Towards a Service-Oriented Architecture framework for educational serious games

Maira B. Carvalho*,†, Francesco Bellotti†, Jun Hu*, Jannicke Baalsrud Hauge‡,
Riccardo Berta†, Alessandro De Gloria† and Matthias Rauterberg*
*Industrial Design, Eindhoven University of Technology, Netherlands
†ELIOS Lab, DITEN, University of Genoa, Italy
‡Bremer Institut für Produktion und Logistik GmbH, University of Bremen, Germany

***Abstract*—Producing educational serious games can be costly and time-consuming. The Service-Oriented Architecture (SOA) approach of software development can offer a solution to reduce costs and foment serious games development. In this work, we apply a model called Activity Theory-based Model of Serious Games (ATMSG) for identifying existing relevant components that can be reused for different educational serious games. We apply the derived structure to classify the elements of an existing game and to identify how it can be refactored and expanded following the SOA paradigm.**

***Index Terms*—Service-oriented architecture, serious games, educational serious games, serious games development, software engineering**

I. Introduction

Educational serious games are designed to meet specific learning requirement. Consequently, such games are typically conceived as one-of-a-kind products fully customized according to the clients' requirements. Thus, main barriers for large scale deployment of serious games are the high production costs, challenging and time-consuming production process, and low reusability of the final product and its components [1]. Based upon well proven concepts of using Service-Oriented Architectures (SOA) for increasing the reusability by maintaining a high customization degree, it has been proposed that SOA can also benefit serious games development by reducing costs and time to market, while at the same time allowing customization in a relatively easy and reconfigurable way [2].

While there is a growing number of services for serious games [3], we are still lacking structures identifying relevant and usable serious games elements and how to interconnect these elements in the game. Consequently, if a group of serious game developers wants to take advantage of such services, there are questions on how to identify components suitable to be converted to services, taking into account that these services should have a high degree of reusability to serve several games.

The main objective of this paper is therefore to identify candidate serious games components that could be developed as services for games within different genres and domains. For that end, we apply a newly developed model which connects different components at different levels within a game, called Activity Theory-based Model of Serious Games (ATMSG) [4]. To show the suitability of our analysis to the development of educational serious games in practice, we apply our results to refactor and propose the expansion of an existing serious game to further exploit the SOA paradigm.

A. Related work

The adoption of SOA for serious games is currently limited [2], but there are examples of using SOA in entertainment games, particularly online multiplayer games, upon which we can draw, as the examples below show.

Houten and Jacobs [5] present an architecture for distributed multiplayer simulation games. However, even though originally an entertainment game, it can be used for training and learning purposes. Nevertheless, the architecture is restricted to the genre of simulation games; furthermore, the architecture does not explicitly support an environment in which several games share the same resources. Shaikh, Sahu, Rosu, *et al.* [6] describe the design of an on-demand service platform to enable sharing resources across online games. Their architectural solution focuses on solving problems of scaling the infrastructure in response to players' demand in massively multiplayer online games. BinSubaih and Maddock [7] use the SOA design philosophy to enable game portability across different game engines, with the objective of removing the strict dependency of a game to the engine underneath it. While not strictly a SOA, the work highlights well the benefits of separation of concepts and reusability in game design.

The previous mentioned works show the benefits of SOA for game development. However, they also show that there is still room for improvement, since what has been proposed so far is either restricted to a specific genre of games, or it solves one single aspect of architectural decisions in service-based game development.

To expand these works also for the development of educational serious games, we need to explicitly identify how learning elements can be incorporated in such architectures. This has to be done in a way that allows for reusability while still supporting the specific learning goals of the game. To reflect upon which serious games components can be reused in service format, we applied a model called Activity Theory-based Model of Serious Games (ATMSG), which we describe in the next section.

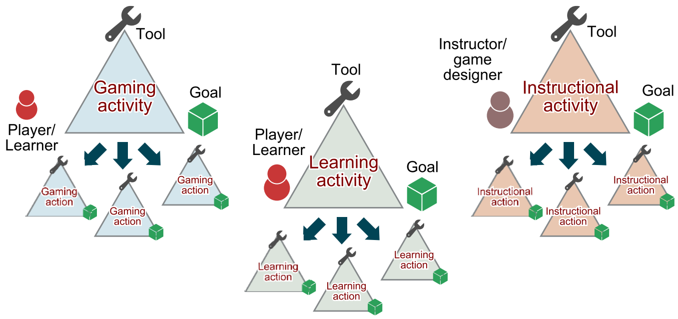

Fig. 1. Each activity is formed by a sequence of actions mediated by tools, with specific goals

B. The Activity Theory-based Model of SGs

The Activity Theory-based Model of Serious Games (ATMSG) [4] uses activity theory [8] to delineate a model that represents several different low-level components of an educational serious game as the game unfolds, and how these components are connected to the educational and entertainment high-level objectives of the game. In ATMSG, educational serious games are seen as used in the context of four activities: the gaming activity, the learning activity, the intrinsic instructional activity (performed inside the game) and the extrinsic instructional activity (performed by the instructor outside of the game).

