

Yet Another BPEL Extension for User Interactions

Mohamed Boukhebouze , Waldemar P. Ferreira Neto and Lim Erbin
PReCISE Research Center, University of Namur

WISM 2011-ER Workshop
November 3rd, 2011
Brussels, Belgium



Wallonie



Service public
de Wallonie

Avec le soutien de la DGO6
Département des Programmes
de Recherche

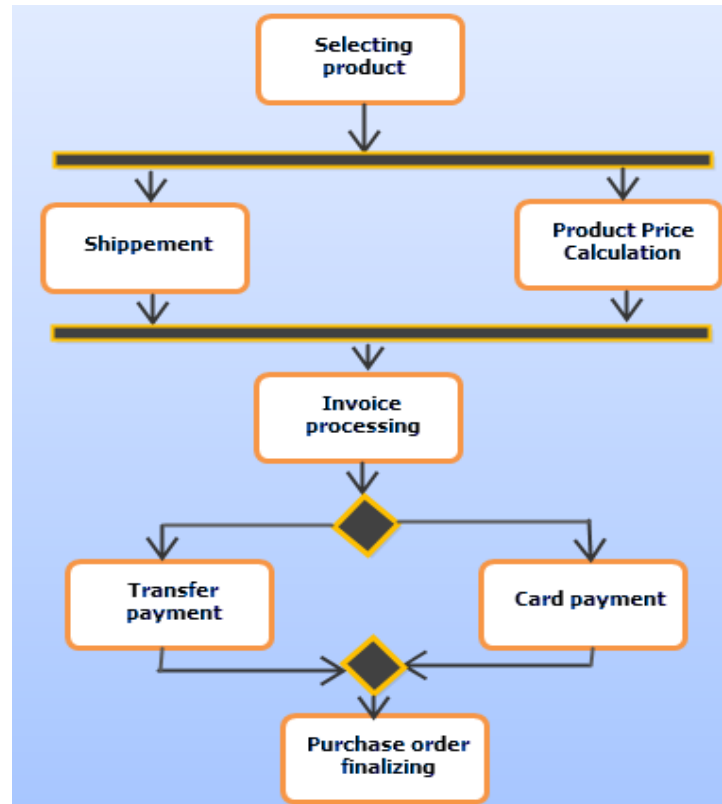


Agenda

- Introduction
 - Web service composition
 - BPEL standard
 - User interaction expression challenge
- UI-BPEL
 - UI-BPEL meta-model
 - UI-BPEL designer
- User interface generation process
- Related work
- Conclusion & future work

Introduction

- Web service composition
 - Orchestration or choreography of a set of Web services in order to implement a business process
 - Example: purchase order process scenario



Introduction

- Web service composition
 - Web service composition description languages
 - Defining the Web service composition aspects
 - Behaviour aspect (control flow)
 - Information aspect (data flow)
 - Several languages have been proposed
 - E.g. XLANG (Microsoft), WSFL (IBM) and BPEL (OASIS standard)
 - BPEL is the de facto standard for the description of the service composition execution

