MCA Driven Interaction Interfacing

Daniela Elisabeth Ströckl
Institute of Applied Research on Aging
Carinthia University of Applied Sciences
Klagenfurt, Austria
e-mail: d.stroeckl@fh-kaernten.at

Heinrich C. Mayr
Institute of Applied Informatics
Alpen-Adria-Universität
Klagenfurt, Austria
e-mail: Heinrich.Mayr@aau.at

***Abstract*— Assistive technologies can help older people stay at home longer and more independently. Like any other user group, seniors have specific profiles and demands that are to be considered when designing such technologies. This paper presents an approach to multimodal user interface design and development based on the Model-Centered Architecture paradigm. The research is part of the Human Behavior Monitoring and Support (HBMS) project that aims at providing a comprehensive assistive system. The approach should enable assistive systems to be flexibly adapted to any user needs with regard to interaction and corresponding end devices.**

Keywords-Model-Centered Architecture; Conceptual Modelling; Smart Home; Active & Assisted Living; AAL; Assistance System; Activities of Daily Living; ADLs.

I. Introduction

In a technology-driven approach, the use of devices such as smartphones, tablet computers or household appliances is gradually becoming a standard in everyday life. However, elderly people or people with impairments have trouble to interact with such devices depending on their individual disabilities. Therefore, several researchers are working on multimodal development strategies to facilitate the interaction between humans and systems.

Complex interactions are a challenge for users and developers. In particular, a system should allow the user to communicate in a convenient and natural way in any usage situation. Therefore, a system should be aware of the context, learn, and behave according to the particular user. Examples for related work dealing with multimodal human-system interactions are: models for multimodal interaction by Dausend et al. [1], multimodal interaction in smart environments by Blumendorf [2], or Conceptual Modeling of Interactions by Aquino et al. [3].

This paper presents an approach that exploits the Model Centered Architecture (MCA) paradigm for designing and developing multimodal interfaces of assistive systems in the context of Active and Assisted Living (AAL). The aspects communicated via such interface are conceptualized in a meta-model that, in turn, forms the basis of a Domain Specific Modeling Language (DSML). Based on this, interface specifications for any device and content consist in models created using that DSML. The system should be flexible enough to select from a group of different interactive media and devices one suitable for each person to help with activities at home. We will illustrate this approach by means of an interface specification for the Human Behavior Monitoring and Support (HBMS) system [4].

The paper's structure is as follows: Section II outlines the main aspects of MCA. Section III shortly describes the HBMS system. In Section IV, we sketch the scenario, which we will use for illustrating our approach. In Section V, we present the most important concepts of the metamodel for the multimodal interface specification we have developed and a model instantiated from this metamodel. Section VI discusses an excerpt from a concrete interface description. The paper closes with an outlook on future work.

II. Model Centered Architecture

MCA focusses on models and their meta-models in any development step of software systems. For example, the functionality as well as all interfaces of a system are specified as conceptual models using appropriate Domain Specific modeling Languages (DSMLs) for that purpose. Consequently, all system components are seen as model handlers in the sense of model consumers and/or producers [5].

MCA exploits the model hierarchy as defined in the Meta Object Facility (MOF) [6]: Using a standard meta-meta-model with a pre-defined notation on the M3 level, we determine the meta-models (and for representational purposes the DSMLs) [7] for the various interfaces and the application domain on the M2 (meta-model) level. The M1 level models are extensions of these meta-models and determine a particular system application. The extensions of these models on the M0 level then represent the concrete processes and data of the running system. For example, every input/output device as well as any functional system module deals with or produces extensions of one or more M1 models [8].

III. Example Application Environment: HBMS

The aim of the HBMS system is to provide elderly people with unobtrusive support in their everyday activities. It uses the output of Human Activity Recognition (HAR) systems [9] to learn a user's strategies for mastering his or her home activities, establishes a Human Cognitive Model (HCM), which then is exploited when situations are detected in which the user needs support [4]. In the best case, a HBMS system instance is set up in a user's environment before dementia processes interfere with the learning process

[10]. The development strategy of the HBMS system strictly follows the MCA paradigm.