In the model, these activities are further decomposed into a sequence of actions mediated by tools with specific goals, giving us a set of categories (Figure 1). These categories form the basis for a taxonomy of serious games components, in which existing taxonomies of learning, instruction, games and serious games are reorganized according to the ATMSG model [4]. Serious game components are understood as the pieces of an educational serious game — its actions, tools and goals — that constitute the gameplay over time, e.g. characters, tokens, tips, help messages, challenges, graphics, feedback mechanisms, assessment, etc.

The full taxonomy gave us an overlook of a large number of commonly found elements of serious games, which we could then analyze and classify in order to identify the possible components of a service-based serious game framework.

II. Identifying reusable serious games elements

A crucial initial step in implementing Service-Oriented Architectures is decomposing the business domain into its functional areas, subsystems and desired goals, in order to identify processes and high-level business use cases. These processes are then selected as possible candidates for being implemented as services. This step is common among several methodologies for the development of SOA solutions (e.g. SOMA [9], SDLM [10] and SOAF [11]). In this section, we therefore provide such an initial analysis of the business domain.

From ATMSG's taxonomy of serious games elements, we collected a number of relevant items. The criteria for the selection were (a) relevance for the effectiveness of educational serious games, and (b) possibility of reuse across different games and learning domains, at least within the same game genre.

These items are discussed below, grouped by the activities according to the ATMSG model.

A. Game activity

Game engines (such as Unity) have been successfully used to abstract a number of gaming elements, particularly those related to building a game world and its rules. Nevertheless, even when game engines are used, games are typically still built in a way that does not reveal the game's inner workings. It is thus necessary to create a way to expose in-game events to external modules in a reusable manner, translating those events into information that will be useful to external modules.

Because the majority of gaming elements are highly related to game genre and topic, we did not specify gaming elements as candidates for being converted as services, but instead treated most of the gaming actions as events to which external modules can listen. The exception are actions related to obtaining information, which are game- and genre-independent, and thus excellent candidates for abstraction.

Among gaming tools, elements that are independent of the genre are those related to goal metrics and feedback on the goals, such as achievements, performance scores, leaderboards. And, in a related fashion, among gaming goals, the elements related to competition based on performance can be abstracted, particularly if existing social networks are used to connect players to their peers.

B. Learning activity

Learning activity elements are highly dependent on game genre and on the specific implementation of the serious game.

This is particularly true in the case of learning actions. For example, actions such as memorizing, locating, classifying or assessing will have particular implementations depending on both the topic and on the game itself. Consequently, no reusable elements were selected from this list for inclusion as candidate services.

Among learning tools, however, some elements were considered general enough to be useful for a subset of game genres, even if they are not be relevant to all kind of games at all times. In particular, surveys and questionnaires — similarly to knowledge bases — are tools that can be easily abstracted and implemented separately from the game. We also identified that stand-alone modules to store and display media (e.g. audio, video and pictures) can be developed. Finally, student diaries can also be implemented in a way that can be game- and topic-independent.

Lastly, among learning goals, the most genre- and topic- independent element is "learning how to learn", or self-reflection on the learning process. A module to provide students with an overview of their own progress is another good candidate for a reusable service for serious games.

C. Instructional activity

Intrinsic instruction: The following intrinsic instructional actions were selected as candidate services: scaffolding, repetition, show similar problems and supporting recovery from errors. These elements are related to adapting the level of challenge to the player's current capabilities, which is an important factor for the efficacy of educational mediums [12].

Among intrinsic instructional tools, we highlighted quantitative assessment of performance, either using simple metrics (goal achievement, scores, etc.) or more complex methods such as the Competence-based Knowledge Space Theory [12].

From the list of intrinsic instructional goals, several items were selected: presenting the stimulus, providing feedback, assessing the performance, fostering confidence and providing satisfaction to the player. The items above relate to in-game assessment, feedback, and automatic adjustment of instruction (adaptivity).