Introduction

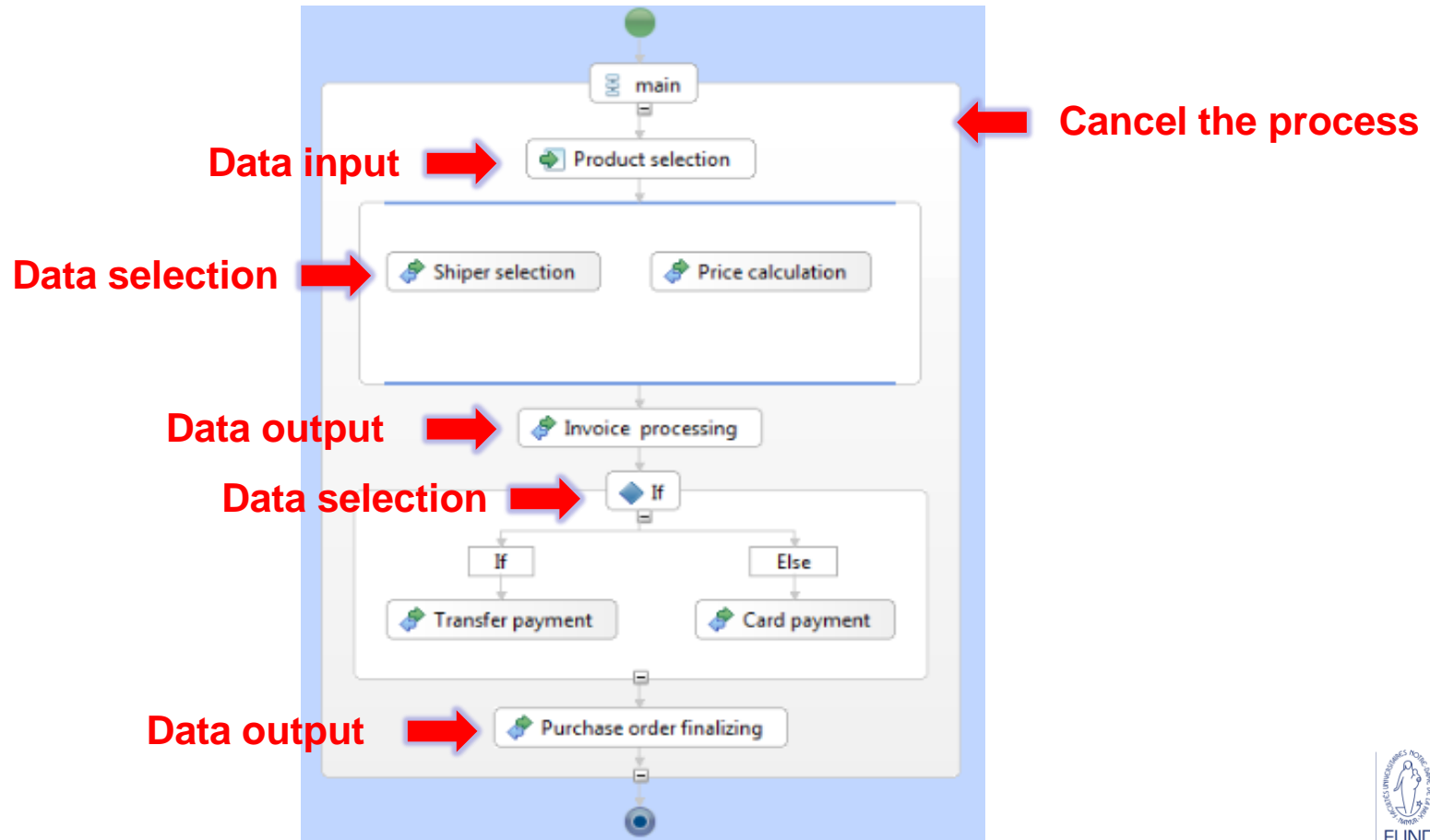
- BPEL standard
 - WS-BPEL Web Services Business Process Execution Language (BPEL for short)
 - BPEL is currently maintained by the OASIS
 - BPEL is an XML based language

```
<process>
  <partnerLinks> .. Process Partenaires.. </partnerLinks>
  <variables> ..Process Variable.. </variables>
  <faultHandlers> .. Activities to be executed when a fault occurs.. </faultHandlers>

  [Activities]* example:
  <invoke> .. Web service invocation..</invoke>
  <assign> ..variable value assignment.. </assign>
  <sequence> .. Activities order..</sequence>
  <if> ..condition expression..</if>
  <pick>.. .. Activities to be executed when a specific event occurs .. </pick>
</process>
```

Introduction

- User interaction expression challenge
 - BPEL is not able to express the user interactions at runtime
 - Illustration: purchase order process scenario



Introduction

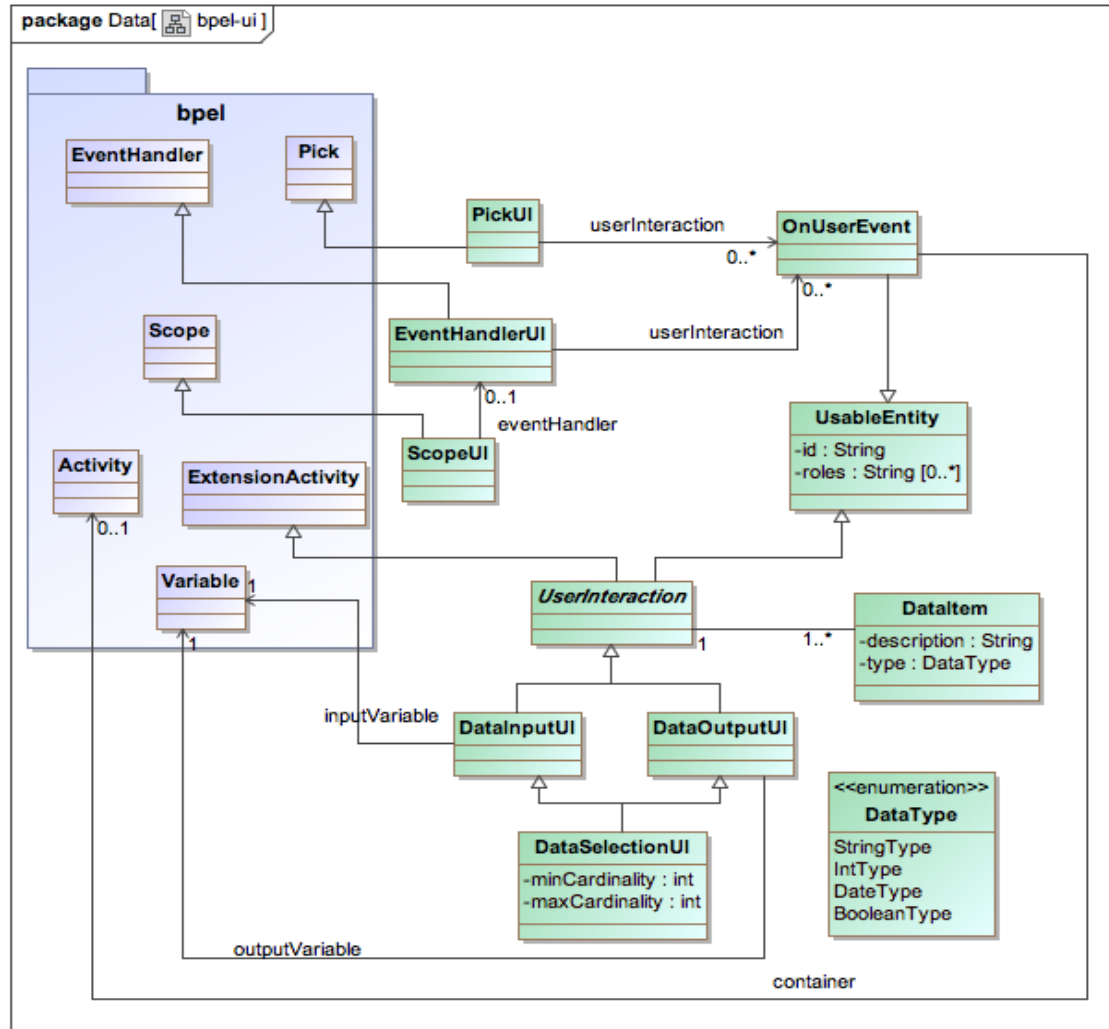
- User interaction expression challenge
 - BPEL meta-model should be extended to support the user interaction types in runtime
 - Data input interaction
 - Data output interaction
 - Data selection interaction
 - User event interaction
 - User interface should be generated for the Web service composition
 - The user interface should be adapted to the user context

Our proposition

- UI-BPEL meta-model
 - Extension of the BPEL meta-model
 - Supporting the user interaction expression
- UI-BPEL designer
 - UI-BPEL support tool
- Generation of a context-aware user interface for the interactive composition

UI-BPEL

- UI-BPEL meta-model

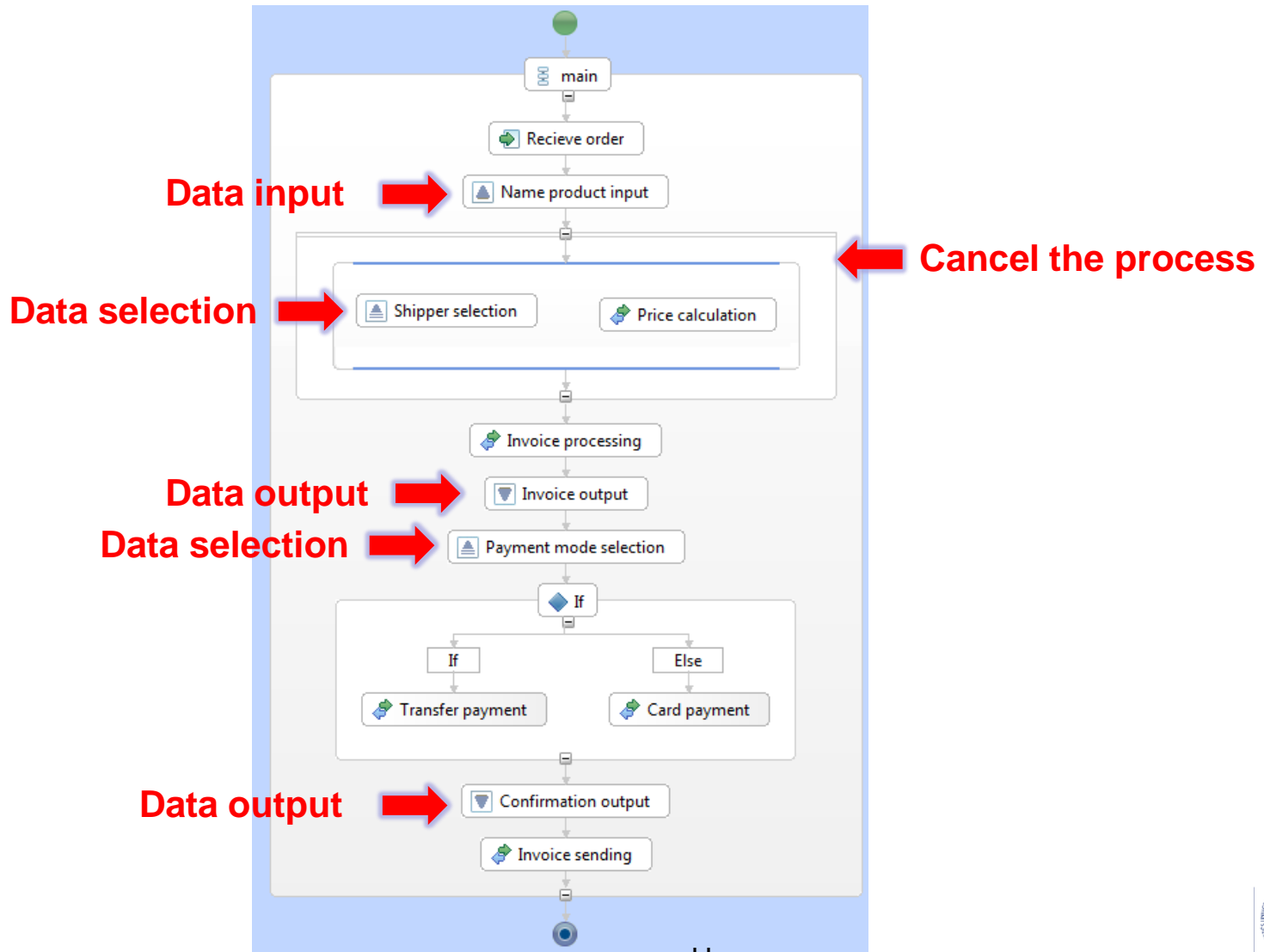


UI-BPEL

- UI-BPEL meta-model
 - New BPEL activities
 - DataInputUI
 - DataOutputUI
 - DataSelectionUI
 - New type of BPEL event
 - OnUserEvent
 - New BPEL attributes
 - UserRole

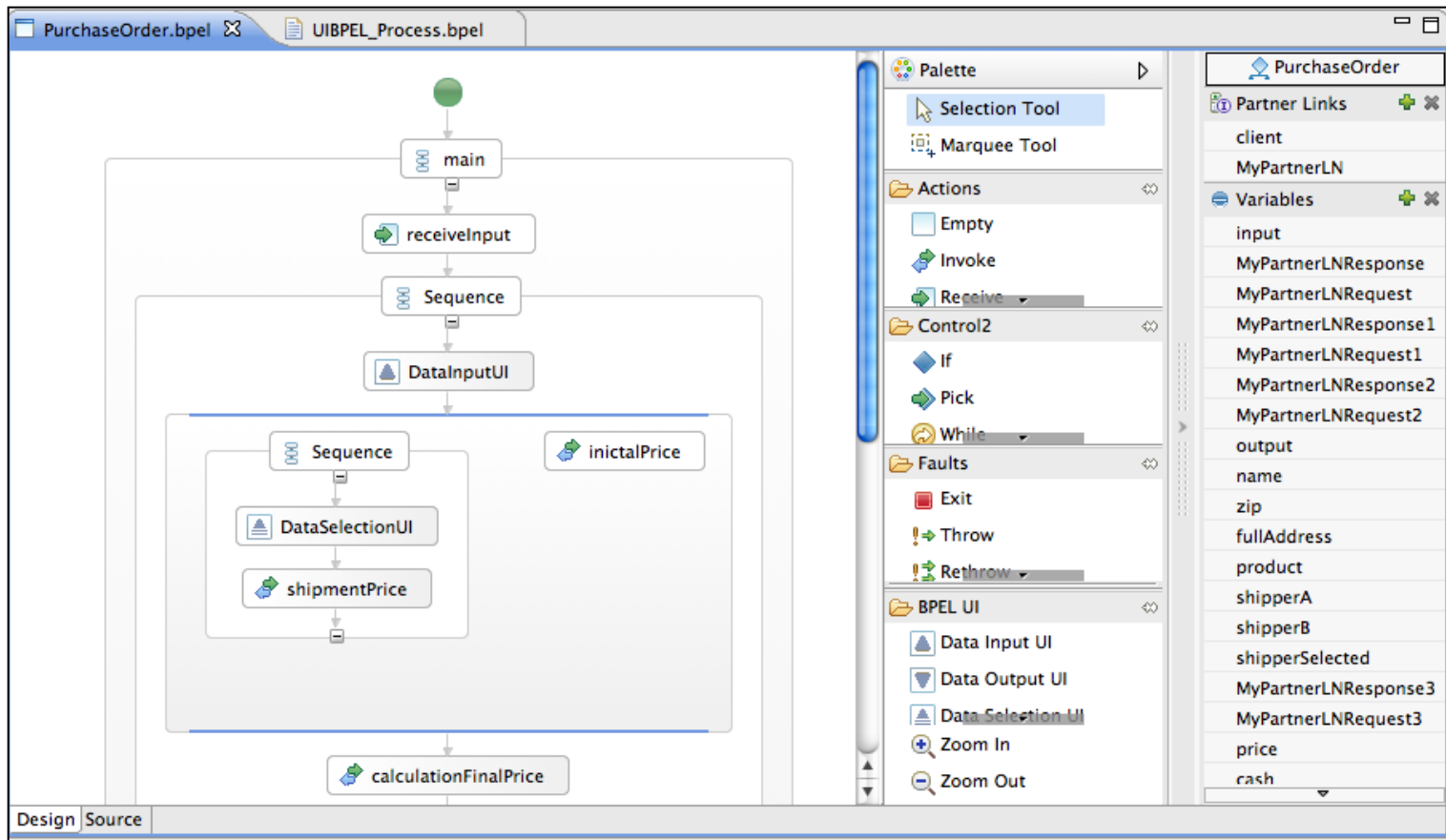
Contribution

- UI-BPEL meta-model: illustration



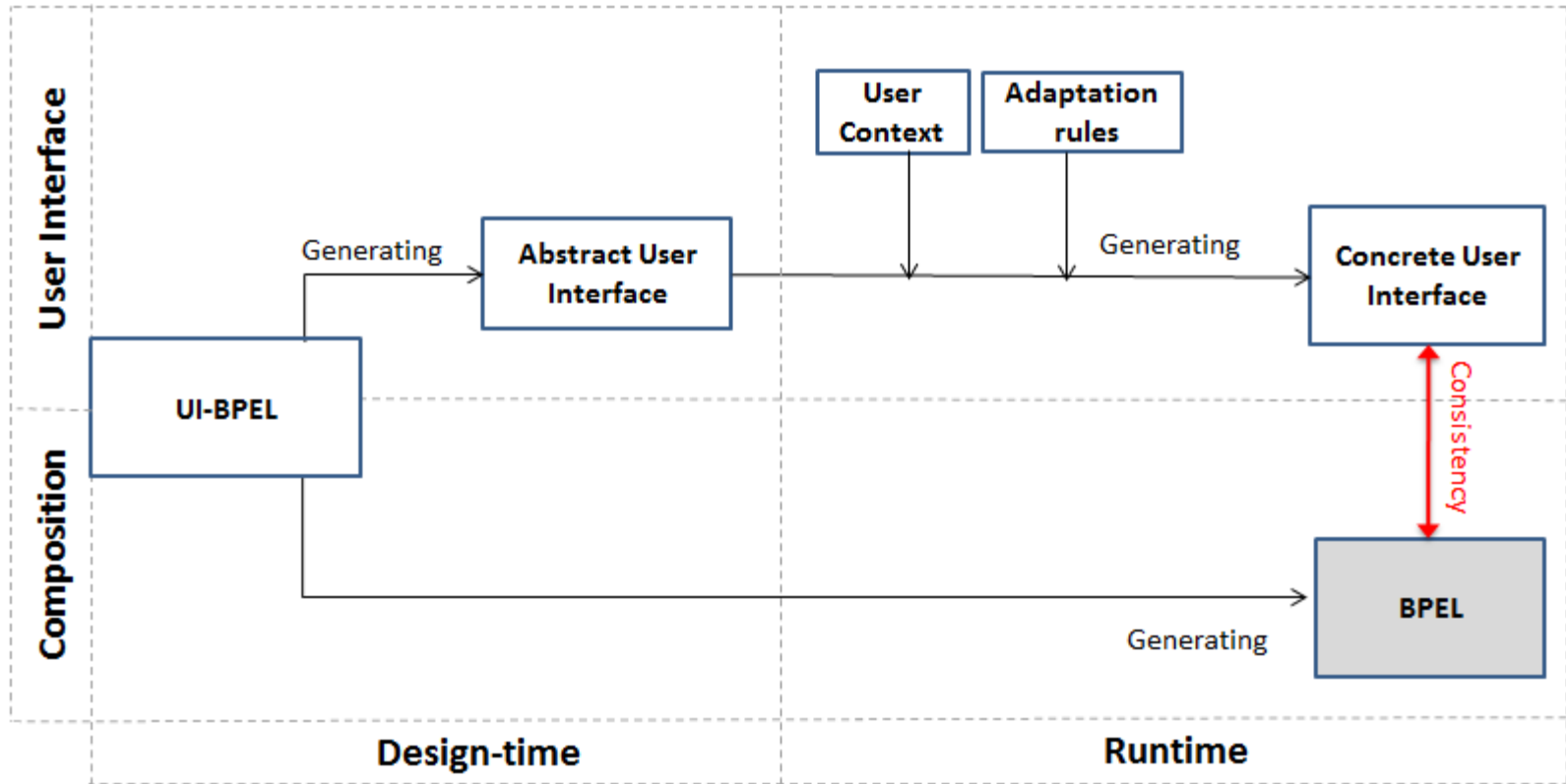
UI-BPEL

- UI-BPEL Designer
 - The designer is an Eclipse plug-in
 - The designer is built based on the Eclipse BPEL Designer



