Model-Based Systems Engineering Tools
Devoloping the GUILTE System

Ana Luísa Ramos and José Vasconcelos Ferreira
Department of Economics, Management and Industrial Engineering, GOVCOPP Research Unit, University of Aveiro,
Campo de Santiago, 3810-193 Aveiro, Portugal

Keywords: GRAPHITE Matrices, GUILTE System, LITHE Methodology, Model-based Systems Engineering, Systems Engineering, Traffic & Environment.

Abstract: The GUILTE (Guiding Urban InteLligent Traffic & Environment) system is part of a large, complex, interdisciplinary, socio-technical domain at the agenda of world leaders, local authorities, academia, and society-at-large. The main purposes of the system are to provide an integrated development framework for the municipalities, and to support the operations of the urban traffic through Intelligent Transportation Systems, highlighting two fundamental aspects: the evaluation of the related environmental impacts, and the dissemination of information to the citizens, endorsing their involvement and participation. The development process of the GUILTE system was supported by a novel Model-Based Systems Engineering methodology, the LITHE (Agile Systems Modelling Engineering), which aims to lightening the complexity and burdensome of the existing methodologies by emphasizing agile principles such as continuous communication, feedback, stakeholders involvement, short iterations and rapid response, and also by a graphical tool, the GRAPHITE matrices, developed to support model-based cooperative development environments.

1 INTRODUCTION

The modern world is crowded of large, interdisciplinary, complex systems made of other systems, personnel, hardware, software, information, processes, and facilities. The Systems Engineering (SE) field proposes an integrated holistic approach to tackle these socio-technical systems that is crucial to take proper account of their multifaceted nature and numerous interrelationships, providing the means to enable their successful realization. The Model-Based Systems Engineering (MBSE) paradigm is an emerging approach in the SE field and can be described as the formalized application of modelling principles, methods, languages, and tools to the entire lifecycle of those systems, enhancing communications and knowledge capture, shared understanding, improved design precision and integrity, better development traceability, and reduced development risks (Friedenthal et al, 2008).

A modern MBSE environment involves several interacting elements (e.g., standards, methods, modelling tools, applications) that must work together to achieve a successful system which fulfils the stakeholders' expectations (Figure 1).

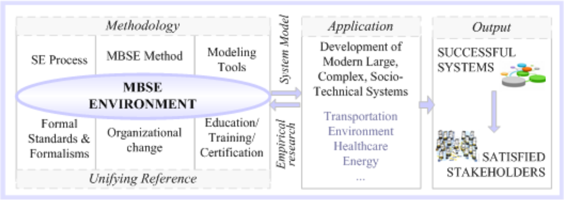

Figure 1: Integrated MBSE environment.

It is also characterized by a human-centric perspective where the human role is regarded as particularly important not only as a component of the system but also as an integrative/decision-maker element, enforcing the consideration of cognitive aspects and human systems integration (HSI) concerns throughout the engineering process and the system life cycle (Ramos et al, 2013).

This work intends to present some MBSE tools that can be used to support the agile development of large, complex, and multidisciplinary systems. The existing SE processes, methods, and tools to support a MBSE environment were evaluated regarding their adequacy to tackle the development of modern,

Luísa Ramos A. and Vasconcelos Ferreira J..
Model-Based Systems Engineering Tools - Devoloping the GUILTE System.
DOI: 10.5220/0004695501660173
In *Proceedings of the 2nd International Conference on Model-Driven Engineering and Software Development* (MODELSWARD-2014), pages 166-173
ISBN: 978-989-758-007-9
Copyright © 2014 SCITEPRESS (Science and Technology Publications, Lda.)

large, complex, interdisciplinary, socio technical systems; it is proposed an agile graphical tool to enhance MBSE methodologies and to facilitate the work of systems engineers in cooperative development environment and the GUILTE System (a case study) was designed using a Model-Based Systems Engineering approach supported by the appropriated agile tools.

2 MBSE FUNDAMENTAL TOOLS

A MBSE methodology is a set of related processes, methods, and tools used to support the discipline of SE in a model-based context. It implements a given process ("the what" activities are to be performed), which is supported by a given method ("the how" to execute), which is enhanced by a set of tools (the resources used to improve the efficiency of the tasks). The capabilities and limitations of the surrounding environment enable or disable the methodology and the resulting success or failure of system's development (Ramos et al, 2012). One of the primary artefacts of a MBSE methodology is the System Model. The modelling methodology must be chosen from the problem in analysis so, the systems engineer must be aware of the problem domain to understand which methodology will be more appropriate (Bahill and Szidarovsky, 2009).