Extrinsic instruction: Some elements in the extrinsic instruction list are similar to the ones already described in the intrinsic instructional activity. Here, however, the focus is on elements that a service-based framework could still incorporate to support the instructor in assessing and giving feedback to the student via the game.

Qualitative assessment of performance is an action that is both important and potentially relevant across serious games genres. It is the only item selected among the list of extrinsic instructional actions, since the other actions happen, by definition, outside of the game.

Among tools and goals, the selected items are also featured in the intrinsic instruction activity: performance measures, performance assessment and feedback. These elements were selected as candidates because they can be used to inform instructors about learners' performance, and then provide a way to incorporate the instructor's input back into the game.

In this section, we described our analysis of the educational serious game domain, in which we selected a list of elements that were considered both relevant and with a high potential for reuse. Table I summarizes the elements according to the activity to which they belong.

TABLE I
ELEMENTS OF THE ATMSG TAXONOMY THAT COULD BE IMPLEMENTED AS SERVICES, WITH THEIR IDENTIFIED FUNCTIONAL DOMAINS

Activity	**Element type**	**Items**	**Functional domains**
Gaming	Actions	Events listener	Game connectors
		Watch / Listen to / Read information	Information storage and retrieval
		Ask questions	Information storage and retrieval
	Tools	Social network score	Between-players interaction
		Leaderboards	Between-players interaction
	Goals	Competition	Between-players interaction
Learning	Actions	-	-
	Tools	Surveys, questionnaires	Student-instructor interaction
		Student diary	Student-instructor interaction
		Media assets (audios, films, graphics, etc.)	Information storage and retrieval
	Goals	Reflective observation	Feedback
		Learning how to learn	Feedback
Intrinsic instruction	Actions	Quantitative assessment	Assessment
		Scaffolding	Adaptivity
		Show similar problems	Adaptivity
		Support recovery from errors	Adaptivity
	Tools	Performance measurements	Assessment
	Goals	Assess performance	Assessment
		Provide feedback	Feedback
		Confidence	Assessment, Adaptivity
		Satisfaction	Assessment, Adaptivity
Extrinsic instruction	Actions	Qualitative assessment	Assessment
	Tools	Performance measures	Assessment
	Goals	Assess performance	Assessment
		Provide feedback	Feedback

III. REUSABLE COMPONENTS FOR A SERVICE-BASED ARCHITECTURE FOR SERIOUS GAMES

Once the relevant educational serious games components described in the ATMSG taxonomy were identified and collected, we regrouped them according to their domains and functionalities, so that we could identify the clusters of reusable components to be implemented as services in a SOA framework for serious games. The result of this grouping is shown in the column "Functional domains" in Table I. For each of these functional domains, we identified a list of candidate functionalities that could be implemented as services, which are represented in Figure 2 and further described below.

- **Between-players interaction:** Services to collect, display and compare scores, such as social leaderboards.
- **Student-instructor interaction:** Services for querying players/students, prompting for answers to questions (questions and surveys) or for reflections in their learning process (student diaries).
- **Information storage and retrieval:** For topics related to the knowledge inside the game (i.e. non-playing characters' knowledge about the game world) or about the learning domain itself, services to allow for management and use of this information can be useful. Descriptive metadata can facilitate resource discovery and subsequent reuse in different games. We also include in this category services that can connect to knowledge databases and convert this information to different formats that can be useful in the game (e.g. adapters for natural language

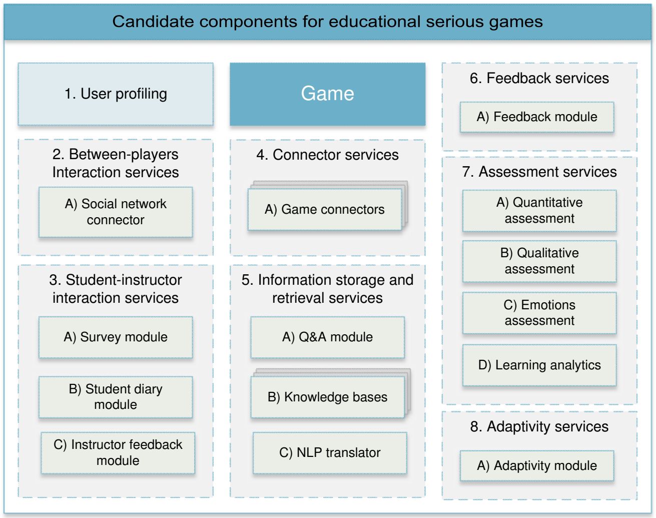

Fig. 2. The identified functional domains and their candidate components for implementation as services in a SOA framework for serious games

interaction or converters of knowledge bases to question and answer formats).

