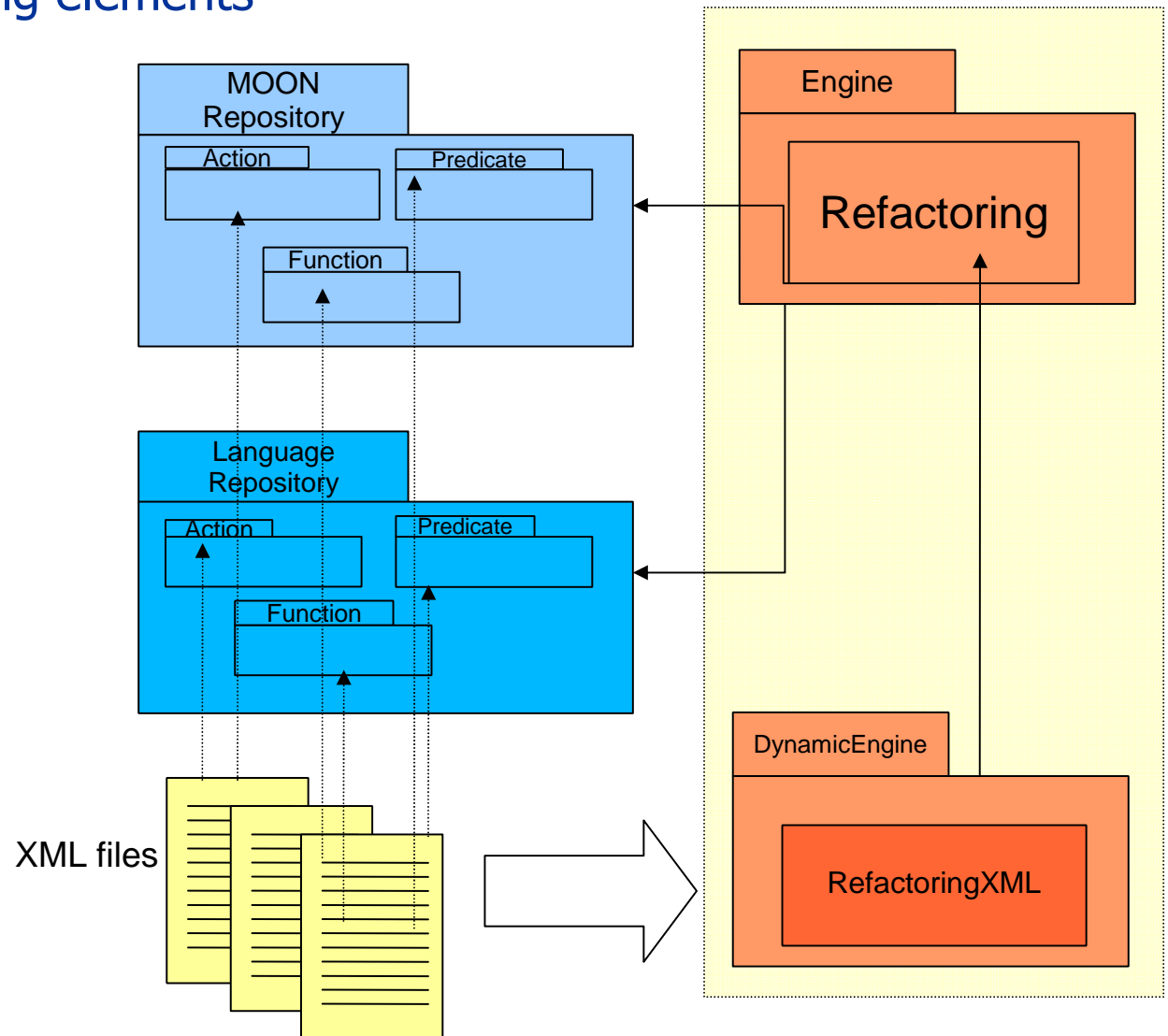


Refactoring Construction and Evolution

- Introduction
- MOON as Metamodel
- Running Refactorings
- **Refactoring Construction and Evolution**
- Current Work
- Conclusions and Future Work

