Extending Telecom Service design activities for early verification of telecom services

PhD student: Iyäs Alloush (2nd year)

Prof. Yvon Kermarrec, A/Prof. Siegfried Rouvrais

Telecom Bretagne
Our scope & Aim

- Our work is in the domain of Telecom Service Creation (TSC) Activities.
- A recent contribution [Chiprianov, 2012] applies Enterprise Architecture (EA) and Model Driven Engineering (MDE) principles for the collaborative Construction of TS.
- EA shares the views of the different stakeholders, and decomposes a complex architecture into aspects and layers.

- We aim to propose methods and tools to assist different stakeholders (designer, service provider) that are involved in the TS specification activity.
Outlines

- Context
- Challenges
- Model Driven Engineering and Model Driven Analysis
- Approach and Methodology
- Related work
- Contributions
- Example
- Conclusions
- Future work
Context: Functional and Non-Functional Requirements

-Functional requirements (FR) represent the functions that the TS is required to perform.
  • In our approach, FRs are represented in the behavioral view.

-Non-Functional requirements (NFR) is a software requirement that describes not what the software will do, but how the software will do it.
  • NFRs can be described by the softgoals.
Context: Qualities

- Performance is one of the qualities that belongs to the NFRs.
  - It has attributes such as QoS attributes (e.g. delays) “End-User satisfaction”.
  - Other attributes such as CPU load can be measured at a specific node. “Service and Network providers satisfaction”.
  - Network simulators are able to provide the measurements of performance attributes (e.g. OPNET, NS-3).
Challenges

- How to detect the TS design errors and flaws earlier before the implementation phase?

- How to benefit from the detected errors and flaws to correct the design and improve the qualities?
Outlines

- Context
- Challenges
- Model Driven Engineering and Model Driven Analysis
- Approach and Methodology
- Related work
- Contributions
- Example
- Conclusions
- Future work
Model Driven Engineering and Model Driven Analysis

- In the scope of MDE, model transformation is a key concept to transform a source model to a target one automatically.
  - There are different types of model transformation such as M2M, and M2T.
  - We use M2T, as we need to generate configuration scripts for the tools.

- Model Driven Analysis is an approach that aims to verify if the input design model meets the quality requirements.
Approach and Methodology

- Applying Model Driven Engineering and Model Driven Analysis we propose to extend the design activities to verify the TS design earlier and before the implementation phase.

- Thus, we divide our verification activity into 4 sub-activities: Linking, Model Adaptation, Measurement Analysis, and Feedback.
Outlines

- Context
- Challenges
- Model Driven Engineering and Model Driven Analysis
- Approach and Methodology
- Related work
- Contributions
- Example
- Conclusions
- Future work
Related work

<table>
<thead>
<tr>
<th>Approach/Aspect</th>
<th>Supports Multi-Actor views</th>
<th>Uses Simulation</th>
<th>Uses multiple simulators</th>
<th>PIMs to code</th>
<th>PSMs to code</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Pervasive Services validation using PN</td>
<td>+</td>
<td>+</td>
<td>-</td>
<td>+</td>
<td>-</td>
</tr>
<tr>
<td>2. Pervasive Services validation using Micro Java</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>+</td>
<td>+</td>
</tr>
<tr>
<td>3. TSCE relying on IMS and using Eclipse</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>+</td>
<td>+</td>
</tr>
<tr>
<td>4. TSCE and validation using Java and C++ exe codes</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>+</td>
<td>-</td>
</tr>
<tr>
<td>5. Our Approach</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
</tr>
</tbody>
</table>

As far as we know, our approach is the only one that can achieve all of the mentioned aspects, that are basics for the early verification of the telecom services

1. Achilleos et al., 2008
2. Yang et al., 2005
4. Adamopoulos et al., 2002
Outlines

- Context
- Challenges
- Model Driven Engineering and Model Driven Analysis
- Approach and Methodology
- Related work
- Contributions
- Example
- Conclusions
- Future work
Contributions: Linking between technical spaces

- We have implemented code generation from the upper layers of EA (Business, Application) to Java code => (PIM to Java)
- Using XPAND code generation, we have linked directly the high abstract models to network simulators relying on IMS platform => (PSM to OPNET simulator)
Contributions: Extending the MM (Generalization of Methodology)

Using the same Archimate language, we have extended the mentioned MM to include new verification entities: Measurements, tool specification, and new features that make it possible for the MM to be used with different network simulators such as (OPNET and NS-3).
Contributions: Extending the MM - New entities (a simplified view)

```
<<struct>>
SIP_HEADER
+To: apstring
+From: apstring
+Cseq: apstring
+Callid: apstring
+Max-Forwards: apstring
+Via: apstring

<<struct>>
SIP_MESSAGE
+StLine: apvector<apstring>
+Headers: apvector<SIP_HEADER>
+Body: apstring
```
Contributions: Tool Selection

- Relying on the same extended MM we have implemented a method to select between different tools according to the measurement capability of each.
Outlines

- Context
- Challenges
- Model Driven Engineering and Model Driven Analysis
- Approach and Methodology
- Related work
- Contributions
- Example
- Conclusions
- Future work
Example: Video Conference

- We use the same TS model that is experimented with OPNET, and use it after extensions to generate code for NS-3.
- The new entities that are added are the measurement models, and their relations with the design.
  - Measurement configuration is generated automatically and assigned to the related node or function in the IMS structure.
Example: Video Conference

```c
int main (int argc, char *argv[]){ // We define the whole Telecom Service in
    //**Design Part
    //*** Structural Configuration
    NodeContainer NC;
    NC.Create(7);
    //To add the names of the nodes to the name space of NS-3
    Names::Add ("Terminal1",NC.Get(0));
    Names::Add ("HomeSubscriberServer",NC.Get(1));
    Names::Add ("Interrogating_Call_SessionControlFunction1",NC.Get(2));
    Names::Add ("ApplicationServer1",NC.Get(3));
    Names::Add ("Serving_Call_SessionControlFunction1",NC.Get(4));
    Names::Add ("MediaResourceFunctionProcessor1",NC.Get(5));
    Names::Add ("Proxy_Call_SessionControlFunction1",NC.Get(6));

    Ipv4AddressHelper Network1;
    Network1.SetBase ("10.1.1.0",255.255.255.0);

    //***This is to configure the link component
    PointToPointHelper CommunicationPathTerminal1_P_CSCF1;
    //****Here, we can define the predefined QoS attributes of the P-P 1r
    NetDeviceContainer dev_CommunicationPathTerminal1_P_CSCF1;
    dev_CommunicationPathTerminal1_P_CSCF1 = CommunicationPathTerminal1_P_CSCF1.Install (Proxy_Call_SessionControlFunction1,Terminal1); //function

    // define the 4th Layer issues for the interfaces that form the 2 ends of a communication path Communication Path Terminal 1 - P-CSCF 1
    InternetStackHelper CommunicationPathTerminal1_P_CSCF1_stack;
    CommunicationPathTerminal1_P_CSCF1_stack.Install(Terminal1);

    // define the 3rd Layer issues for the interfaces that form the 2 ends of a communication path Communication Path Terminal 1 - P-CSCF 1
    Ipv4InterfaceContainer CommunicationPathTerminal1_P_CSCF1_IpContainer = Network1.Assign(dev_CommunicationPathTerminal1_P_CSCF1);
}
```
Outlines

■ Context
■ Challenges
■ Model Driven Engineering and Model Driven Analysis
■ Approach and Methodology
■ Related work
■ Contributions
■ Example
■ Conclusions
■ Future work
Conclusions

- We have defined a method to link high abstract models to simulation level.
- Our approach is capable to generate GPL code from PIMs, and Simulation codes from PSM.
- We generalized our methodology so the MM is reusable for different network simulators, this conserves the Telecom standards and design constraints from frequent changes.
- We are able to obtain measurements from network simulators, this helps in detecting design errors and improving the qualities.
Future work

- We intend to:
  - Define an approach to perform the measurement analysis according to the functional and non-functional requirements.
  - Integrate between the different tool results.
  - Enhance our tool repository with more tools that can serve our approach such as model checkers.
  - To select the measurements from the quality requirements using MDE principles.
Thank you

Thank you for your attention

Email: iyas.alloush@telecom-bretagne.eu

Publishes

- Alloush, I.; Kermarrec, Y. & Rouvrais, S. On verifying the telecom services through design models: A generalized model transformation approach to link design models to simulators (SimuTech2013) [submitted]
- Alloush, I.; Kermarrec, Y. & Rouvrais, S. An Automated Tool Selection Filter based on Model Transformation: OPNET and NS-3 Case Study (Spects 2013) [submitted]