- **Feedback:** Feedback services provide a way to send the results of in-game assessment back to the game, in order to enable the provision of learning feedback, and to support the player's self-reflection on learning.
- **Assessment:** Assessment services can include modules for quantitative (automatic) and qualitative (instructor-provided) assessment, in addition to usage data that can help identify patterns of usage (learning analytics). It can also include modules for assessment of player/student's engagement, confidence and satisfaction.
- **Adaptivity:** Adaptivity services are responsible for consolidating information coming from several different assessments services, evaluating this information, making decisions on how the game should react, and serving this information back to the game.
- **Game connectors**: Game connector services provide adapter modules and data models that link external services to the game. This is possibly a game engine plug-in responsible for implementing trigger managers that detect important in-game events and forward messages to other services. These connectors will most likely be game- or at least genre-specific.
- **User profiling**: Although not directly derived from the taxonomy, a common user profile service is required in order to enable interaction, synchronization and persistent features across different games and learning settings.

The functional areas described above are an important asset to translate the theory, i.e. the ATMSG model, to the practical implementation of a SOA framework for serious games. Clustering elements allows us to identify relationships and to map information flows. This step will be invaluable to the development of a framework that enables reusability while still meeting the specific requirements of each serious game.

IV. Adopting SOA in Serious Game Development

In the previous section, we identified the key functionalities of reusable components that are relevant for serious games development. In order to deliver a proof of concept, we applied our findings to propose the refactoring of an existing game, called "The Journey" [2]. The game is a serious game for teaching basic elements of probability to high school and entry-level university students, in which players have to understand how to calculate the probabilities of events and use their knowledge to make the best possible decisions along the way. "The Journey" implements some basic aspects of a Service-Oriented Architecture to provide adaptation features for learning. With this analysis, we show how the game can be expanded to further exploit the SOA paradigm.

In the current architecture of the game, two main functionalities are realized by an external service (called "CbKST service"): the assessment of the player's competences, and the computation and suggestion of the next appropriate task for the player's current abilities. These functionalities correspond to the items "quantitative assessment" and "adaptivity module". In addition, the service also realizes one auxiliary function, which is managing the current learning session. This can be considered a very simplified form of user profiling that exists simply as a way to connect a user to the results of his or her assessments in the game.

The functional domains and components described in the previous section help us in analyzing the game to identify possibilities for refactoring. Among the elements in Figure 2, items 7-A and 8-A (quantitative assessment and adaptivity) are already services in the current implementation. Elements 1, 5-A, 5-B and 6-A (user profiling, Q&A module, knowledge base and feedback module), on the other hand, are current game functionalities, but not implemented as services. According to our analysis, these would be the best candidates for being converted to reusable services.

Another point to notice is that, in the current version of "The Journey", one external service (i.e. the "CbKST service") congregates two different functionalities, of two different (but related) domains: assessment and adaptivity. The problem with this combination is that the service, by treating both functionalities as one block, eliminates the possibility that other types of assessment (e.g. of the player's emotions) are incorporated in the decision model that provides the adaptation suggestions back to the game.

This case study shows us how migrating educational serious games to a SOA architecture based on the components proposed in this work can help developers in structuring their serious game in a way that promotes reusability and compositionality. For example, implementing the user profile as a service brings the immediate added benefit of decoupling the information about the player's achievements from the game itself, facilitating the use of this information by an instructor to provide feedback to the learner in an easier way. In addition, the use of knowledge bases for game tasks and learning content opens the possibility that other games and learning tools reuse the information. Finally, this structure helps developers in deciding the optimal level of granularity for the desired functionalities as to allow their easy integration within an overarching service architecture for serious games.

The downsides of applying this approach to serious game development involve the time and cost to refactor the code, and the increased complexity in how to test it. For this reason, it is crucial that the future service-oriented architecture framework for serious games includes not only the specifications of the services and interoperability standards, but also recommendations on how to minimize costs, and particularly how to perform quality assurance.