The existing methodologies (in fact, informal methodological principles) for MBSE are well described in Ramos et al (2012), Estefan (2009) and Estefan (2008) and include: Harmony SE from IBM Telelogic, Object-Oriented Systems Engineering Method (OOSEM) from INCOSE, Rational Unified Process for Systems Engineering (RUP SE) from IBM, Vitech Model-Based Systems Engineering Methodology from Vitech Corporation, and Object Process Methodology (OPM) from Dov Dori. They are particularly focused on the implementation of the concept and development phases of the SE process, where they can provide considerable value-added.

According to Ramos and colleagues (2013), these methodologies are quite complex and intricate and reveal some shortages in terms of utilization of standard modelling languages (e.g., Systems Modeling Language – SysML) and incorporation of HSI concerns. To overcome these limitation they propose a new methodology (LITHE) with simple, lean, and customizable processes, methods, and tools to enable the widened utilization of MBSE practices.

2.1 Lithe Methodology

The LITHE (Agile Systems Modelling Engineering) methodology (Ramos et al, 2013), aims to be a more agile methodology than the existing ones, using a universal and intuitive SE process (a revised version of the well-known SIMILAR process (Bahill and Gissing, 1998), reducing the complexity and intricacy of the supporting method, emphasizing the agile human-centred development principles such as continuous communication, feedback, and stakeholders' involvement, short iterations and rapid response, and rousing the utilization of a coherent model of the system developed through the benchmark SE graphical modelling languages (SysML and OPM). As the authors state, LITHE is an agile methodology (Figure 2) for human-centric MBSE cooperative development environments.

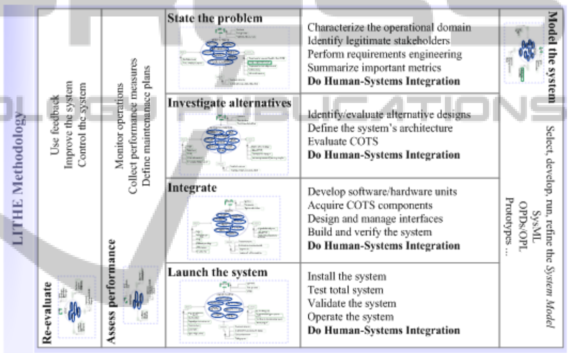

Figure 2: LITHE methodology for MBSE environments.

The inherent method provides a systematic approach to support the different stages of the process and includes, explicitly, the HSI considerations. Given that SE deals with socio technical systems, this aspect (highlighted in Figure 2) is crucial to attain a usable successful system. The activities within each function should be performed iteratively, with successive refinements like in a spiral development approach, with continuous support provided by the Assess performance and Re-evaluate transversal functions. The Model the system is the key function in a MBSE environment and holds up all the other functions (State the problem, Investigate alternatives, Integrate, Launch the system) by implementing the System Model. The LITHE methodology is simple and flexible enough to be customized for each concrete MBSE development scenario.

2.2 GRAPHITE Matrices

The methodology also embraces a novel supporting

SCITEPRESS SCIENCE AND TECHNOLOGY PUBLICATIONS

graphical tool, the GRAPHITE (GRAPHIcal Tools for stakEholders' interaction), which is a set of matrices with the rows standing for the type of stakeholders involved in the development process, the columns standing for the activity to perform, and the entries symbolizing the model(s) more suitable for that given circumstance. The matrices are an easy-to-use graphical representation for MBSE cooperative development environments where the systems engineer is the "glue" central person (Ramos et al, 2013; Ramos, 2011). Figure 3 depicts, as an example, the "STATE" matrix for the State the problem development stage.

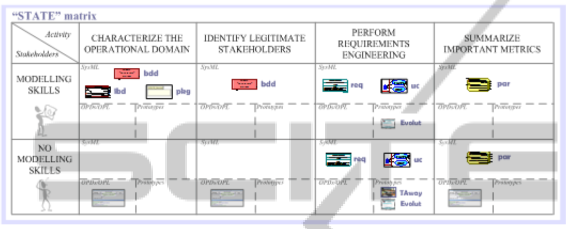

Figure 3: STATE matrix from GRAPHITE tool.

The models used in the GRAPHITE matrices correspond to the diagrams from the reference systems modelling languages and to prototype models, namely: i) SysML diagrams pkg (package), bdd (block definition), ibd (internal block), req (requirements), uc (use case), par (parametric), sd (sequence), act (activity), and stm (state machine), and allocation matrix (am) (Friedenthal et al, 2008), ii) OPDs (object-process diagrams) with the corresponding OPL (object-process language) (from OPM) and animated OPDs (running the OPCAT simulation engine) (Dori, 2002), and iii) prototypes (throw away and evolutionary).

The LITHE methodology and the supporting GRAPHITE matrices were applied to engineer the GUILTE system in order to evaluate its adequacy to be used in a modern MBSE context and to support the development of large, complex, interdisciplinary, socio technical systems. This development process is described in the following sections.

3 GUILTE SYSTEM: GENERAL DESCRIPTION

The GUILTE (Guiding Urban InteLligent Traffic & Environment) system is, simultaneously, a framework and a system. Is a framework since it is a logical structure or an organizational skeleton that enables the planning, development and deployment of traffic & environment intelligent operations by integrating and coordinating several "building blocks" that need to be interrelated to generate the emergent properties of the "whole". In this sense, the framework gives the "big picture" to guide the future developments. Is a system because it is a collection of different parts or elements (people, hardware, software, facilities, policies, documents) that together, by their interconnections, produce the system level results (characteristics, functions, behaviour, performance) achieving one or more stated purposes. Given that the system concept is more extensive and encloses the concept of framework, the GUILTE will be referred, from this point forward, as a system.

The main purposes of the GUILTE are to provide an integrated development framework for the municipalities, and to support the (short and real time) network operations of the urban traffic, through Intelligent Transportation Systems (ITS), highlighting two fundamental aspects: the evaluation of the related environmental impacts (in particular, the air pollution and the noise) and the dissemination of information to the citizens, endorsing their involvement. These purposes are obviously related with the high level complex and multifaceted challenge of developing sustainable urban transportation networks. The GUILTE aims to support the contemporary challenge of "green" management of the network with a qualitative improvement ("quality of roads" instead of "quantity of roads") based on timely information, personalized traffic & environment services, and high quality guidance provided to the entities moving on the roads. The networking technologies, the real time based interactive systems, and the service packaging are the perspectives that support this "quality" and are the basis of the proposed system.

The major elements of GUILTE are:

- Urban Traffic & Environment Real System: it refers to the actual system being analyzed or managed that is, in this case, the urban transport network and specifically, its traffic operations and related environmental impacts; the area under analysis can be a hotspot, a link, an area or the entire city;
- Data Acquisition: it is the element that gathers the different ways of attain data from the real system in order to feed the "brain" platform;
- Database, Modelling & Simulation Platform: it is the system's central part and is based on three fundamental pieces: a Geographical Information System for Transportation (GIS T), a traffic microsimulation tool, and a set of environmental

SCITEPRESS SCIENCE AND TECHNOLOGY PUBLICATIONS

models;

- Applications for Municipalities: it is the element that congregates and manages the relevant functions or applications that the municipalities need to carry out Urban Intelligent Traffic & Environment Operations; the municipality is taken as the major local authority in these matters;
- Sensing & Surveillance, Communications, Information & Control: it is a transversal element that includes technologies like, for example, sensors, GPS, wireless communications, mobile devices, and web, which can be grouped into the three clusters enumerated in its name and that are used by all the other elements; it also represents the "physical" support of ITS;
- Stakeholders: it is also a crosswise element that considers the different parties that have a right, a share, a claim, a need, or an expectation in the system or in part of it like, for example, local authorities, technicians, citizens, and academics; this element, which interacts with all the other elements of the GUILTE, highlights the Human factor in SE by modelling, explicitly, people as inherent parties of the system.

The main output of the GUILTE system is a collection of traffic & environment information services that are based on the information provided by the Applications for Municipalities element to the different Stakeholders (e.g., local governors, general public). These services should act like inputs when converted into actions or decisions (which, desirably, should be increasingly sustainable actions) including, for example, political resolutions, regulatory measures, modal and route choices, or even do nothing. These actions are executed on the Urban Traffic & Environment Real System by the Stakeholders thus, closing the loop. Each of the mentioned elements (the parts) serves a specific purpose but they must work together (the dependencies/interrelationships) to achieve the system's overall principle (the whole).

The GUILTE can be considered as a system-of-interest for the Systems Engineering field because it is a large system, a complex and interdisciplinary system, and a socio-technical system. It is also a network centric system since it is characterized by a complex set of people, devices, information, and services (the nodes) which are interconnected by a communications network to achieve optimal benefit of resources and a better synchronization of events and their consequences. Furthermore, "it comprises the so-called "Super Systems" such as Intelligent Transportation and Sustainable Environment (involve large interdisciplinary teams and considerable infrastructures which are globally connected), and considering an extensive set of " ilities" like flexibility, modularity, sustainability, real time capability, interoperability, expandability, reliability, usability, and delivery of value to society" (Ramos et al, 2013).

4 GUILTE SYSTEM: MODEL BASED DEVELOPMENT

Table 1 displays the mapping of the GUILTE system's development process to the SIMILAR process (the base process of the LITHE methodology). This mapping is a roadmap model that defines a set of iterative high level activities which are closely related with the system life cycle stages (Concept, Development, Production, Utilization, Support and Retirement). The art of systems engineers is to tailor this recipe to the concrete situation never overlooking the big picture.

Table 1: GUILTE system mapped to the SIMILAR process.

SIMILAR Process	GUILTE System
State the problem	Assess the need for the system and the relevant stakeholders, and do requirements engineering
Investigate alternatives	Design the architecture of the system
Model the system	Apply a Model-Based Systems Engineering methodology
Integrate	Integrate the different subsystems and interfaces and verify the system
Launch the system	Install, validate, operate and manage the system
Assess performance	Monitor operations, measure and evaluate the system
Re-evaluate	Use feedback, upgrade, enhance, extend, and dispose the system

As expected, the Model the system function and the fundamental role of the modern MBSE paradigm are emphasized. In fact, the Model function is transversal to the entire process being used to define the users needs, the system functionalities, the system architecture, the interfaces, etc.

The LITHE methodology was applied horizontally to different subsystems and components within a given element (subsystem 1, subsystem 2,…, component 1, component 2,…), and vertically to the different levels (system, elements, subsystems, components) of a one well defined system alternative. The following subsections will describe the key functions of the methodology but, due to space constraints, only the Investigate alternatives function will be explained and illustrated in detail.

4.1 State the Problem

The State the problem function of the methodology is supported by the following iterative method: characterize the operational domain, identify legitimate stakeholders, perform requirements

SCITEPRESS SCIENCE AND TECHNOLOGY PUBLICATIONS

engineering, summarize important metrics, and do Human Systems Integration. Its key objectives are to define the problem/opportunity to address, the high level capabilities of the system, and the general system's performance measures agreed by the various stakeholders. This function is supported by the transversal functions: Model the system, Assess performance and Re-evaluate.

Using the STATE matrix of the GRAPHITE tool the systems engineering team decided to use a SysML bdd and an OPD (a model more easily interpreted without formal modelling aptitudes) to communicate an initial high-level solution to discuss and to develop with the relevant stakeholders, a SysML bdd to identify legitimate stakeholders, several SysML req, SysML uc, and throwaway prototypes to formalize the requirements engineering stage, and SysML par to define measures of effectiveness for the system and constraints on the physical and performance properties.

This stage was developed incremental and iteratively until it was found an accepted solution (agreed by the key stakeholders) which was considered appropriate to guide the concept development: the Concept of Operations (ConOps) model.

4.2 Investigate Alternatives

The Investigate alternatives function of the methodology supported by the following incremental and iterative method: identify/evaluate alternative designs, define the system's architecture, evaluate COTS components, and do Human Systems Integration. Its key objectives are to explore alternative concepts for the GUILTE final solution and to design its baseline architecture that is, the system's elements, their characteristics, their arrangement, and their interactions. The architecture establishes a framework for the development of the system that will satisfy the requirements, and its definition is one of the systems engineer most critical and creative tasks. This function is supported by the transversal functions: Model the system, Assess performance and Re-evaluate.

4.2.1 Identify/Evaluate Alternative Designs

As suggested by the INVESTIGATE matrix of the GRAPHITE tool, the identification/evaluation of alternative designs was accomplished through a set of collaborating diagrams: SysML req (to define requirements and their relationships), SysML ibd (to clarify key actors and interaction flows), and SysML sd (to describe critical functionalities). It were also used several OPDs since OPM has a simulation engine that enables the time-dependent execution and testing of processes (particularly helpful to communicate with nontechnical stakeholders).

The following pictures illustrate some of these diagrams (the SysML diagrams were developed in the Artisan® software tool). Figure 4 depicts the SysML req for GUILTE system's requirements, Figure 5 shows a SysML sd for a critical functionality of the system, Figure 6 illustrates two snapshots for a SysML sd, and Figure 7 shows an OPD for the top-level functionalities of the system.

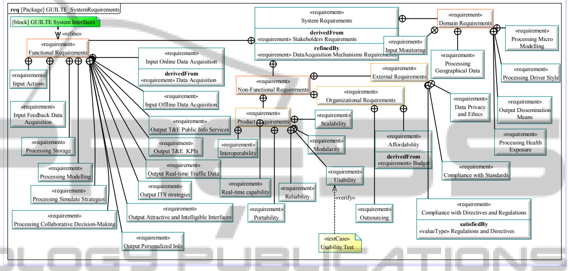

Figure 4: SysML req for GUILTE system's requirements.

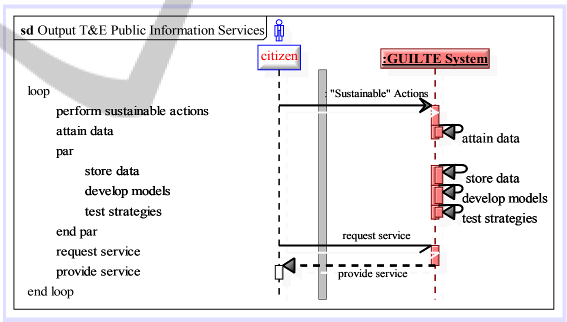

Figure 5: SysML sd for a top-level critical functionality.

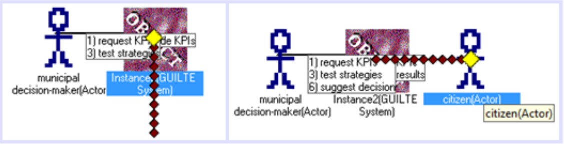

Figure 6: Animation snapshots for a SysML sd.

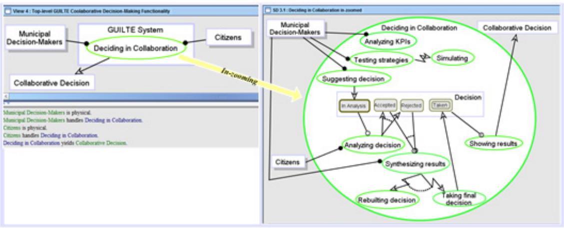

Figure 7: OPD for the top-level functionalities of the GUILTE system.

SCITEPRESS SCIENCE AND TECHNOLOGY PUBLICATIONS

The involvement of the relevant stakeholders in the requirements engineering process highlights, once more, the HSI concerns taken into account throughout the project. The agreed set of Systems Requirements (reflecting the ConOps) was the basis to start developing the architecture of the system.

4.2.2 Define GUILTE Architecture

The architecture of the system is a key element of modern SE providing conceptual integrity. The logical architecture and the physical architecture enable the definition of an integrated architecture (known as allocated architecture) that shall meet the system's requirements. The development of these architectures was accompanied by performance analysis studies and trade off decisions. Being the GUILTE a software intensive system, like the bulk part of the modern systems, the development of the architecture should take into account the principles underlying computer-based system architectures. The software architecture considered for the GUILTE was a distributed client-server where the system can be thought of as a collection of services, provided by the servers, and a set of clients that make use of those services.

The first diagrams to be used were SysML bdd to illustrate the main logical components and the main physical components (Figure 8) of the GUILTE.

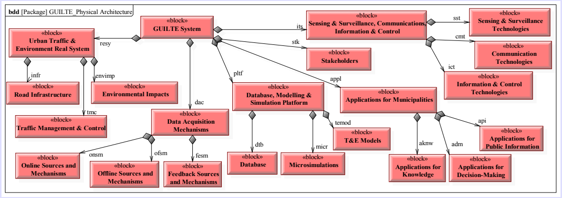

Figure 8: SysML bdd for the physical architecture of the GUILTE system.

Then, it were developed some SysML ibd to specify the connections between the different logical components and their interfaces. In order to specify the behaviour or the functional activity of the system and to provide details on the internal flow based behaviour of block operations, it were developed SysML act. These diagrams are similar to the traditional EFFBDs but provide additional capabilities such as the modelling of continuous flows and the establishment of relationships with structural elements of the system. The activities are based on token flow semantics (identical to Petri nets). Figure 9 shows a SysML act for the call action 'Store, model and simulate'.

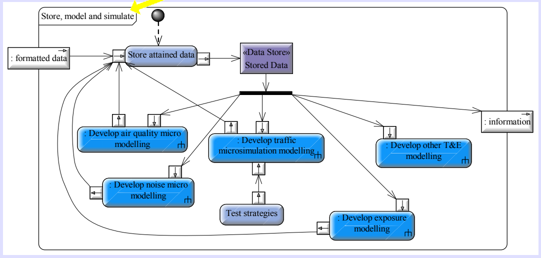

Figure 9: SysML act for the physical architecture of the GUILTE system.

The integrated Allocated Architecture involves the mapping of functions and requirements to resources. This allocation can be modelled, in SysML, through several mechanisms known as cross cutting constructs, and can be depicted graphically (Figure 10) or in a tabular format.

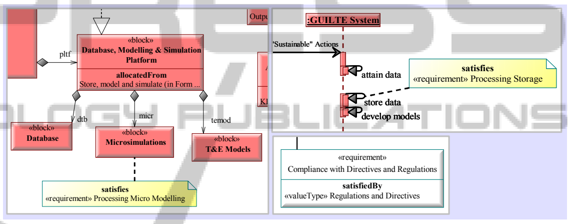

Figure 10: System requirements allocation to model elements of GUILTE.

For this particular stage, the representations through OPM are not as expressive and detailed as the ones provided by SysML.

4.2.3 Evaluate COTS Components

The evaluation of COTS components was accomplished when the physical architecture was instantiated. The GUILTE embraces several COTS components that were instantiated and selected through SysML bdd (Figure 11) and par diagrams.

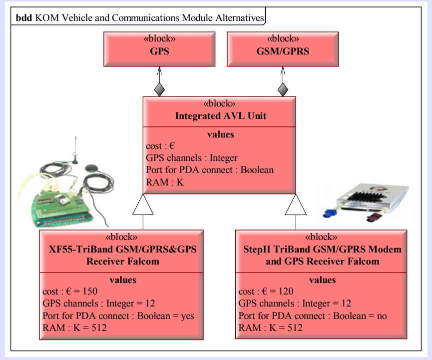

Figure 11: SysML bdd for the instantiation of a physical resource.

SCITEPRESS SCIENCE AND TECHNOLOGY PUBLICATIONS

Several other components of the system were developed internally and, in this case, the prototypes (as suggested by the corresponding GRAPHITE matrix) were also a working valuable tool to interact with the different stakeholders. Figure 12 shows maps developed in the ArcMap application (from ArcGIS® software tool, from ESRI) belonging to the prototype GeoMoving. This evolutionary prototype is a geodatabase schema for a Geographical Information System that will help the municipalities to organize their geographical information and to create a complete traffic & environment data repository.

Figure 12: Thematic maps from GeoMoving prototype.

The Investigate alternatives function outputs "the best" architecture for the system, and the associated specifications to develop the hardware and software components, during the integration of the system.

4.3 Integrate

The Integrate function of the methodology is supported by the following method: develop software/hardware units, acquire COTS components, design and manage interfaces, build and verify the system, and do Human Systems Integration. Its key objectives are to bring things together so they work as a whole (system) and produce emergent behaviour. This function is supported by the transversal functions: Model the system, Assess performance and Re-evaluate. This function is intimately related with the previous ones since a MBSE approach implies an integrated design process with interactions prominence. For example, the establishment of Integration Requirements, in the early stages of the development process, is a critical aspect to ensure successful holistic end-to-end operation.

Using the INTEGRATE matrix of the GRAPHITE tool the systems engineering team decided to use SysML bdd, SysML req, SysML us, and evolutionary prototypes to develop hardware/software units. To design and manage interfaces between the different components of the system and between other external systems, SysML ibd were very effective. The SysML dedicated stereotype «testCase» and the SysML sd were very useful to verify the system, elements, subsystems, components and ensure that they were built right. The HSI concerns were considered throughout the system's development process but, they were critical in this Integrate function where the interfaces man-machine were completed. After this stage, the system was able to be launched in its real operational environment.