User interface generation process

- User interface generation process overview



Related Work

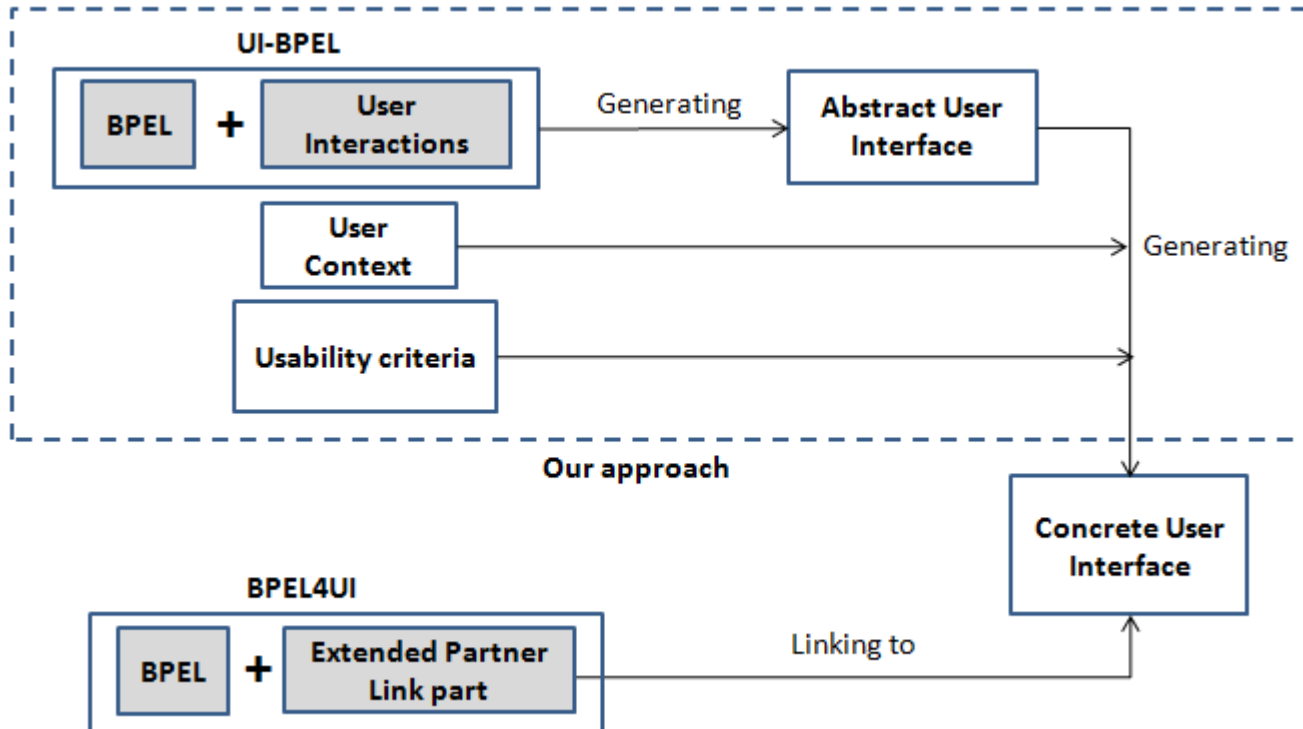
- BPEL extensions for user interaction expression
 - BPEL4People (IBM and SAP, 2005)
 - Introducing a new type of BPEL activities for the specification of human tasks
 - Pro:
 - Expressing the data interactions
 - Con:
 - Do not deal with the generation of a user interface for the Web service composition
 - Do not express the event interaction

Related Work

- BPEL extensions for user interaction expression
 - BPEL4UI (Daniel et al., BPM 2010)
 - Introducing new the *Partner Link part Type*
 - Pro:
 - Defining a binding between BPEL activities and an existing user interface
 - Con:
 - This user interface is developed separately from the composition
 - Do not allow the generation of a user interface adapted to the user context

Related Work

- User interface generation process
 - Our approach VS. BPEL4UI



Conclusion & future work

- **Conclusion**

- Extending BPEL meta-model to support user interaction
- Generating a context-aware user interface for Web service composition

- **Future work**

- Transformation of UI-BPEL to an abstract user interface and a concrete user interface
- Managing the the consistency between the control flow of the composition and the navigation flow of its user interface

Thank you for your attention

Questions ?

Appendix

- BPEL meta-model