12 of 16

■ Loading elements



Refactoring Construction and Evolution

- Introduction
- MOON as Metamodel
- Running Refactorings
- **Refactoring Construction and Evolution**
- Current Work
- Conclusions and Future Work

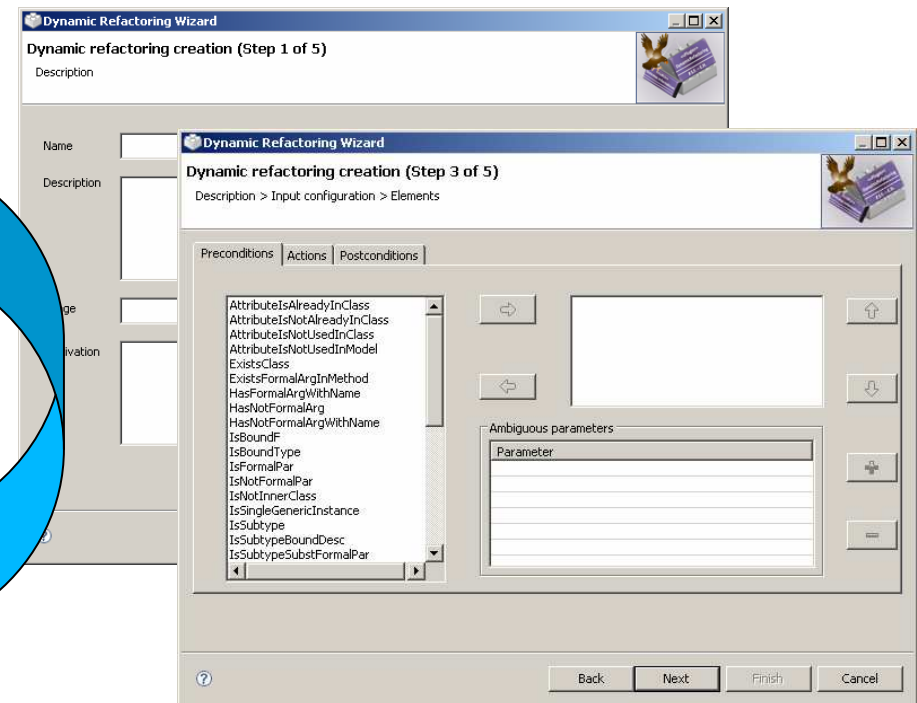
13 of 16

■ Problem

- Solution is prone to failures building the XML file manually
- Wizard to assist the construction
 - Using reflection
 - Showing inputs, pre/postconditions (queries) and actions

```
<xml>
<refactoring>
<input>
</input>
...
</refactoring>
</xml>
```

```
<xml>
<refactoring>
<input>
</input>
...
</refactoring>
</xml>
```



Current Work

- Introduction
- MOON as Metamodel
- Running Refactorings
- Refactoring Construction and Evolution
- **Current Work**
- Conclusions and Future Work

14 of 16

First prototype (static solution)

■ Hand coded refactorings

- Add Parameter (275)
- Rename Method (273)
- Rename Class
- Rename Parameter
- Move Field (146)
- Move Method (142)
- Remove Parameter
- Refactorings on generic features...
 - Specialize Bound S
 - Etc.

■ Second phase:

- XML solution with wizard
- Reflection mechanism as solution

■ Current phase:

- Improving XML format
- Developing an Eclipse plug-in
 - Partial language support
 - Currently working with example codes

Conclusions and Future Work



- Introduction
- MOON as Metamodel
- Running Refactorings
- Refactoring Construction and Evolution
- Current Work
- **Conclusions and Future Work**

15 of 16

- Users can construct and evolve refactorings
 - Not re-compilation
 - Aided modification using wizard
- Reuse refactorings
 - As composition of items / elements
- New refactorings to measure the reuse level
- Apply refactorings to migrate libraries with new features

Dynamism in Refactoring Construction and Evolution

A Solution Based on XML and Reflection

Thank you very much



Authors:

Raúl Marticorena
Yania Crespo

rmartico@ubu.es
yania@infor.uva.es

ICSOF
2008

3rd International Conference on Software and Data Technologies

5 - 8 July, 2008

Porto, Portugal