Designing Mobile Assistive Technologies in The Model Driven Development Framework

MONICA DRAGOICEA¹, MIHAIL DUMITRU SACALA², ALINA COJOCARU¹, NAIDEN SHIVAROV³, CLAUDIU BALAN¹
¹Automatic Control and Computers Faculty
University Politehnica of Bucharest
313 Splaiul Independentei Str., 060042-Bucharest
ROMANIA
m.dragoicea@ics.pub.ro    http://monicadragoicea.ics.pub.ro
²Statistics and Economics Department
The Bucharest Academy of Economic Studies
6, Romana Square, District 1, Bucharest
ROMANIA
³Central Lab. of Mechatronics and Instrumentation
Bulgarian Academy of Sciences
Bloc II Acad. Bonchev Str., 1113-Sofia
BULGARIA
nshivarov@code.bg

Abstract: - This paper proposes a framework for improving design of MAT Mobile Assistive Technology vehicles by applying the model driven generative domain engineering method to develop self-organizing architectural solutions for mobile vehicles. Developments were carried out by using the Rhapsody™ tool, a MDD Model Driven Development environment for embedded real-time systems based on the UML 2.0 standard. The MAT Mobile Assistive Technology device is treated here as a special type of mobile robot that eventually would become autonomous after the further integration of specific algorithms for obstacle avoidance, navigation and orientation and the design of accessible environments. The modeling framework is developed within the social model of disability. The aim of this unified modeling framework proposal is to allow the analysis and synthesis of assistive technology systems, as well as appropriate matching to potential end-users. Requirements definition and analysis is strongly taken into consideration. Agreed requirements are the basis of acceptance testing.

Key-Words: - modelling, assistive technologies, mobility, requirements definition and analysis, model driven development

1 Introduction
The position of disabled people has experienced significant changes in the last decade while new regulations, legislation and standards were adopted in order to increase the social integration and participation of the disabled people in every aspect of community activities. The social model of disability is the driving force of these changes and the tools of assistive technologies could enable the social inclusiveness [5]. In terms of the social model the aim of assistive technology is to overcome the gap between what a disabled individual intends to do and what the existing social infrastructure allows he or she to do [8]. Improving the interaction options between the end user and the device could further increase the quality of life resulting from assistive technology use. Therefore there is a need today for a common "language" between the agencies and professionals dealing with different aspects of social integration of disabled people.

There is a strong evidence today that models of development and evaluation of assistive technologies should be used in order to provide common definitions that would describe a unified framework for assistive technology system development [2]. Development and evaluation should be possible against quality of life measures, taking into consideration the social model of disability. The social integration of disabled people through different kinds of assistive technologies requires a human-centered design approach, where problem definition, functional specification and built and test activities are required [7]. The Human Activity Assistive Technology (HAAT) model [1] is an approach on developing a generic structure to be used for device analysis, synthesis and development, but not for matching the device to the person. The Comprehensive
Assistive Technology (CAT) model [5] is an extension to the approach considered in the HAAT model, and it consists on a certain number of attribute layers that describes relevant aspects of a disabled person, his or her environment and of the assistive technology system used to support his or her activities. The CAT model defines all assistive technology systems in terms of four components, that means context, person, activities and assistive technology itself [5]. The proposed framework for improving design of MAT vehicles by applying the model driven generative domain engineering method that is introduced in this paper will use the CAT model in order to develop generic specifications for assistive technology systems that could reduce the functional gap between the disabled people and his or her environment.

Mobility is the major fundamental activity category that will be further approached in this work. It involves special tasks associated with movement and travel, for example obstacle avoidance, navigation and orientation, navigation to a target (reaching the right destination). We are mainly approaching here the first two components of the assistive technology system attribute in the CAT model [5], that means activity specification (task specification and user requirements) and design issues, without technology selection.

In terms of the MDD Model Driven Development framework this translates to requirements analysis (analysis for domain and application through use cases, class and object diagrams, scenarios), design and architecture (subsystem design - structure views, packages - layers and dependencies, use case scenarios - sequence diagrams, object behavior - state machines) as well as animation and verification (sequence diagrams, state machines diagrams) [3].

This paper is organized as follows. Section 2 describes the MAT Mobile Assistive Technologies in the framework of mobile robotics development. The importance of using executable UML models in the early stages of design is emphasized. Section 3 introduces the main aspects of requirements analysis for the MAT design in the framework of MDD Model Driven Development. Section 4 introduces the main characteristics of an executable specification for the MAT design. The analysis for domain and application is carried out through use case analysis, class and object diagrams, and animated scenarios. Section 5 draws conclusions and highlights further direction of development.