4.4 Launch the System

The Launch the system function of the LITHE methodology is supported by the following method: install the system, do Human Systems Integration, test total system, validate the system, and operate the system. Its key objectives are to install the developed system in its operational environment and to guarantee that the system is producing the desired outputs being working according to the key stakeholders' requirements (ConOps). This function is supported by the transversal functions: Model the system, Assess performance and Re-evaluate.

Due to budget constraints, the GUILTE system is being installed in stages (in some Portuguese municipalities), with progressive association of components under the control of the defined system's architecture. The HSI concerns, present throughout the development process, are involving the definition of new jobs, the establishment of new working processes and training schemes in order to guarantee that the system will work properly. The operation of the system corresponds to its utilization under ordinary conditions and according to the stakeholders' needs, producing the desired outputs. The supporting Assess performance and Re-evaluate functions will ensure continuous system's monitoring, control, maintenance, preservation, improvement, and, eventually, system's disposal.

5 CONCLUSIONS

The LITHE methodology encloses a process, a method, and a set of tools that, due to their simplicity, flexibility, and compliance with standards, are believed to be customizable to each situation thus, allowing the systems engineer to put creativity into the process and introduce his/her unique perspective. The methodology was applied to the development of the GUILTE system, revealing an effortless utilization and great flexibility to use the functions, the activities, and the tools when needed and how intended. The emphasis on the

SCITEPRESS
SCIENCE AND TECHNOLOGY PUBLICATIONS

continuous and concurrent functions Model the system, Assess performance, and Re-evaluate was critical to ensure constant communication with the stakeholders, intense involvement and feedback, and rapid response. The Human Systems Integration concerns had guaranteed the permanent consideration of the human element, as well as the synergetic interaction between man-machine. The GUILTE is not completed (in full effective operation) and consequently, it cannot be concluded that the LITHE methodology had supported the development of a "successful system" but the intermediate deliveries are fulfilling stakeholders' expectations (these stakeholders have been involved during all the design and development stages of the system).

As suggested by the supporting GRAPHITE tool, the benchmark SE modelling languages, SysML and OPDs/OPL, were used to develop a coherent and integrated model of the system, along with prototypes. The utilization of graphical models had incited intense interactions and efficient communication with stakeholders. The synergies between the different types of models provided considerable value-added to the model-based engineering process.

The LITHE methodology and the GRAPHITE tool are examples of effective MBSE tools that can be used to develop large, complex, socio technical systems in agile modern environments, facilitating the work of systems engineers. This area offers significant research opportunities since the existing MBSE tools are still immature and require a proof of value in real world contexts.

REFERENCES

Bahill, A. and Guissing, B., 1998. *Re-evaluating systems engineering concepts using systems thinking*, IEEE Transactions on Systems, Man, and Cybernetics: Part C – Applications and Reviews, vol. 28, no. 4, pp. 516-527.

Bahill, A. and Szidarovsky, F., 2009. *Comparison of dynamic system modelling methods*, Systems Engineering, vol. 12, no. 3, pp. 183–200.

Dori, D., 2002. *Object Process Methodology: A Holistic Systems Paradigm*. New York: Springer.

Estefan, J., 2008. *Survey of Model-Based Systems Engineering (MBSE) Methodologies*. INCOSE Technical Document INCOSE-TD-2007-003-01, INCOSE, Pasadena, CA.

Estefan, J., 2009. *MBSE methodology survey*, INSIGHT-INCOSE Journal, vol. 12, no. 4, pp. 16–18.

Friedenthal, S., Moore, A. and Steiner, R., 2008. *A Practical Guide to SysML, The Systems Modeling Language*. Burlington, MA: OMG Press, 2008.

Ramos, A., 2011. *Model-Based Systems Engineering: A System for Traffic & Environment*, PhD Thesis, University of Aveiro, Portugal.

Ramos, A., Ferreira, J. and Barceló, J., 2012. *Model-based systems engineering: An emerging approach for modern systems*, IEEE Transactions on Systems, Man, and Cybernetics: Part C – Applications and Reviews, vol. 42, no. 1, pp. 101–111.

Ramos, A., Ferreira, J. and Barceló, J., 2013. *LITHE: an agile methodology for human-centric model-based systems engineering*, IEEE Transactions on Systems, Man, and Cybernetics: Systems, vol. 43, no. 3, pp. 504–521.