

A Process to Integrate Transformation Templates in the Model-Driven Engineering of User Interfaces

Nathalie Aquino^{1,2}, Luca Cernuzzi¹, Oscar Pastor²

¹Departamento de Electrónica e Informática, Universidad Católica "Nuestra Señora de la Asunción"
Asunción-Paraguay

{nathalie.aquino, lcernuzz}@uca.edu.py

²Centro de Investigación en Métodos de Producción de Software, Universitat Politècnica de València
Valencia-España

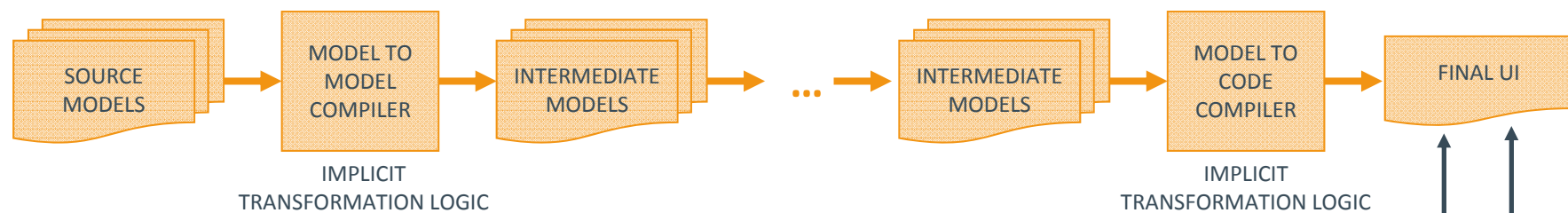
{naquino, opastor}@pros.upv.es

Motivation

- In the context of Model-Driven Engineering (MDE), interaction modeling is still an open issue
- There are several approaches for MDE of user interfaces (UIs)
 - They are not widely known or used
 - Their related processes are not always reusable

Open issues in MDE of UIs

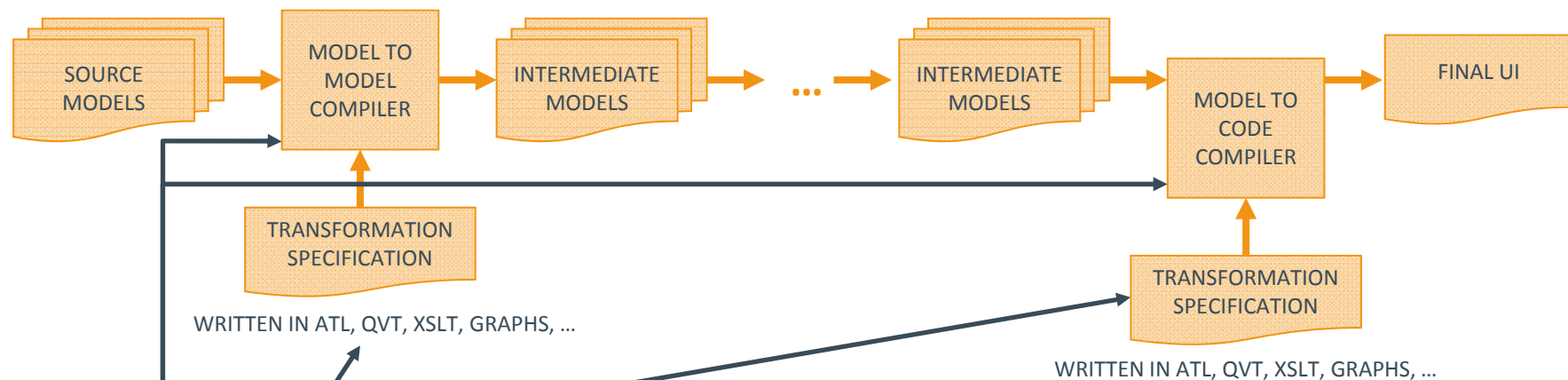
- Approaches for MDE of UIs
 - *Implicit transformation logic*



- **CONCERN 1:** Implicit design knowledge and presentation guidelines
- **CONCERN 2:** End-users always perceive similar UIs
- **CONCERN 3:** UI designers must manually modify the generated UI code

Open issues in MDE of UIs

- Approaches for MDE of UIs
 - *Explicit transformation logic*



- **CONCERN 4:** UI designers face a complex process for specifying transformations
- **CONCERN 5:** Difficulties for gradually extend the capabilities of tools for MDE of UIs

Goals and scope

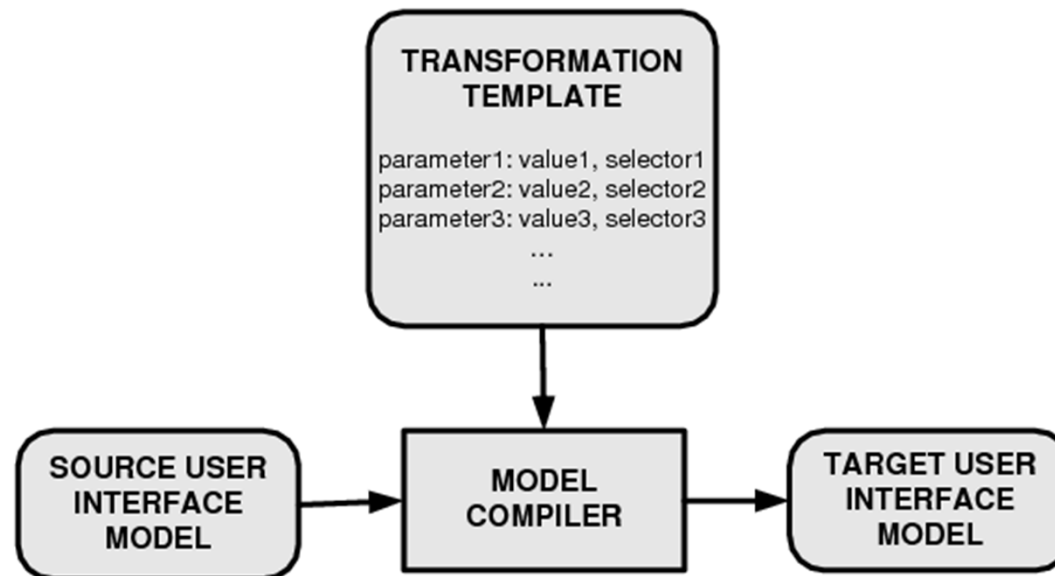
- How to deal with the previous concerns?
 - Transformation Templates
- Transformation Templates can be integrated with different approaches for MDE of UIs
- Focus of this presentation
 - Present guidance on how to proceed with the integration process

Agenda

- Transformation Templates
- Integrating Transformation Templates in approaches for MDE of UIs
 - Adopt a Parameter Type in an MDE approach
 - Specify a Parameter Type for an MDE approach
- Applying the process to adopt a Parameter Type in OO-Method
- Conclusion