The system consists of several components, among which we list:

- HCM-L Modeling Tool: supports the Human Cognitive Modeling Language (HCM-L), a DSML defined for establishing human cognitive models, as mentioned before. The tool has been generated using the ADOxx meta-modeling framework [11].
- HBMS Kernel, consisting of
  - Observation Engine: investigates the behaviour and context of the end user through coupled HAR systems in order to identify her/his goals and actions to achieve these goals.
  - Behavior Engine: evaluates the observed behavior and context, as well as a database of former support cases, and decides on how to proceed according to the HCM; in parallel, new activities and situations are learned.
  - Support Engine: provides the end user with situational support information via the human-system interface.
- HBMS Database: A triple store containing all data: HCM, models and meta-models, domain ontology, case base, and a situational cache describing the current situation [4].

IV. Human-System Interaction: Interfaces & Devices

Within this section, we sketch a scenario we selected for a proof of concept, the input/output devices used in each case, and the DSML for specifying the interfaces for these devices.

The scenario "Morning Routine" consists of several sub-scenarios: brushing teeth, washing face, washing hands, comp hair and apply a cream on face. All these sub-scenarios are feasible in different sequences during the morning routine. The user of the assistive system should get support in this routine whenever she/he needs, for example when searching items like the toothbrush, forgetting to put toothpaste on the brush, and so forth [12].

For this scenario, suitable devices are a boundary microphone for user commands, a mini-beamer for showing short videos, a lightbulb for warnings and, possibly, a tablet computer to show mixed content like text, pictures and videos [12]. Figure 1 shows a use case interaction diagram 2.0 (UC-UI Diagram 2.0) describing the interaction in the "washing face" sub-scenario [13].

V. Meta-Model and DSML For Interface Specification

This section presents the specification of our multimodal interface on all MOF levels excluding level 3 for which we choose a self-explanatory standard notation (as used in Figure 2 for the meta-model specification).

The symbols in the left upper corner of the meta-model represent the notation of the corresponding DSML concepts used when creating models on M1. The main concepts are "Interaction Mode" for capturing the various kinds of information representation, and "Interaction Entity" for capturing the interaction units including user input.

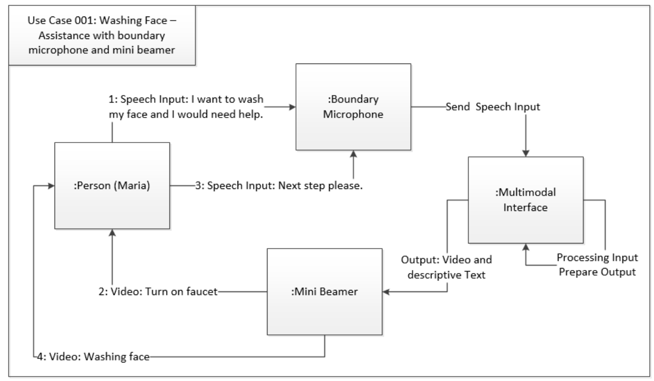

Figure 1. UC-UI Diagram 2.0 for the scenario "Washing Face"

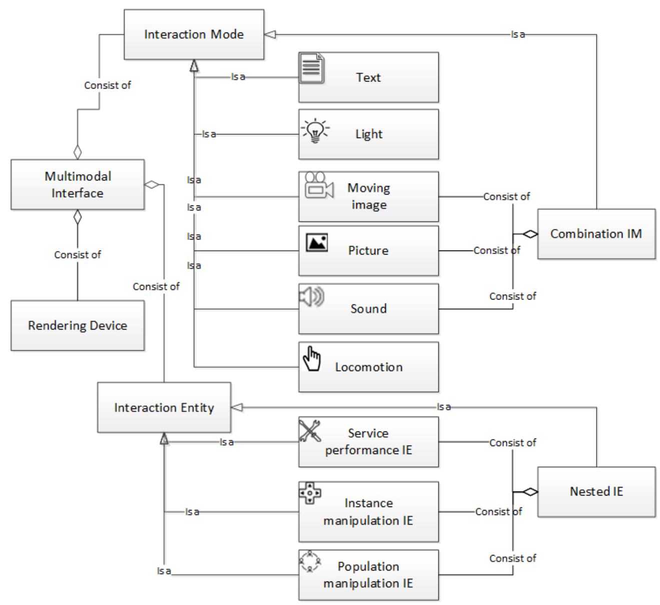

Figure 2. Meta-model (simplified) for the HBMS multimodal human-system interaction

The concepts "Combination IM" and "Nested IE" are introduced to cover situations where multiple modes are combined or user interactions consist of multiple partial interactions. "Rendering device" captures, for a given interaction mode, the device(s) used. For reasons of simplicity and also due to the space limit we omitted in Figure 2 the M2 concept "Attribute" as well as the concept "Relation" (e.g., for describing the relationships between Interaction Modes and Interaction Entities, and between Interaction Entity and Rendering Device).

Figure 3 shows an instantiation of the meta-model given in Figure 2 on level M1. It describes the modes and units that are relevant for the morning routine scenario. Note that this is still an abstraction (a conceptual model), which could be appropriate also for describing the "interfaces" for other activities to be supported.

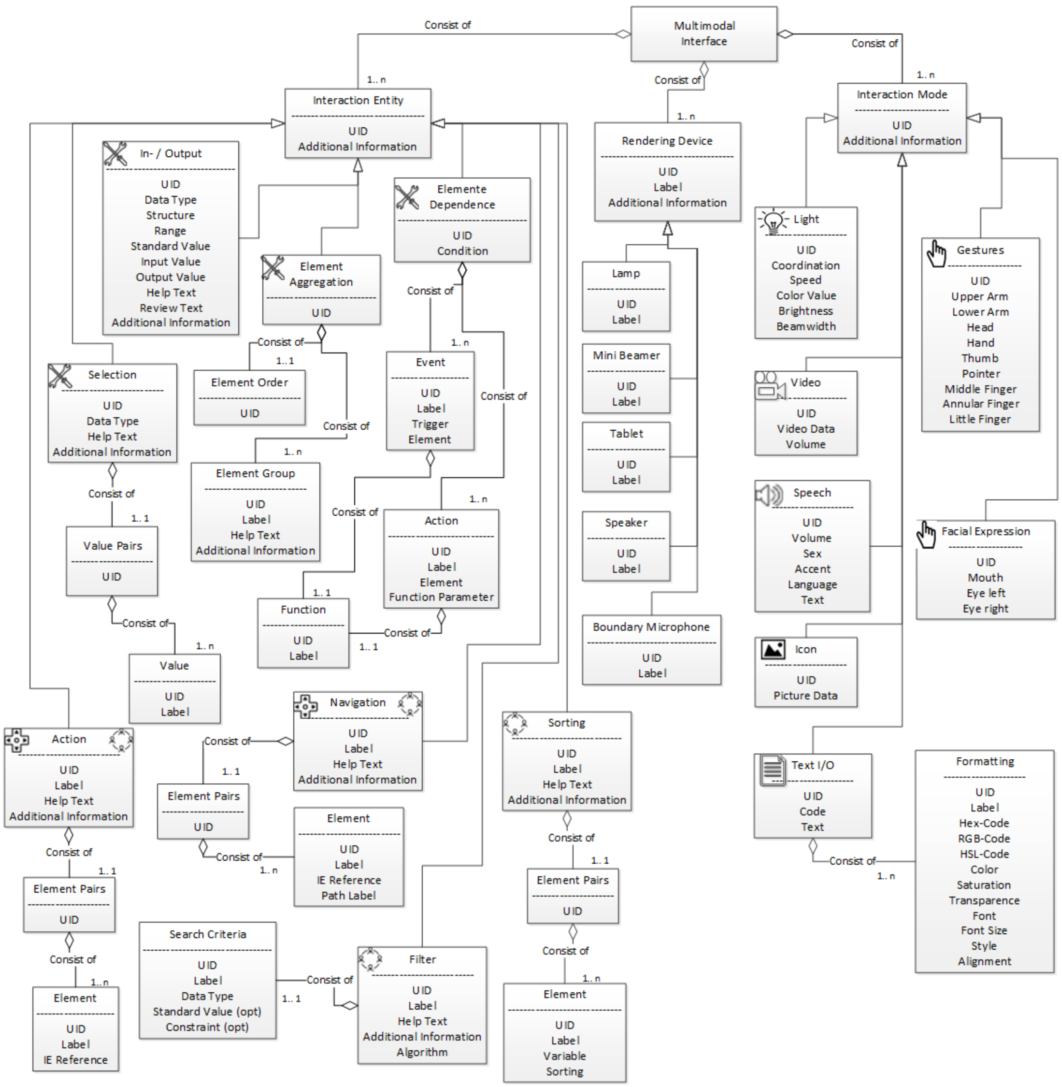

Figure 3. Example of a model (M1)

Consequently, the concrete devices, interaction units and their relationships are instantiated on level M0. We omitted, again for simplicity, the model details of the rendering devices.

The advantage of this approach is that this data can largely be generated on the basis of the M0 models if the corresponding units (text fragments, video clips, etc.) are available in a database. In addition, a standard module (in the case of the HBMS system, the support module) can then control any type of modeled interaction.

VI. REPRESENTATION ON LEVEL M0

In order to show the "code" that results on level M0 on the basis of such models, we give below an excerpt from this code. It describes the possible interactions during a "washing face" action (as part of the "morning routine" activity). In particular, the devices "boundary microphone" and "mini beamer" are addressed.

All these tasks provide the possibility to show a short video about how to process. While the user finished the task, a "finished" speech input will be recorded with the boundary microphone and this leads to the next introduction video.

In this code (Figure 4), fictional variables are chosen for all the elements represented in the M0 level models.

<Multimodal Interface>
<Rendering Device>
<Mini beamer UID="1"; label="Output device mini-beamer";/>
<Boundary Microphone UID="2"; label="audio record with a boundary microphone";/>
</ Rendering Device >
<Interaction Mode>
<Speech UID="1"; Volume="74dB"; Sex="NULL"; Accent="NULL"; Language="English"; Text="NULL";/>
</Interaction Mode>
<Interaction Entity>
<Input UID="1"; Data Type="MP3"; Structure="NULL"; Range="NULL"; Standard Value="NULL"; Input Value="I want to wash my face and I would need help."; Output

```
        Value="NULL"; Help Text="NULL"; Review Text="NULL";
        Additional Information="NULL"; />
        <Action UID="1"; Label="Speech recognition with grammer";
        Help Text="NULL"; Additional Information="Instruction will be
        extracted out of speech input";/>
                <Elements UID="1"; />
                <Element UID="1"; Label="Fragment out of
                speech recognition"; IE Reference="Input 001"; />
        <Action UID="2"; Label="Play video Wash face"; Help
        Text="NULL"; Additional Information="Starting assistance for
        washing the face"; />
                <Elements UID="2"; />
                <Element UID="2"; Label="Loading
                turn_on_faucet.mp4"; IE Reference="Input 001"; />
    </Interaction Entity>
    <Interaction Mode>
        <Video UID="1"; Video Data="turn_on_faucet.mp4";
        Volume="80dB"; />
        <Description UID="01"; Code="UTF-8"; Text="Before you can
        wash your face you have to turn on the faucet."; />
            <Format UID="1"; Label="Video description"; HEX-
            Code="#000000"; RGB-Code="0, 0, 0"; HSL-Code="0,
            0%, 0%"; Color="Schwarz"; Saturation="100%";
            Transparence="0%"; Font Type="Verdana"; Font
            Size="15pt"; Stile="San Serif"; Alignment="left"; />
        <Speech UID="2"; Volume="75dB"; Sex="NULL";
        Accent="NULL"; Language="NULL"; Text="NULL"; />
    </Interaction Mode>
    <Interaction Entity>
        <Input UID="2"; Data Type="MP3"; Structure="NULL";
        Range="NULL"; Standard Value="NULL"; Input
        Value="What's next??"; Output Value="NULL"; Help
        Text="NULL"; Review Text="NULL"; Additional
        Information="NULL"; />
        <Element Dependence UID="1"; Condition="Video 2 after
        Video 1"; />
        <Event UID="1"; Label="Play Video 2 after Video 1";
        Trigger="Speech input"; Element="speechinput.mp3"; />
                </Function UID="1"; Label="Turn of Video 1"; />
        <Action UID="3"; Label="Play wash_face.mp4";
        Element="Input 02"; Function Parameter="wash_face.mp4"; />
                <Function UID="02"; Label ="Start video 2"; />
            …
</ Multimodal Interface >
```

Figure 4. Code example referring to the scenario "Washing face"

Clearly, after having formally defined a comprehensive representation language, such code may be generated automatically, e.g., based on the framework we have developed in the context of HBMS for linking activity recognition systems to the HBMS system. In particular, we can reuse a parser exploiting the ANTLR framework to generate code fragments out of internal data representations that are stored as OWL individuals [14].

VII. Conclusion

This paper explained the current state of work regarding our metamodel for specifying multimodel interfaces for ambient systems that support people in a smart home. The next step will be integrating this approach into the HBMS environment, and providing the models for all kinds of interactions to be supported. These will be refinements of the model shown in Figure 3 by following the multilevel modeling approach, as presented in [15]. Then, tests will be carried out in a lab situation by user experiments. This means that, besides functional testing, seniors will also test the system: with a "think aloud method" they will give feedback to improve the approach.

References

[1] M. Dausend and M. Poguntke, "Executable UML models of multimodal interaction applications." In: Ziegler, J. (eds), i-com: Vol. 10, No. 3. München: Oldenbourg Wissenschaftsverlag GmbH. 2011. pp. 33-39.

[2] D. Roscher, M. Blumendorf, and S. Albayrak, "A meta user interface to control multimodal interaction in smart environments." IUI '09 Proceedings of the 14th international conference on Intelligent user interfaces, 2019, Florida. ISBN: 978-1-60558-168-2

[3] N. Aquino, J. Vanderdonckt, J. Panach, and O. Pastor, "Conceptual Modelling of Interaction", Springer. D.W. Embley and B. Thalheim (eds), Handbook of Conceptual Modeling. Springer, 2011, DOI: 10.1007/978-3-642-15865-0

[4] J. Michael, et al.: "The HBMS Story." In: Enterprise Modelling and Information Systems Architectures – International Journal of Conceptual Modeling, Vol. 13, 2018, pp. 345-370.

[5] H. C. Mayr, J. Michael, S. Ranasinghe, V. Shekhovtsov, and C. Steinberger, "Model Centered Architecture". Cabot et al. (eds.), Conceptual Modeling Perspectives. Springer International Publishing. 2017.

[6] Object Management Group (OMG) OMG Meta Object Facility (MOF) Core Specification Version 2.5.1., 2016, [Accessed: June 2018]

[7] S. Kelly, and J. Tolvanen, "Domain-Specific Modeling: Enabling Full Code Generation." Wiley, 2008, IEEE. ISBN: 978-0-47003-666-2

[8] J. Alvarez, A. Evans, and P. Sammut, "Mapping between Levels in the Metamodel Architecture." In: Gogola M.. Kobryn C. (eds.): UML 2001 – The Unified Modeling Language: Modeling Languages, Concepts and Tools. Springer, Berlin, 2001.

[9] S. Ranasinghe, F. Al Machot, and H. C. Mayr, "A Review on Applications of Activity Recognition Systems with Regard to Performance and Evaluation." Int. Journal of Distributed Sensor Networks, 12(8), 2016. DOI: 10.1177/1550147716665520

[10] H. C. Mayr, "HBMS Research Topics". In: HBMS – Human Behavior Monitoring and Support. Accessed: June 2018. http://hbms-ainf.aau.at/en/hbms-research-topics.html

[11] M. Esperguel, and S. Sepulveda, "Feature modeling tool: a proposal using ADOxx technology." Computing Conference (CLEI), 2016 XLII Latin American, Chile. 2017, ISBN: 978-1-5090-1634-1

[12] D. E. Ströckl, "Scenario based Development Approach towards a Multi-modal Interface Presentation Meta-model." Proceedings Smarter Lives '18, 2018, Innsbruck.

[13] D. E. Ströckl, D. Krainer, and E. Oberrauner, „Improving Interdisciplinary Communication – Use Case Focused User Interaction Diagram 2.0." Ambient '18, 2018, in press.

[14] V. Shekhovtsov et al., "Domain Specific Models as System Links." Advances in Conceptual Modeling, Springer, LNCS Vol 11158, pp. 330-341, 2018.

[15] U. Frank, "Multilevel Modeling. Business & Information Systems Engineering," Vol.5, 2014.