2 Modeling Mobile Assistive Technology Systems

Mobility defines the activity of moving from one place to another. Mobility implies fulfillment of special tasks like obstacle avoidance, navigation and orientation, as well as reaching a desired (target) position in the (near) space. In terms of local and environmental context we include here indoors and outdoors, or both. For this kind of "engineered" environments, special navigation strategies that include the above mentioned tasks are already available, in the field of mobile robot research. Therefore, MAT Mobile Assistive Technologies can be developed in the framework of mobile robotics [2].

A MRP Mobile Robot Platform for MAT Mobile Assistive Technology implementation could be a multi-use mobile robot, developed in its basic configuration and having all the most important and vital components for its mobile functions (driving system, main control system, navigation system, locomotion system, communication system). In this respect, mobile robot engineering encompasses techniques from a wide variety of scientific and engineering domains and is constrained to components, architectures, middleware, hardware or software. Traditionally, there is no design strategy that accounts for the totality of the robot system. For example, figure 1 presents a hardware setup that we’ve built in order to test the design framework presented in this paper [2].

Fig. 1 The complete MRP setup for MAT design

However, some recent research studies proposed a unified design framework that takes into consideration the robotic system as a whole [4]. It approaches the creation of the framework using the UML terminology that would allow to clearly define the functionality of many common robot system components. It also accounts for driving requirements analysis, design and implementation of new robotic systems. Also such approaches provide the developer with a visual representation of the designed system, none of them can be translated into an executable specification.

The MAT design framework that is proposed in this work concentrates on early stages of design. Requirements capturing and analysis are naturally
integrated in further stages of design and the UML / SysML standards are used in the framework of MDD Model Driven Development, in order to obtain a high level of abstraction and a visual view on the design. In this respect, the designer is able to decouple design from realization and to focus on what the system should do, that means that the specification of system functionality can be separated from the specification of the implementation of that functionality on a specific MAT platform [6].

We use models at different levels of abstraction, and transformation between them, to drive a concrete application implementation. The UML design model is further translated into an executable specification which can be tested and visually verified early in the design process.

3 Requirements Analysis of the MAT

Requirements are the driving force that could help engineering practitioner to understand how a person actually interacts with technology. People with disabilities may have varying degrees of dysfunction in their sensory, cognitive, and functional abilities. The interface between a disabled person and the device that he or she normally uses have to be selected and adapted to these varying abilities in order to allow his or her effective interaction with the device.

Our MAT development methodology that takes into account requirements, design and implementation requires an iterative design process (Harmony/SE process) [6], a tool (Rhapsody™) and a modeling language (UML / SysML visual modeling standards).

The requirements model for the MAT should take into consideration the common interfaces between the components in the CAT model. Through interfaces the assistive technology device would receive inputs from a contextual situation, from the person itself, or from activity driving module (task specification), and from the environment itself (figure 2).

In this section we are mainly concerned with the conceptual design of the human device interface that would allow a dedicated analysis for a clear definition and integration of the user needs into MAT design.

In the requirements analysis phase, functional requirements were used and a model of requirements was implemented in order to visualize the taxonomy of requirements. Figure 3 presents the corresponding SysML requirements diagram for the MAT device that provides mobility (Rhapsody™ diagram).

Requirements Diagram – MAT User Interface

Fig. 3 A requirements diagram for the MAT device

From the point of view of the human – device interface definition, the following aspects should be taken into consideration (see also figure 3):

- the user should be able to select the operation mode manual / automated through the hardware interface (button or touch panel);
- the user should be able to provide manual commands through the interface (manual operation mode);
- the user should be able to configure the device for automated operation mode;
- the user should be able to select or build maps (for navigation);
- the user should be able to select navigation algorithm for the required mobility task;
- the user should be able to navigate when an algorithm is selected.

The working procedure concentrated mainly on creating a Platform Independent Model PIM of the...
structure and the functions of the system to be modeled, that means an UML model, without technical aspects. The main steps approached requirements capture and analysis (special tasks for a person with mobility disability), together with operator module design, controller module design and specification of the interface to the hardware components.