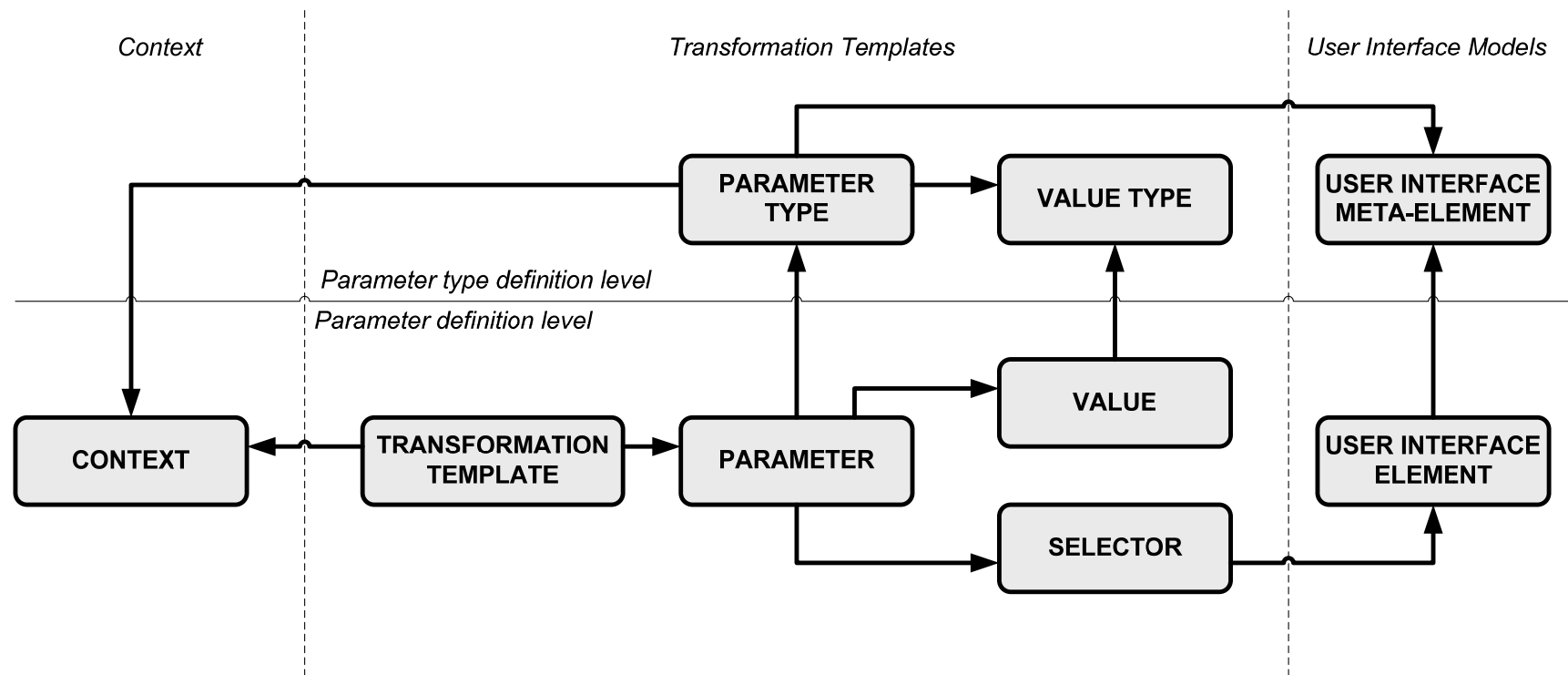
Transformation Templates

- **Transformation Templates**
 - Specify the structure, layout, and style of a UI
 - Are composed of **parameters** with associated values that parameterize UI model transformations
 - Are inputs for transformation tools


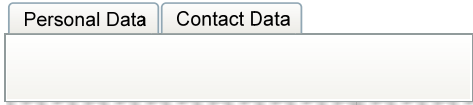
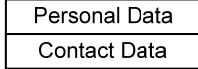


Transformation Templates

- Main concepts of the Transformation Templates approach



Transformation Templates

Parameter Type				
Name	Affects	Possible values enumeration		
		Value	Graphical description	
Grouping layout	Service Interaction Unit, Argument Grouping (elements of the OO-Method Presentation Model – Pastor 2007)	group box		
		tabbed dialog box		
		accordion		
Contexts				
		<i>Desktop – C# on .NET</i>		<i>Web – ASP.NET</i>
Possible value	Importance level	Development cost	Importance level	Development cost
group box	high	low	high	low
tabbed dialog box	high	low	high	low
accordion	low	high	high	medium
Possible value		Usability guidelines (for desktop context)		
group box		Visual distinctiveness is important. The total number of groups will be small.		
tabbed dialog box		Visual distinctiveness is important. The total number of groups is not greater than 10.		
accordion		Visual distinctiveness is important. The total number of groups is not greater than 10.		

Transformation Templates

- Transformation Templates gather Parameters

<i>Parameter</i>	<i>Value</i>	<i>Selector</i>	<i>Comments</i>
Grouping layout	accordion	id=5	The element with id=15 will be rendered with an accordion
Grouping layout	group box	Argument Grouping	All Argument Groupings will be rendered with group boxes
Grouping layout	tabbed dialog box	Argument Grouping directly contained in Service Interaction Unit	In all Service Interaction Units, all the Argument Groupings which are in the first level of containment, will be rendered with tabbed dialog boxes

Transformation Templates

- Concerns about MDE of UIs vs. Transformation Templates

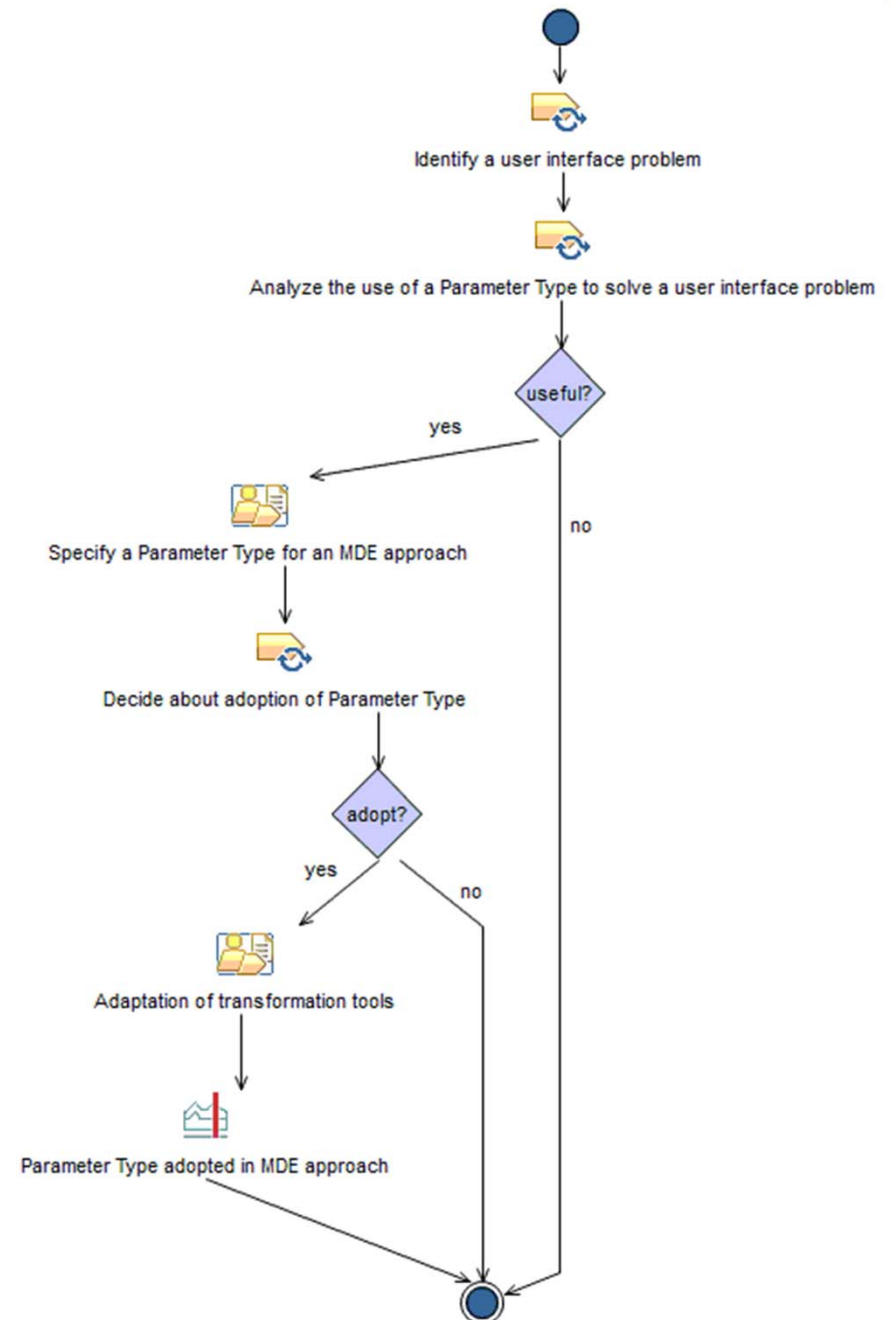
<i>Concern</i>	<i>Transformation Templates - Flexibility</i>
CONCERN 1: Similar UIs	A catalogue of Parameter Types provides <i>many design options</i> which leads to <i>many different UIs</i>
CONCERN 2: Implicit transformation logic	Transformation Templates externalize the transformation logic making it <i>editable, customizable, and reusable</i>
CONCERN 3: Manual modifications on generated UI code	UI designers use an <i>easier process</i> which consists on selecting parameters, values, and selectors
CONCERN 4: Complex process for modifying external transformation logic	
CONCERN 5: Difficulties for gradually extending the capabilities of tools	Estimations of importance level and development cost for different contexts of use allow to decide a <i>priority of implementation</i> for parameter types and values

