A metamodel to guide a requirements elicitation process for embedded systems

Tarcísio Pereira, Aêda Sousa, Reinaldo Silva, Deivson Albuquerque, Fernanda Alencar, and Jaelson Castro

tarcisio.couto@ifsertao-pe.edu.br
tcp@cin.ufpe.br
Outline

• Motivation;
• Goals;
• Metamodel Development Process;
• Requirements Elicitation Process;
• Conclusions and Future Work.
Motivation

• Broy (1999) defines Embedded System as a system that regulates a physical device by sending control signals to actuators in reaction to input signals provided by its users and by sensors capturing the relevant state parameters of the system.

• Embedded Systems (ES) are present in different domains such as automation technology, automotive, avionics, energy technology or medical technology.
Motivation

• In everyday life, people are dependent on several services supported by software, many of them transparent to the final user.

• The majority of ES are less visible, and they run in engines, brakes, seat belts, airbag, and audio system in your car.

• Furthermore, such systems are designed to repeatedly carry out a specific function, keeping its operation under different constraints from the ones of general purpose systems [2], [4]
Motivation

• According to Broy (1999) [1], in the embedded system domain, more than 50% of the problems occur when the system is delivered. [misconceptions in capturing requirements];

• These shortcomings are the result of inappropriate Requirements Engineering (RE) tasks, resulting in incomplete requirements, incorrect elicitation and specification, high complexity, and economic or human loss.

• A requirements engineering process is crucial to meet time, cost, and quality goals [7];
Motivation

• Important point;
• Some studies [8-10] have investigated the concepts that should be considered during Embedded Systems development. However, these studies did not capture the core concepts and appropriate evidence.

• Additionally, a systematic investigation was not performed;
Motivation

• In fact, there is much confusion among requirements engineers and stakeholders due to the different kinds of information that need to be managed.

• A metamodel that capture embedded systems concepts would constitute a significant step forward to improve the requirements quality of embedded systems.
Goals

• Update and evaluate a metamodel (MM4ES) for embedded systems;

• Develop a Requirements Engineering Process for Embedded Systems (REPES) based on the metamodel concepts; In this research, we focused on elicitation actions.
Metamodel Development Process

Adapted from Usman et al. [19]
(A) Planning

• Knowledge area
• What is the core set of requirements engineering information that should be specified by requirements engineers in the development of embedded systems?

(B) Identification and extraction

• Data source: articles from our SLR;
• Inspection to identify the embedded systems concepts;
• Concepts definition;
(C) Design and Construction

- Define relationships and cardinality between concepts;
- Implement the metamodel;

(D) Evaluation

- First evaluation – The metamodel was analyzed three times by a domain expert to check if the concepts were correct from his point of view in the context of medical device development;
- Second evaluation - we tried to demonstrate the utility of the metamodel by tracing the results of the elicitation process actions to the embedded systems concepts using the metamodel as a basis.
(C) Design and Construction

- Define relationships and cardinality between concepts;
- Implement the metamodel;

(D) Evaluation

- First evaluation – The metamodel was analyzed three times by a domain expert to check if the concepts were correct from his point of view in the context of medical device development;
- Second evaluation - we tried to demonstrate the utility of the metamodel by tracing the results of the elicitation process actions to the embedded systems concepts using the metamodel as a basis.
(C) Design and Construction

• Define relationships and cardinality between concepts;
• Implement the metamodel;

(D) Evaluation

• First evaluation – The metamodel was analyzed three times by a domain expert to check if the concepts were correct from his point of view in the context of medical device development;
• Second evaluation - we tried to demonstrate the utility of the metamodel by tracing the results of the elicitation process actions to the embedded systems concepts using the metamodel as a basis.
Metamodel for Embedded Systems – MM4ES
Metamodel for Embedded Systems – MM4ES

- **MemoryROM**
- **Processor**
- **Microprocessor**
- **MemoryRAM**
- **Buttons**
- **Controller**
- **Keypad**
- **InputUserInterface**
- **Trimmer Potentiometer**
- **LegacyHardware**
- **PowerSupply**
- **StorageDevice**
- **Seven-Segment display**
- **Microcontroller**
- **Digital Signal Processor**
- **Graphical Display**
- **LCD Display**
- **OutputUserInterface**

**HardwareDevice**
- Electronic: Requi
- Mechanical: Requi
- Name: String
- Description: String

(32-34) (37) (38)
Metamodel for Embedded Systems – MM4ES
Process Development Methodology

- (1) Knowledge acquisition
- (2) Problem definition
- (3) Mapping of concepts
- (4) Identification of information sources
- (5) Definition of process design
- (6) Development of the process
Process Development Methodology

- (1) Knowledge acquisition
- (2) Problem definition
- (3) Mapping of concepts
- (4) Identification of information sources
- (5) Definition of process design
- (6) Development of the process

SLR
Process Development Methodology

- (1) Knowledge acquisition
- (2) Problem definition
- (3) Mapping of concepts
- (4) Identification of information sources
- (5) Definition of process design
- (6) Development of the process

(i) the lack of a requirements engineering process. (ii) what should be considered to develop the RE process?
Process Development Methodology

- (1) Knowledge acquisition
- (2) Problem definition
- (3) Mapping of concepts
- (4) Identification of information sources
- (5) Definition of process design
- (6) Development of the process
Process Development Methodology

- (1) Knowledge acquisition
- (2) Problem definition
- (3) Mapping of concepts
- (4) Identification of information sources
- (5) Definition of process design
- (6) Development of the process

<table>
<thead>
<tr>
<th>Sub-process</th>
<th>Definition of behavioral requirements</th>
</tr>
</thead>
<tbody>
<tr>
<td>Metamodel Concepts</td>
<td>Action, Software Behavior, Hardware Behavior</td>
</tr>
<tr>
<td>State of the Art (SLR)</td>
<td>S01, S20, S24, S27, S28, S29, S31, S44, S53, S56, S60, S62, S67, S73, S75, S79</td>
</tr>
</tbody>
</table>

Metamodel concepts
- Studies of the SLR
- RE Standards
Process Development Methodology

- (1) Knowledge acquisition
- (2) Problem definition
- (3) Mapping of concepts
- (4) Identification of information sources
- (5) Definition of process design
- (6) Development of the process
The REPES Process

- Using the process, ES engineers can manage the elicitation activity in an organized way;
- 24 sub-processes;
- 89 actions;
- 49 actions related to requirements elicitation.
# The REPES Process – Elicitation Actions

<table>
<thead>
<tr>
<th>#</th>
<th>ID</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>22</td>
<td>SR.DSG.a1</td>
<td>Provide short statements describing what the system must accomplish</td>
</tr>
<tr>
<td>23</td>
<td>SR.DSG.a3</td>
<td>Elicit a set of system goals from stakeholder’s needs</td>
</tr>
<tr>
<td>24</td>
<td>SR.IEA.a1</td>
<td>Identification of Environmental Assumptions</td>
</tr>
<tr>
<td>25</td>
<td>SR.IEA.a5</td>
<td>Establish a list of devices that monitor the monitored and controlled variables</td>
</tr>
<tr>
<td>26</td>
<td>SR.DBR.a1</td>
<td>Provide a set of software behaviors to document the actions the software should perform</td>
</tr>
<tr>
<td>27</td>
<td>SR.DBR.a2</td>
<td>Provide a set of hardware behaviors to document the actions the hardware should perform</td>
</tr>
<tr>
<td>28</td>
<td>SR.MCI.a1</td>
<td>Management of Contextual Information</td>
</tr>
<tr>
<td>29</td>
<td>SR.MCI.a2</td>
<td>Identify and document statements, facts, and variables</td>
</tr>
<tr>
<td>30</td>
<td>SHR.HRC</td>
<td>Software and Hardware Requirements</td>
</tr>
<tr>
<td>31</td>
<td>SHR.HRDA2</td>
<td>The constraints for each hardware device previously identified are defined and documented</td>
</tr>
<tr>
<td>32</td>
<td>SHR.HRDA3</td>
<td>Hardware Requirements Definition</td>
</tr>
<tr>
<td>33</td>
<td>SHR.HRDA4</td>
<td>Elicit the mechanical requirements</td>
</tr>
<tr>
<td>34</td>
<td>SHR.HRDA5</td>
<td>Elicit electrical requirements</td>
</tr>
<tr>
<td>35</td>
<td>SHR.HRDA6</td>
<td>Identify and document the microcontroller of the embedded system based on the results of previous actions</td>
</tr>
<tr>
<td>36</td>
<td>SHR.HRDA9</td>
<td>Elicit a set of manufacture requirements</td>
</tr>
<tr>
<td>37</td>
<td>SHR.DIO.a1</td>
<td>Define Input and Output User Interface</td>
</tr>
<tr>
<td>38</td>
<td>SHR.DIO.a2</td>
<td>Define and document a set of input user interface</td>
</tr>
<tr>
<td>39</td>
<td>SHR.DIO.a3</td>
<td>Define and document a set of output user interface</td>
</tr>
<tr>
<td>40</td>
<td>SHR.DSA.a1</td>
<td>Define Sensors and Actuators</td>
</tr>
<tr>
<td>41</td>
<td>SHR.DSA.a5</td>
<td>Define a set of sensors and actuators standards</td>
</tr>
<tr>
<td>42</td>
<td>SHR.DELA1</td>
<td>Identify and document a set of external interfaces and hardware adapters standards</td>
</tr>
<tr>
<td>43</td>
<td>SHR.DELA2</td>
<td>Identify and document a set of external interfaces</td>
</tr>
<tr>
<td>44</td>
<td>SHR.DELA3</td>
<td>Identify and document the hardware adapters</td>
</tr>
<tr>
<td>45</td>
<td>SHR.SFC.a1</td>
<td>Identify and document the software constraints</td>
</tr>
<tr>
<td>46</td>
<td>SHR.SRD.a1</td>
<td>Elicit and document a set of functional software requirements</td>
</tr>
<tr>
<td>47</td>
<td>SHR.SRD.a5</td>
<td>Elicit and document a set of non-functional requirements</td>
</tr>
<tr>
<td>48</td>
<td>SER.RMG.a1</td>
<td>Identify and document potential risks</td>
</tr>
<tr>
<td>49</td>
<td>SER.FMGA1</td>
<td>Identify, classify, and document potential software and hardware failures</td>
</tr>
</tbody>
</table>
Process Usage

• Use of the process: when an organization decides to introduce requirements engineering activities in its embedded systems development process;

• The actions can be performed in a sequential way to get a document with the requirements elicited;

• We used a real world scenario related to a medical device as a proof of concept. The goal is to demonstrate the utility of the process by presenting examples as results of the actions;
• We used a set of documents as input for the requirements elicitation process;
• Scenario: Patient-controlled analgesic (PCA) infusion pump;

26 - SR.DBR.a1 - (i) after the start button has been pushed, a timer counter shall be displayed, and (ii) when the infusion is in progress, a boolean signal shall be displayed.

27 - SR.DBR.a2 - (i) after the button A has been pushed, a red light shall be lit, and (ii) after the dose button has been pushed, two beeps shall be sounded, and the pump will begin delivering the demand dose.
Conclusion

• We propose a metamodel for embedded systems to describe the main concepts that should be elicited and specified by requirements engineers; and

• We used the metamodel to guide the development of a requirements elicitation process;

• The process describes actions for the identification and definition of embedded systems requirements.
Conclusion

• The main contributions of this research are the following:
  1. a knowledge-based metamodel;
  2. a reference metamodel that can be updated in future works;
  3. a metamodel that can be used as input for model transformation;
  4. an elicitation process;
  5. the process can be used as a guide to assess the requirements activities of organizations.
Limitations

1. Only a metamodel cannot address all characteristics of an ES for a specific domain;
2. The metamodel and the requirements elicitation process was not evaluated in the industry yet; and
3. A threat may be the selection of actions that are included in the REPES process since they were based on the metamodel concepts and RE standards.
Research directions

1. How can we evaluate the completeness of the proposed metamodel?
2. How can we extend the metamodel to represent the specific characteristics of the different domains of an ES?
3. How can we evaluate whether the process has sufficient coverage of actions?
4. How can we validate the usefulness and ease of use of the process?
5. How can we develop a CASE tool to support the process?
6. How can we conduct a comparative analysis with other studies using the same case study (PCA)?
References


References


Petrolina - Brazil

Thank you!

Caruaru - Brazil
A metamodel to guide a requirements elicitation process for embedded systems

Tarcísio Pereira, Aêda Sousa, Reinaldo Silva, Deivson Albuquerque, Fernanda Alencar, and Jaelson Castro

tarcisio.couto@ifsertao-pe.edu.br
tcp@cin.ufpe.br