V. Conclusion and future work

In this paper, we proposed a list of reusable components that are relevant for the development of educational serious games within different genres and domains, and which will provide the basis for a service-oriented architecture (SOA) for serious games. In order to generate this list of components and functionalities, we utilized a model called Activity Theory-based Model of Serious Games (ATMSG), which helped us address the specific requirements of educational serious games that current SOA frameworks for entertainment games do not cover. To illustrate how the components can be used in practice, we applied the list of components to analyze the current architecture of "The Journey", a SOA-based adaptive game, and to propose the directions for its future improvement.

By determining elements and functionalities being commonly used across educational serious games, we advanced one further step in the process of creating a SOA framework to support reusability of serious game components. This framework will simplify the serious game development process and thus lead to reduction of costs and time to market. It will also encourage the use of interoperability standards and a consistent structure across game components, increasing the overall quality of the process and of the final product.

The next step of this work is service specification: from the list of candidate services, we will categorize them into a service hierarchy, then describe in more detail the components that need to be implemented, such as required data, rules, configurable profiles, messaging, event management, interoperability standards, etc. The final objective of this effort is to create a service framework that will support the establishment of a whole ecosystem of services for educational serious games. In addition, this will serve as a roadmap for service development, indicating which are the critical components that need to be implemented in order to benefit a higher number of serious game developers.

Acknowledgment

This work was supported in part by the Erasmus Mundus Joint Doctorate in Interactive and Cognitive Environments, which is funded by the EACEA Agency of the European Commission under EMJD ICE FPA n 2010-0012. We thank the colleague Erik van der Spek for comments that greatly contributed to this work.

References

[1] I. A. Stanescu, A. M. Stanescu, M. Moisescu, I. S. Sacala, A. Stefan, and J. Baalsrud Hauge, "Enabling interoperability between serious game and virtual engineering ecosystems," in *International Design Engineering Technical Conferences and Computers and Information in Engineering Conference (ASME)*, Buffalo, New York, USA, 2014. DOI: 10.1115/DETC2014-35418.

[2] M. B. Carvalho, F. Bellotti, R. Berta, A. D. Gloria, G. Gazzarata, J. Hu, and M. Kickmeier-Rust, "A case study on service-oriented architecture for serious games," *Entertainment Computing*, vol. 6, pp. 1–10, 2015. DOI: 10.1016/j.entcom.2014.11.001.

[3] Serious Games Society, *Serious games web services catalog*, 2013. [Online]. Available: http://services.seriousgamessociety.org/ (visited on 04/04/2014).

[4] M. B. Carvalho, F. Bellotti, R. Berta, A. De Gloria, C. Islas Sedano, J. Baalsrud Hauge, J. Hu, and M. Rauterberg, "An activity theory-based model for serious games analysis and conceptual design," *Computers & Education*, 2015, (Online first). DOI: 10.1016/j.compedu.2015.03.023.

[5] S.-P. A. van Houten and P. H. Jacobs, "An architecture for distributed simulation games," in *Proceedings of the 36th conference on Winter simulation*, Winter Simulation Conference, 2004, pp. 2081–2086.

[6] A. Shaikh, S. Sahu, M.-C. Rosu, M. Shea, and D. Saha, "On demand platform for online games," *IBM Systems Journal*, vol. 45, no. 1, pp. 7–19, 2006. DOI: 10.1147/sj.451.0007.

[7] A. BinSubaih and S. Maddock, "G-factor portability in game development using game engines," in *Proceedings of the 3rd International Conference on Games Research and Development*, 2007, pp. 163–170.

[8] Y. Engeström, *Learning by Expanding: An Activity-theoretical Approach to Developmental Research*. Orienta-Konsultit Oy, 1987.

[9] A. Arsanjani, S. Ghosh, A. Allam, T. Abdollah, S. Ganapathy, and K. Holley, "SOMA: a method for developing service-oriented solutions," *IBM Systems Journal*, vol. 47, no. 3, pp. 377–396, 2008. DOI: 10.1147/sj.473.0377.

[10] M. P. Papazoglou and W.-J. Van Den Heuvel, "Service-oriented design and development methodology," *International Journal of Web Engineering and Technology*, vol. 2, no. 4, pp. 412–442, 2006.

[11] A. Erradi, S. Anand, and N. Kulkarni, "SOAF: an architectural framework for service definition and realization," in *IEEE International Conference on Services Computing (SCC'06)*, 2006, pp. 151–158.

[12] M. D. Kickmeier-Rust and D. Albert, "Educationally adaptive: balancing serious games," *International Journal of Computer Science in Sport*, vol. 11, no. 1, pp. 1–10, 2012.