Integrating Transformation Templates in approaches for MDE of UIs

- Transformation Templates can be integrated with different approaches for MDE of UIs
 - In order to guide these integrations we propose two processes
 - Adopt a Parameter Type in an MDE approach
 - Specify a Parameter Type for an MDE approach
 - Processes were defined according to the Software and Systems Process Engineering Meta-Model (SPEM) and using the Eclipse Process Framework Composer (EPF Composer)

Adopt a Parameter Type in an MDE approach

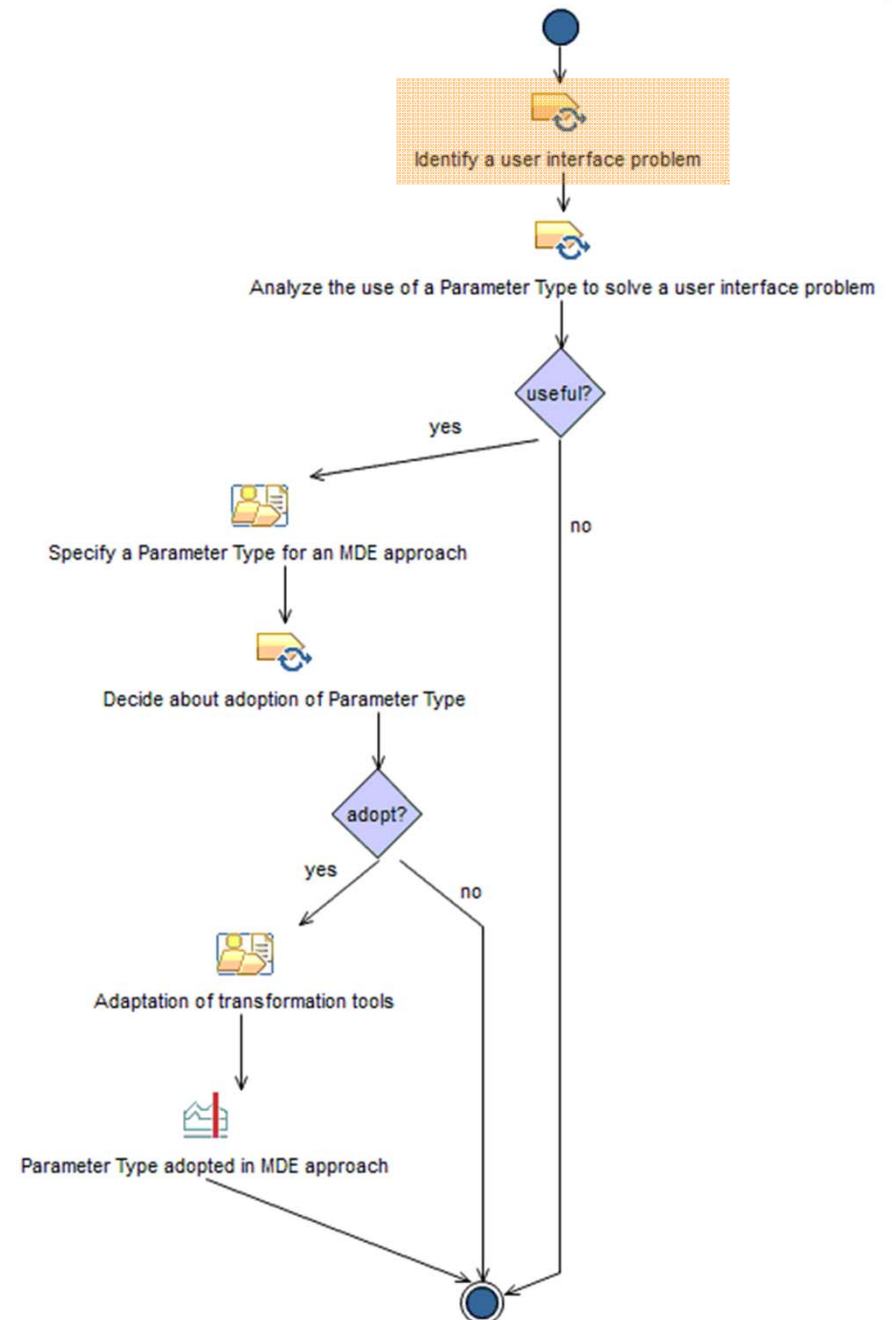
- Considered case:
 - Parameter Types are adopted incrementally in a selected MDE approach in order to cope with problems of the UIs generated by the MDE approach



Adopt a Parameter Type in an MDE approach

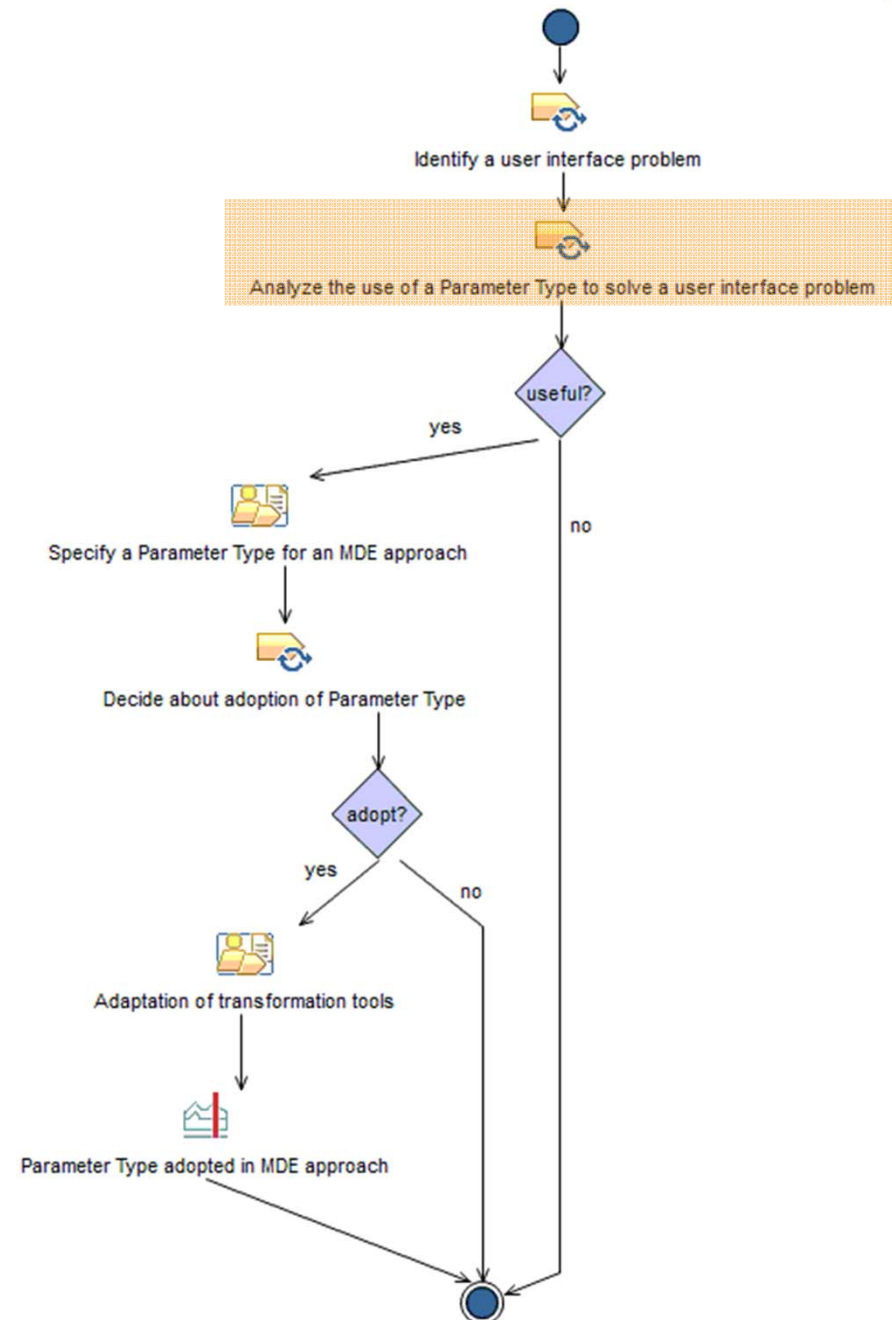
- **Task:** Identify a UI problem
- **Performing roles:** end-users, traditional or MDE UI designers, usability experts, software developers, project leaders
- **Output:** Report of a UI problem
- **Guidance material:** Template to report UI problems

- **Important information**
 - Number of times that the same UI problem is reported
 - Description of the context in which the problem appears (software system, computing platform, etc.)



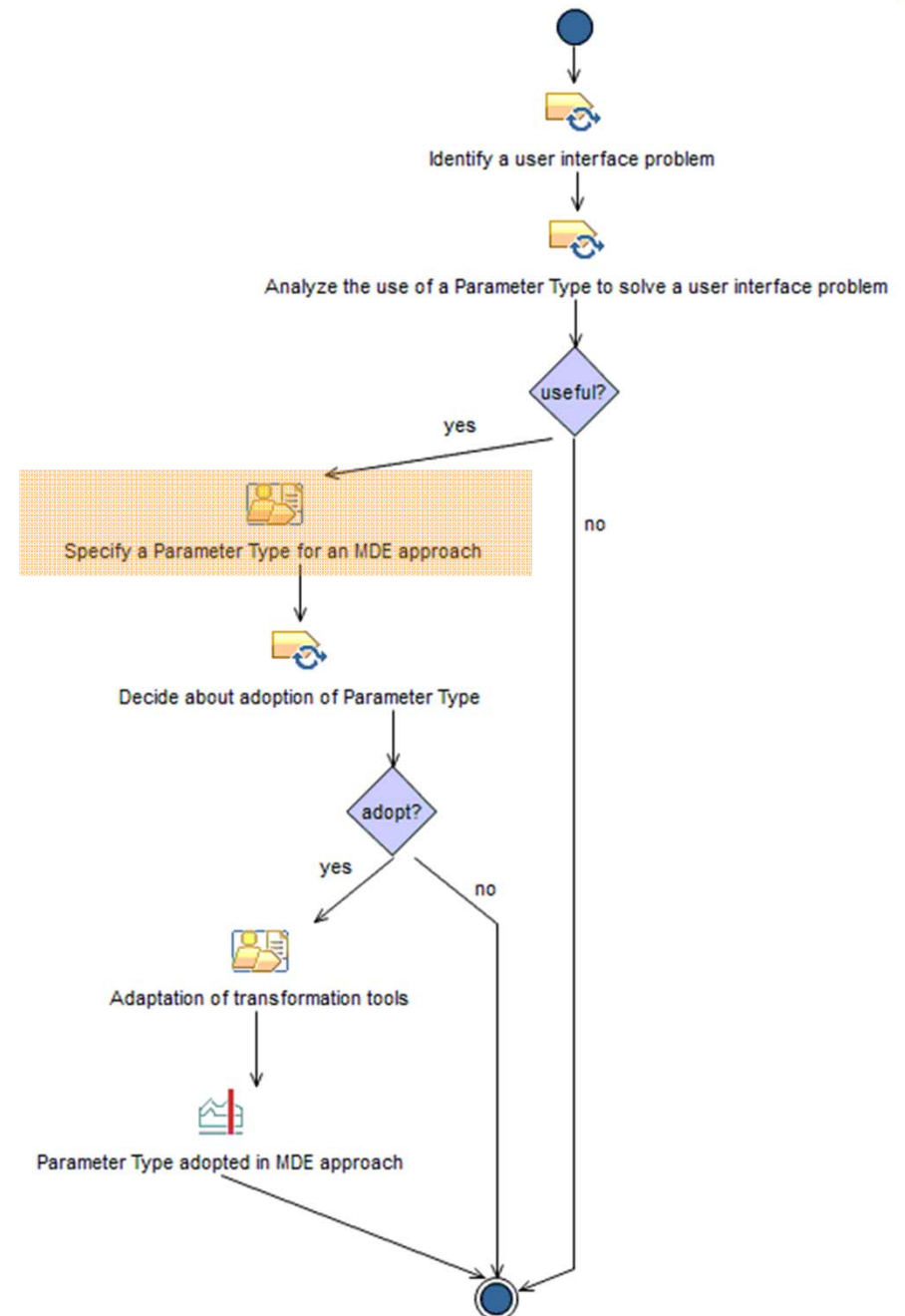
Adopt a Parameter Type in an MDE approach

- **Task:** Analyze the use of a Parameter Type to solve a UI problem
- **Input:** Report of a UI problem
- **Performing roles:** MDE UI expert, project leader
- **Questions to consider**
 - Is the problem related to
 - A customization of the structure, layout, or style of a UI?
 - Data displayed in a UI?
 - Can the problem be faced offering customization options?
 - Has the problem been reported more than once? Many times?
 - How much difficult is to solve the problem without using parameter types?
 - Manual tweaking?
 - Modification of transformation rules?

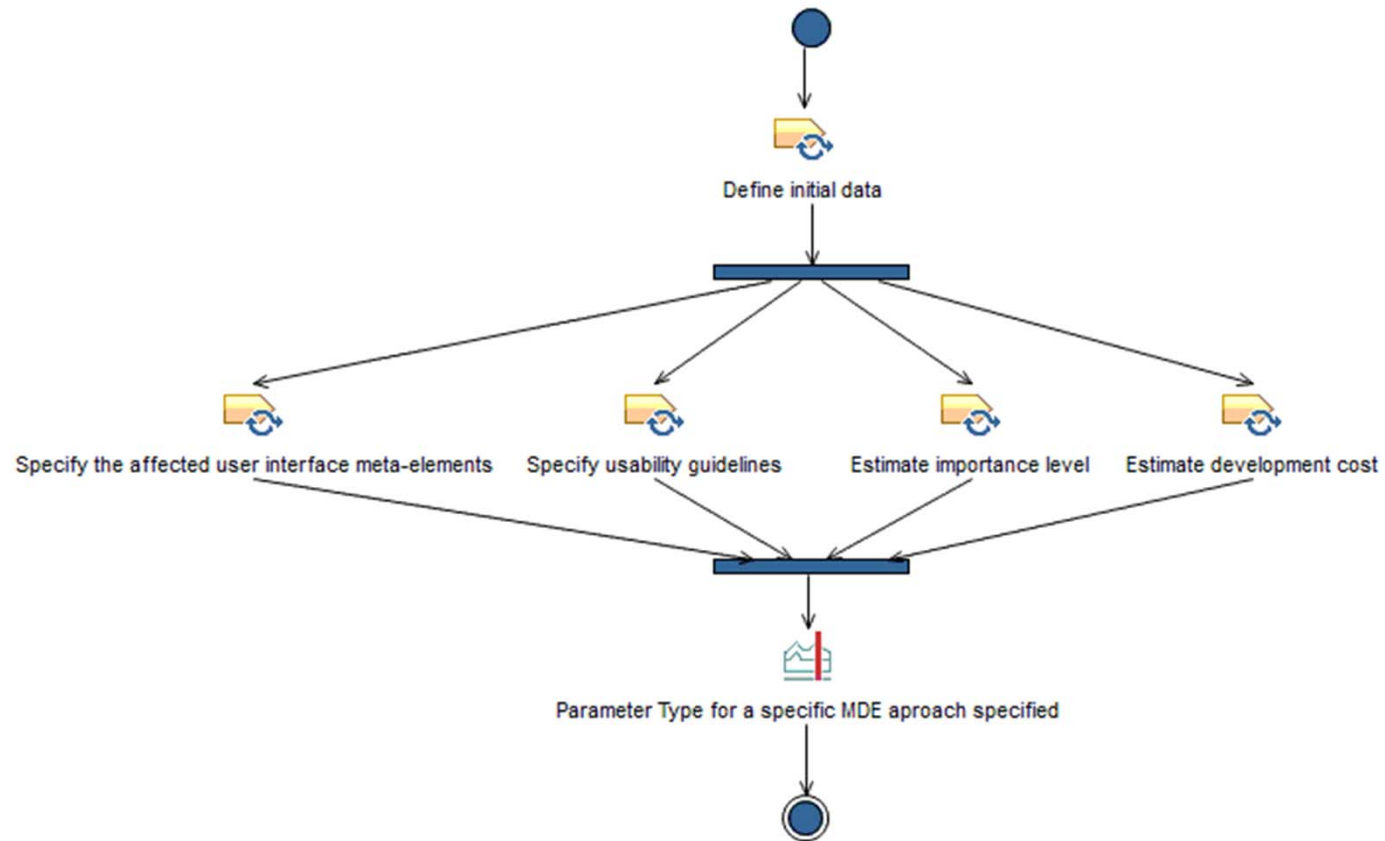


Adopt a Parameter Type in an MDE approach

- **Sub-process**
 - Specify a Parameter Type for an MDE approach

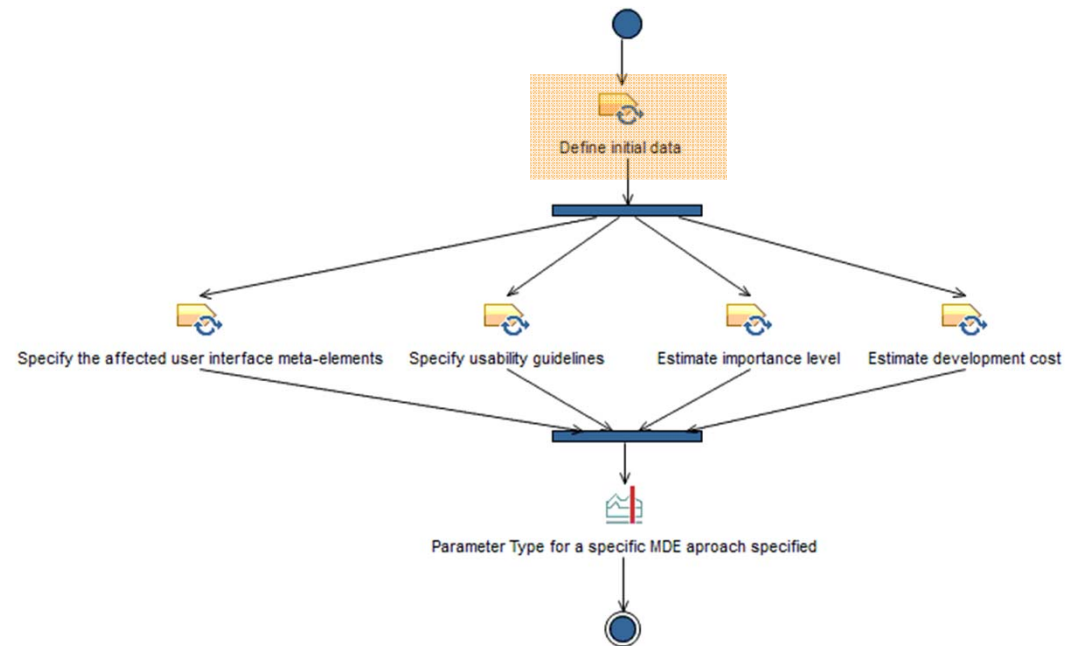


Specify a Parameter Type for an MDE approach



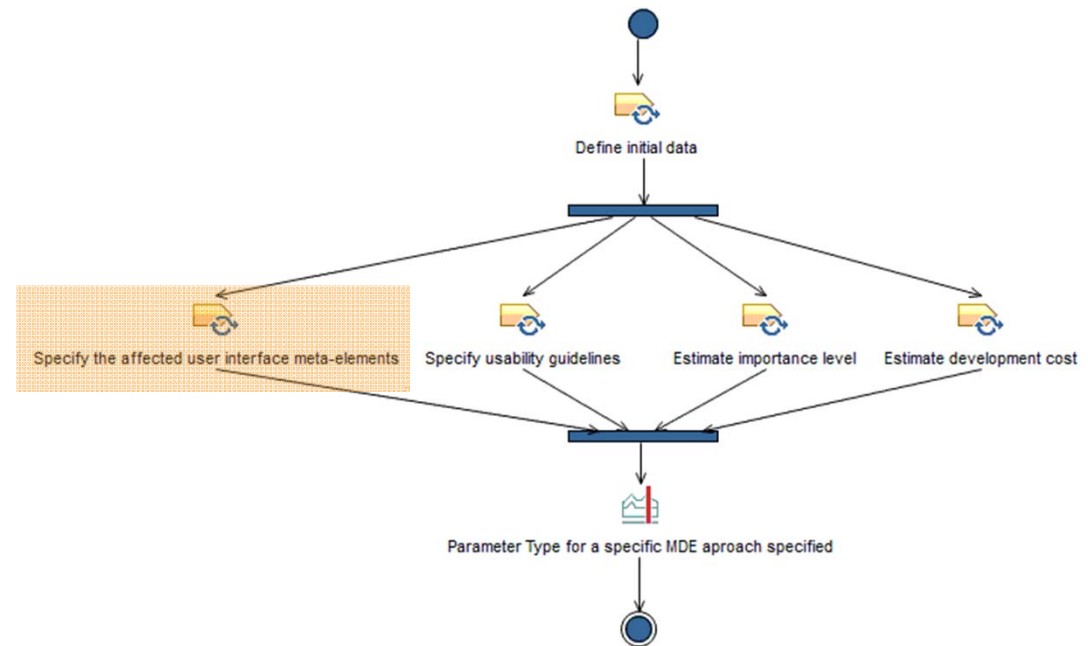
Specify a Parameter Type for an MDE approach

- **Task:** Define initial data
- **Input:** Report of a UI problem
- **Performing roles:** traditional or MDE UI designers
- **Output:** Parameter Type specification
- **Guidance material:** examples of Parameter Types, template to specify Parameter Types, tool mentor of the editor of Transformation Templates
- **Data to be included**
 - Name
 - Description
 - Contexts of use
 - Possible values



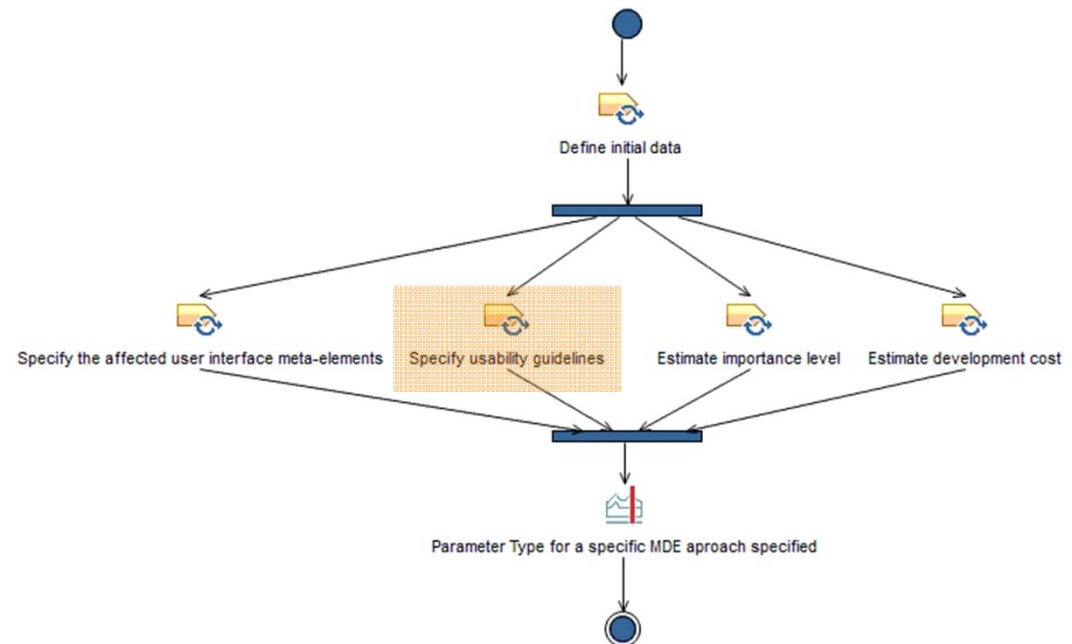
Specify a Parameter Type for an MDE approach

- **Task:** Specify the affected UI meta-elements
- **Inputs:** Report of a UI problem, Parameter Type specification
- **Performing roles:** MDE UI expert
- **Output:** Parameter Type specification (updated)
- Saying that a parameter type affects a UI meta-element means that parameters of the parameter type in question will be able to be applied and to have an effect on the UI elements related to the UI meta-element in question



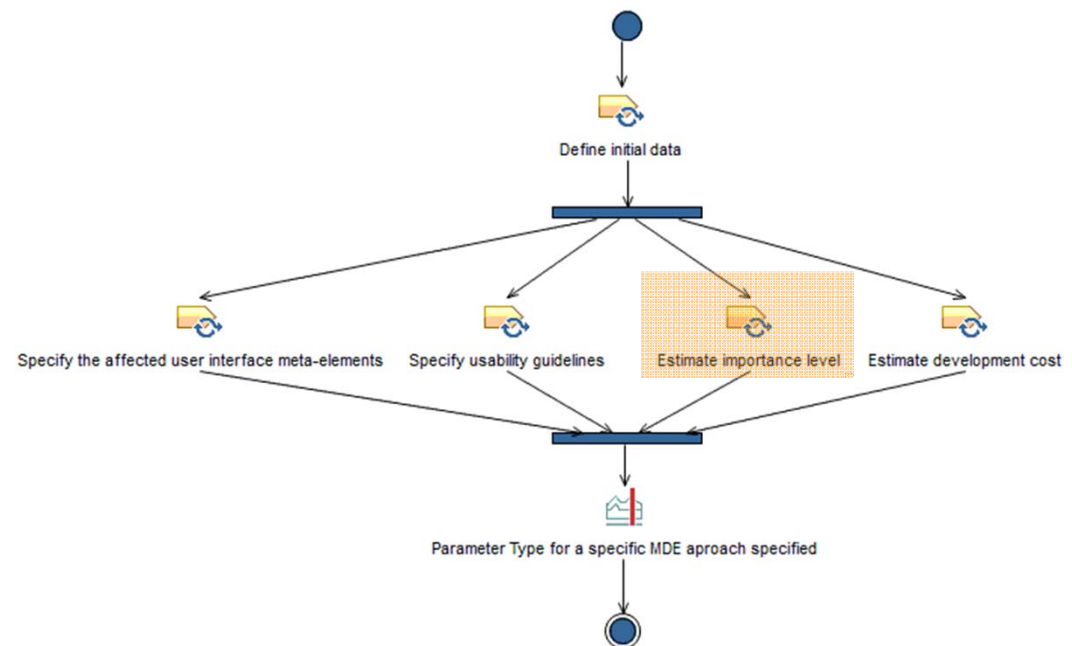
Specify a Parameter Type for an MDE approach

- **Task:** Specify usability guidelines
- **Input:** Parameter Type specification
- **Performing role:** usability expert
- **Output:** Parameter Type specification (updated)
- The usability expert specifies usability guidelines for each pair of context of use and possible value



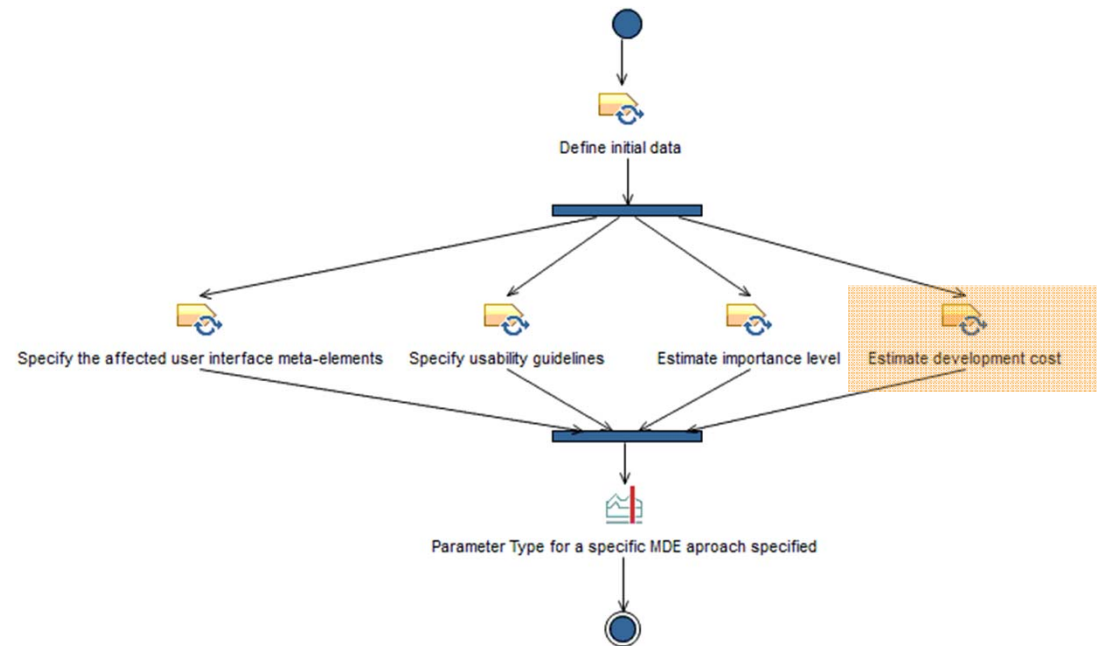
Specify a Parameter Type for an MDE approach

- **Task:** Estimate importance level
- **Input:** Parameter Type specification
- **Performing roles:** UI designers, usability experts, project leader
- **Output:** Parameter Type specification (updated)
- The estimation is performed for each pair of context of use and possible value
- The estimation can be based on
 - Experience of the involved roles
 - Frequency of the UI problem that will be solved using the considered possible value



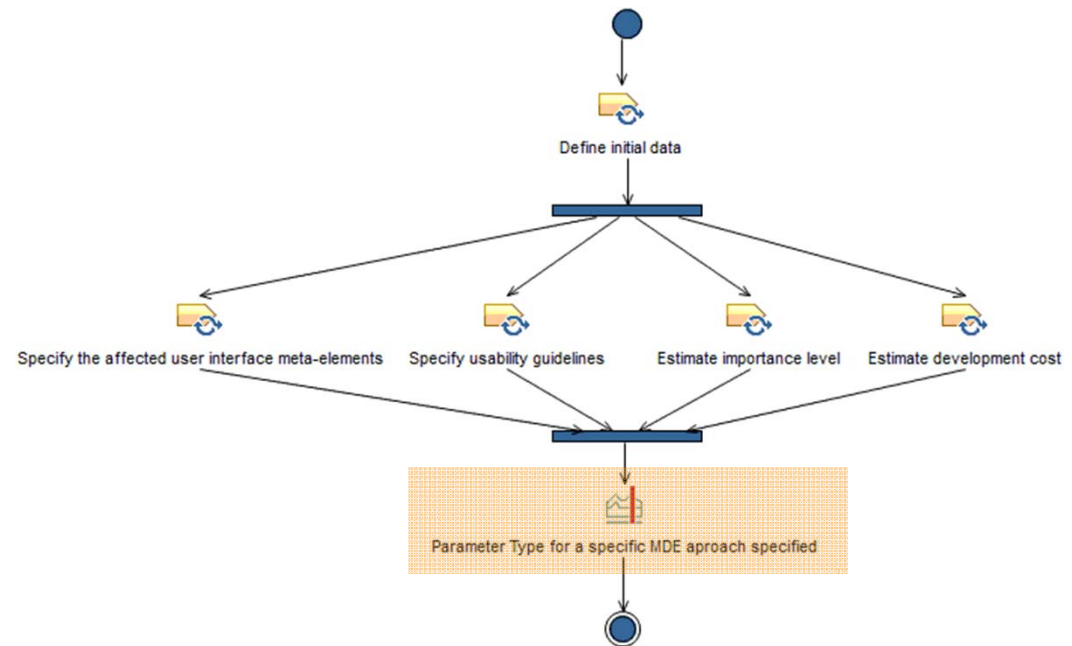
Specify a Parameter Type for an MDE approach

- **Task:** Estimate development cost
- **Input:** Parameter Type specification
- **Performing roles:** MDE UI expert, software developers
- **Output:** Parameter Type specification (updated)
- The estimation is performed for each pair of context of use and possible value
- Method for cost estimation
 - **Recommendation:** use the same method that is used in the development and maintenance of the selected MDE approach
 - Formal, objective method: e.g., COCOMO II, function points
 - Informal, subjective method: e.g., a simple punctuation of low, medium, or high



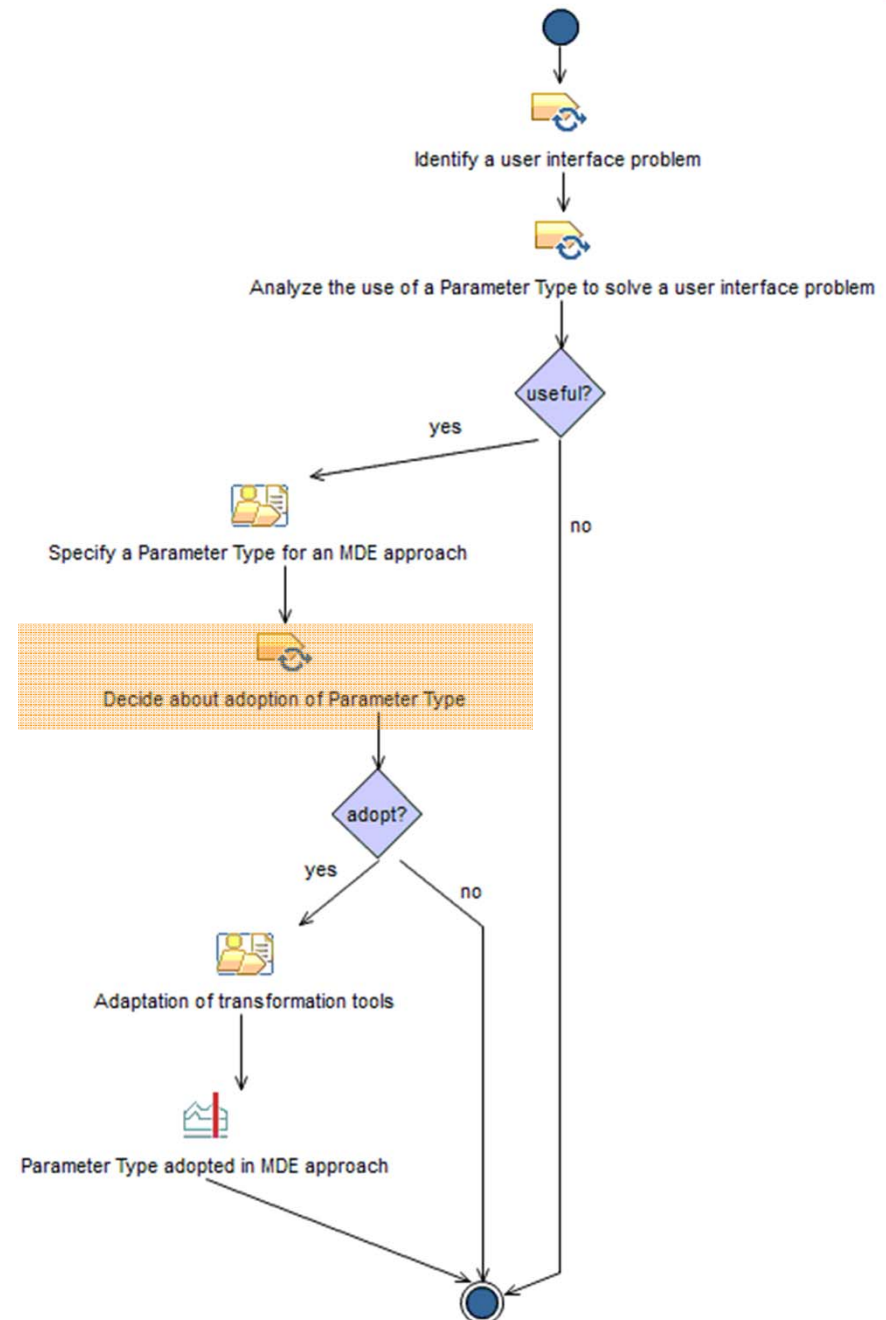
Specify a Parameter Type for an MDE approach

- **Milestone**
 - Parameter Type for a specific MDE approach specified



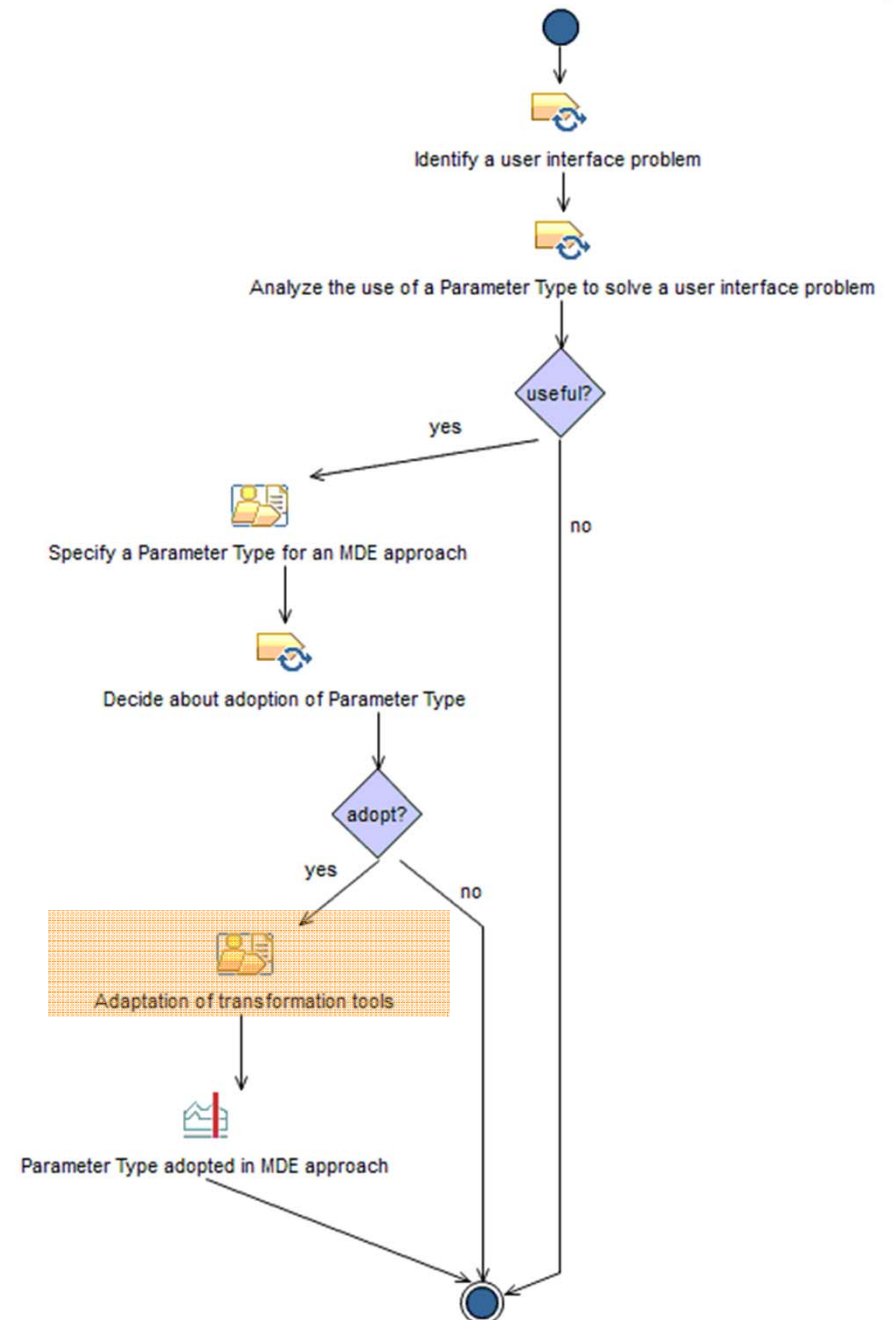
Adopt a Parameter Type in an MDE approach

- **Task:** Decide about adoption of a Parameter Type
- **Input:** Parameter Type specification
- **Performing roles:** project leader
- The project leader decides which possible values of a parameter type will be implemented in which contexts of use and tools
- Estimations of importance level and development cost provide information that is useful to make this decision



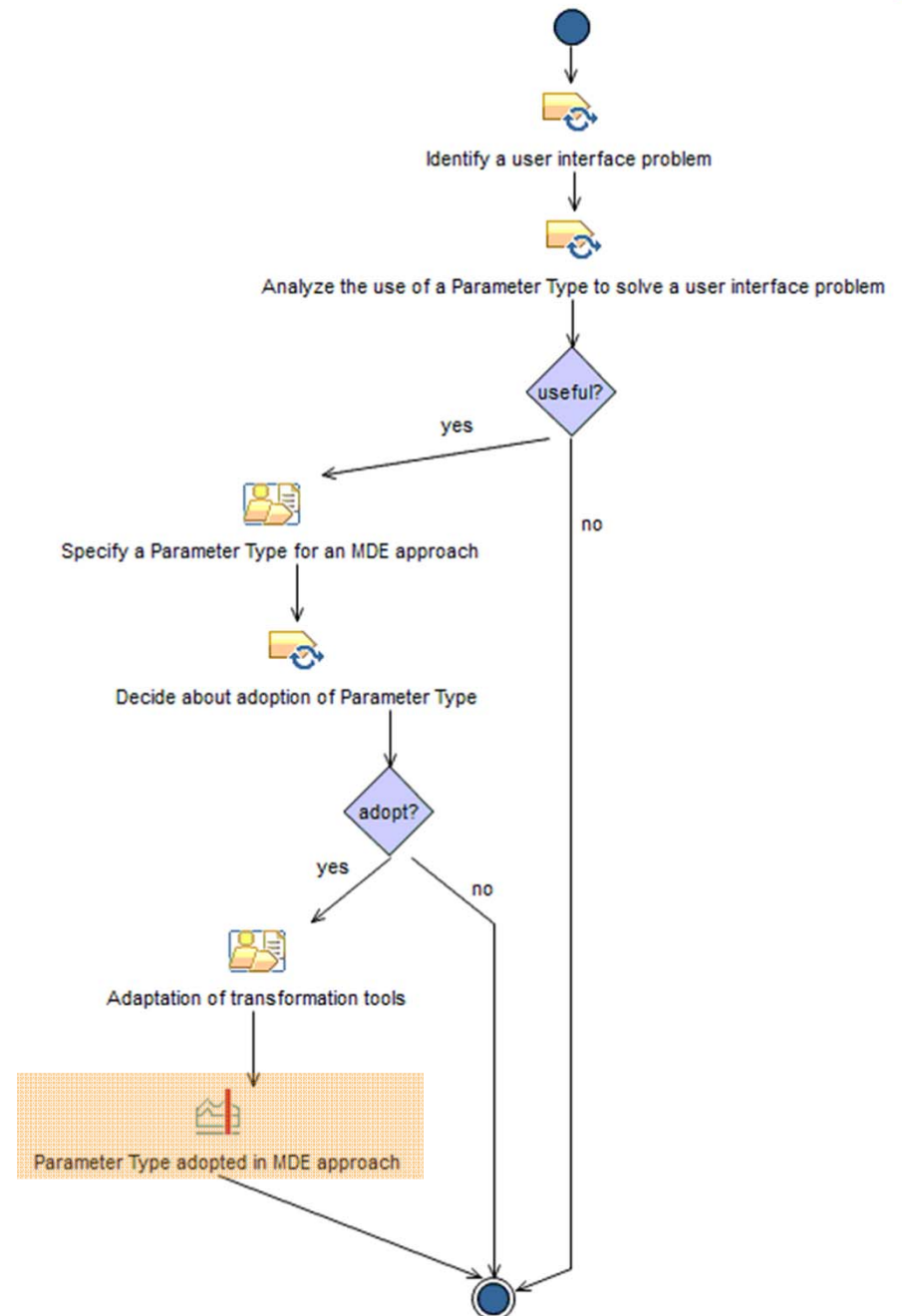
Adopt a Parameter Type in an MDE approach

- **Sub-process**
 - Adaptation of transformation tools
- Black box that represents the development process that is used to adapt the selected MDE approach
- It can comprise a complete development process
 - From analysis and design to implementation, testing, and maintenance
- If the selected MDE approach uses implicit transformation logic, the transformation tool itself should be adapted
- If the selected MDE approach uses explicit transformation logic, the set of transformation rules should be adapted



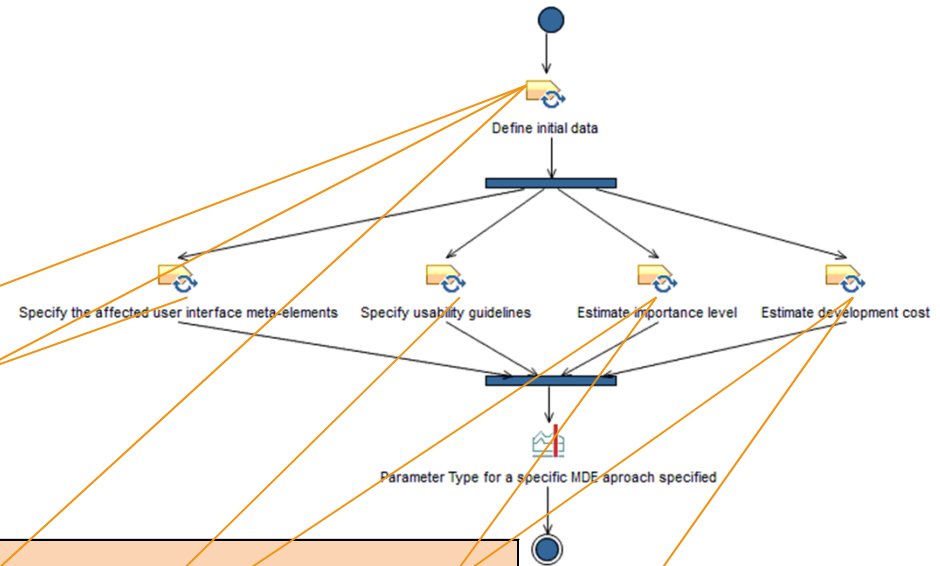
Adopt a Parameter Type in an MDE approach

- **Milestone**
 - Parameter Type adopted in MDE approach



Applying the process to adopt a Parameter Type in OO-Method

- OO-Method
 - Object-oriented method
 - Allows the automatic generation of software applications from conceptual models



Parameter Type				
Name	Affects	Possible values enumeration		
		Value	Graphical description	
Grouping layout	Service Interaction Unit, Argument Grouping (elements of the OO-Method Presentation Model – Pastor 2007)	group box		
		tabbed dialog box		
		accordion		
		Contexts		
		<i>Desktop – C# on .NET</i>		<i>Web – ASP.NET</i>
Possible value	Importance level	Development cost	Importance level	Development cost
group box	high	low	high	low
tabbed dialog box	high	low	high	low
accordion	low	high	high	medium
Possible value	Usability guidelines (for desktop context)			
group box	Visual distinctiveness is important. The total number of groups will be small.			
tabbed dialog box	Visual distinctiveness is important. The total number of groups is not greater than 10.			
accordion	Visual distinctiveness is important. The total number of groups is not greater than 10.			

Conclusion

- Main contribution
 - Two processes that provide guidance to integrate Transformation Templates with different MDE approaches for UI development were defined using SPEM
- Ongoing and further work
 - We have defined other related processes which consider
 - The adoption of a set of Parameter Types, instead of one at a time
 - The specification of a generic
 - Parameter Type
 - Catalog of Parameter Types
 - The derivation of a Parameter Type for a specific MDE approach from a generic Parameter Type
 - Validations: with other MDE approaches and a larger set of Parameter Types

Thank you very much for your attention

Nathalie Aquino

nathalie.aquino@uca.edu.py - naquino@pros.upv.es

Luca Cernuzzi

lcernuzz@uca.edu.py

Oscar Pastor

opastor@pros.upv.es