4 The Executable Specification of the MAT system

The high level architecture of the MAT system presents a way in which the components of the system could be organized (figure 4). The following structural levels were identified:

- PC layer. Includes the interface with the IO components (user interface, decision making, data storage, etc);
- Control layer. Includes the controller (control algorithms);
- Device layer. Includes interface to the hardware components (sensors and actuators).

The logical view of the MAT system can be constructed through an OMD Object Model Diagram that shows the static structure of the classes and instances of the MAT system and the relationships between them (figure 5). The Hardware package contains IMotion, a hardware interface class that describes the operations that can be called on the hardware. The Robot package contains the RobotController, a class that controls the MAT system (a generic system). Robot is a stereotyped package containing only references to model elements owned by another package and it is used to provide a “public view” of some of the contents of a package (interface to the user, control algorithms module, interface to the sensor systems, the interface to the driving module).

![Fig. 4 High Level Architecture of the MAT system](image)

![Fig. 5 Logical view of the MAT system – the global OMD diagram](image)

OMD diagrams can present different logical views of the system (domain diagram, as figure 5, graphical user interface, controller overview). Figures 6, 7 and 8 present the OMD diagrams associated to the graphical user interface.

![Fig. 6 GUI Display Overview – OMD Diagram](image)

![Fig. 7 GUI Application – OMD Diagram](image)

The analysis for domain and application was carried out through use case analysis, class and object diagrams, and scenarios. Models were executed and animation and verification was possible by means of sequence diagrams and statecharts.

Figure 9 presents the statechart associated to the RobotController class, were the two operating regimes (manual and automated) are defined.

Statecharts are associated to the reactive classes (figure 9). Reactive classes may receive events, triggers...
or operations, while non-reactive classes may receive only operations.

Fig. 8 GUI Overview – OMD Diagram

Fig. 9 Statechart of RobotController class

Figures 10, 11 and 12 present snapshots of an animated sequence diagram (the sequence of messages between the generic components in the PIM model of the MAT - the main classes of the system, FrontPanel, RobotController, DifferentialDrive and IRSensor). These messages are the key elements in this diagram, describing the interaction between objects in the system, or between the objects and the environment. A message could be an event, a trigger or an operation (class method). A message defines a special type of communication. A communication could be a signal, an operation, or a message of creation / destruction of an object. The events are asynchronous messages and there is a delay between the time of sending and receiving them (figure 11). The triggers are synchronous messages that are immediately received (figure 10). The example presented here describes the way to create an executable specification that allows testing of different functioning scenarios associated to certain operating regimes. The example depicts an exploration behavior associated to the obstacle avoidance and navigation abilities for a MAT system.

Fig. 10 Animated sequence diagram – the main classes of the MAT system

Fig. 11 Animated sequence diagram – move forward animation

Fig. 12 Animated sequence diagram – left obstacle detected

The exploration regime is available here in the automated operation mode. Exploration means a move
forward default movement, while sensors will automatically scan the environment. A left obstacle is detected and the robot is able to avoid it while continuing exploration forward. Figure 13 presents the statechart associated to the automated control regime that defines 4 functional states. The default state waits until a navigation algorithm is selected (navigation to target, exploration, navigation with obstacle avoidance). When a navigation command is selected, a transition to one of the following state is possible: GoalNavigation, Exploration, ObstacleAvoidance. The system will exit from this state when a stop command is received (while exploring), or when the navigation task is complete (while the vehicle is navigating to the target position). Figure 14 presents the statechart associated to the exploration behavior.

![Statechart for automated control regime](image1)

**Fig. 13** The statechart for the automated control regime

![Statechart for exploration behavior](image2)

**Fig. 14** The statechart for the exploration behavior

5 Conclusion

The main objective of this article is the introduction of a design framework to develop MAT. Significant characteristics of the MAT devices are presented, that require a human-centered design approach. In this respect we try to develop feasible solutions for MAT using a mobile robotics framework. These solutions are designed and tested through scenarios in the MDD framework. While the PIM model introduced here could only be used for requirements analysis and design, further work on this topic will approach the generation of a PSM model that could be implemented on a real mechanical platform whose architecture will be further defined (an MRP Mobile Robot Platform).

The modeling approach introduced in this paper intends to highlight the generic structure of a special type of assistive technology that improves mobility to special groups of disabled people. It could be used to define, capture and analyze requirements. An executable design specification is created and tested by using the Rhapsody™ tool, a MDD Model Driven Development environment for embedded real-time systems based on the UML 2.0 standard.

References:


