Motivation and Multimodal Interaction in Model-Driven Educational Game Design

Mladjan Jovanovic, Dusan Starcevic, Miroslav Minovic, and Velimir Stavljanin

Abstract—In this correspondence, we present an approach to identifying and constructing profiles of user interfaces for educational games. Our approach is based on framing games as educational tools that incorporate fun and learning through motivation as the key ingredient in the learning process and multimodal interaction as the medium for conveying educational material. The proposed solution formalizes the design process, describing educational games in terms of estimated effects that they produce on players. Building upon research on learning and motivation theory, we are connecting these effects with player learning preferences and motivation states. The essence of our solution is the educational game metamodel (EGM), which defines platform-independent educational game concepts. Using the EGM, we have explored novel design approaches for educational games. The metamodel can be used as a conceptual basis for creation of platform-independent educational games, allowing authoring for device and network independence.

Index Terms—Educational game, fun, learning, model-driven architecture, motivation, multimodal interaction.

I. INTRODUCTION

Digital games are becoming more and more considered to have a promising role in education processes. Features of games that could be applied to address the increasing demand for high-quality education are already identified as follows [1]: clear goals, lessons that can be practiced repeatedly until mastered, monitoring learner progress and adjusting instructions to learner level of mastery, closing the gap between what is learned and its use, motivation that encourages time on task, personalization of learning, and infinite patience. In addition, modern video games may develop higher order thinking skills, such as problem solving, strategic thinking, analysis, planning and executing, resource management, multitasking, and adapting to changing work scenarios. In order to exploit these attributes of games to improve learning outcomes, contextual bridging is needed between inherent motivational aspects of games and player learning abilities and preferences. This correspondence addresses the problem of closing the aforementioned gap using ideas from the multimodal human–computer interaction (HCI). We intend to suggest formal guidelines for designing games for education. This correspondence is inspired by the Object Management Group’s (OMG’s) model-driven architecture (MDA).

The structure of this correspondence is as follows. Section II presents the evolution of games for learning, the motivational aspects of learning and education, and the basic concepts of multimodal interaction from our point of interest. Section III gives an overview of the research field and some existing solutions. Section IV describes the research that brought us to the idea of building the framework for development of educational games. This leads us to propose, in Section V, a novel approach where we present a modeling framework based on our existing metamodel of multimodal HCI [2]. In Section VI, we demonstrate our approach by giving a design case-study example on the framework for educational games. Section VII concludes this correspondence.

II. PROBLEM BACKGROUND

In our approach, we are reusing ideas from multimodal user interfaces and psychological theories of motivation and learning in order to combine them in the design of educational games.

A. Evolution of Games for Learning

Video games can teach science and engineering better than lectures [3]. Specific kinds of instructions found in video games support effective learning paradigms and generally improve learning outcomes. Video games stimulate chemical changes in the brain that promote learning [4]. Playing video games stimulates dopamine release, which is a chemical precursor to the memory storage process. Time on task is another potential role of video games in education. Compelling video games that can deliver educational content would increase the time spent on learning.

However, research studies are needed to advance games for learning. These research studies need to determine the following: 1) how to design learning games to deliver positive learning outcomes; 2) how to develop tools to automate the process of developing games for learning; and 3) how to propose methods and techniques to assess the knowledge and skills that learners acquire from educational games.

B. Motivation in Education

Motivation concerns energy, direction, persistence, and equifinality—all aspects of activation and intention [5]. Motivation leads to the activation of efficient cognitive strategies for long-term memory issues like monitoring, elaborating, or organizing information. On the other hand, amotivation decreases memorization and personal development. Motivation appears to be a key asset to get actively involved, in time and mind, in the learning process. Naeve et al. [6] present an empirical pedagogical research that indicates how to effectively and efficiently apply the knowledge creation process by connecting it to several important psychological and social motivators for learning.

C. Multimodal HCI

Multimodal interaction is part of everyday human discourse: We speak, move, gesture, and shift our gaze in an effective flow of communication [2]. Enriching HCI with these elements of natural human behavior is the primary task of multimodal user interfaces. Oviatt et al. [7] gave a practical definition of multimodal systems, saying that they combine natural human input modalities—such as speech, pen, touch, hand gestures, eye gaze, and head and body movements—in a coordinated manner with multimedia system output. Computers and humans establish various communication channels over which they exchange messages with associated effects. Modalities process or produce these effects while various interaction constraints reduce or completely eliminate some of these effects.
III. PREVIOUS WORKS

In recent years, the educational games have become a hot topic, and many researchers have turned their interests toward them. Discovery-based learning is an approach to learning that emphasizes a learner’s active exploration of a subject. Immune Attack [8] is a PC-based single-player video game that combines a realistic 3-D depiction of biological structure and function with educational technologies for teaching immunology. The notion of using computer games for military training is almost irresistible [9]. Defense Advanced Research Project Agency (DARPA) developed DARPA’s universal persistent on-demand training WARS program (DARWARS) as a lightweight training simulation. In educational games, story can be used to structure the player experience to achieve educational objectives. One example is the Storytelling Alice programming environment [10] for creating interactive 3-D virtual worlds. Weaving the Semantic Web requires that humans contribute their labor and judgment for creating, extending, and updating formal knowledge structures. Hiding such tasks behind online multiplayer games presents the tasks as fun and intellectually challenging entertainment. The OntoGame series of games aims to cover the complete Semantic Web life cycle—the construction of ontologies, the semantic annotation of data, and the alignment of ontologies [11].

However, only limited research efforts are concerned with the environments, methods, and tools for the systematic development of educational games [12], [13]. A number of research studies concentrated on game development for supporting the particular science area while considering employing effective learning paradigms as completely separated from the actual game development process [8]–[10], [14], [15]. Principles of learning and motivation theories have to be followed a priori in design. User modeling principles [16] have to be incorporated into educational games, giving insight into learning outcomes and enabling their evaluation. Educational games can be seen as a form of adaptive hypermedia systems since these systems’ main characteristic is the ability of adaptation to the user characteristics and environment [17]. An enhanced adaptive hypermedia application model which exploits Semantic Web technologies has been introduced by Kravcik and Gasevic [18]. The model includes a user modeling layer. In this context, there are several approaches to user modeling. However, either of these approaches is based on application-specific user model needs, and more likely, they will not be suitable for other application domains. One attempt is to develop a common user model ontology that will be adopted by a wider community. This was the main purpose of the user modeling markup language, but there is not any evidence about common acceptance of this approach. The ultimate goal is utilizing educational games as adaptive environments with an enhanced user model that will integrate different user perspectives, such as knowledge, abilities, personal preferences and interests, browsing patterns, and cognitive and physical states.

IV. MOTIVATION FOR EDUCATIONAL GAME FRAMEWORK

Motivation for building the framework rose from the study that we have carried out [19]. The aim of the study was to identify and measure correlation between motivational factors [20], described in terms of cognitive effects produced in communication between players and games, and the overall quality of educational game scenarios. There are 58 participants used as specimens in the experiment. Students were organized into smaller groups (design teams) consisting of two or three members. Every team was given a set of questions covering a specific area of computer networks. Questions were to be modeled in the form of educational games. The assignment was to construct the specification/scenario for the game. The specification/scenario had to be detailed, including dialogs, scene descriptions, character descriptions, flow of communication, etc. In addition, every team had a mentor assigned who monitored the progress of scenario realization.

In the experiment, we conducted two independent evaluations. We used the semantic scale (between six and ten) to grade scenarios, considering the overall quality of the created educational game scenario. On the other side, three independent examiners performed scenario analysis by assessing the utilization level of the motivational effects. They used the scale from one to five for each effect. Finally, we did statistical analysis in order to identify the relation between applied effects and quality of produced educational game scenarios (Fig. 1).

The obtained correlation factor of 77.2% implies that the linear model is specified correctly. The positive slope coefficient (10.23) indicates that the increase in applied effects improves the overall educational game quality.

The given results show that the framework, which incorporates and formalizes motivational effects early in the design phase, can improve the development of educational games.

V. PROPOSED APPROACH

Our approach is inspired by the model-driven development, where software development’s primary focus and products are models rather than computer programs. In this way, it is possible to use concepts that are much less bound to underlying technology and are much closer to the problem domain [21].

MDA is an approach to system specification and interoperability based on the use of hierarchically organized formal models [22]. It uses a platform-independent base model (PIM) and one or more platform-specific models (PSMs), each describing how the base model is implemented on a different platform. In this way, the PIM is unaffected by the specifics of different implementation technologies, and it is not necessary to repeat the process of modeling an application or content each time a new technology comes along.

The proposed approach leads from an abstract view of the educational games’ concepts to a concrete implementation of the application by means of several design steps; high-level models are transformed into software components.

The essence of the proposed solution is the educational game metamodel (EGM) which introduces concepts and mechanisms from different domains into a single uniform view. Architecturally, the EGM is placed within the metamodel layer (Table I).

The views on educational games from different levels of abstraction can be derived by model transformations. The question of model transformations lies at the center of the MDA approach, and therefore,
TABLE I

<table>
<thead>
<tr>
<th>OMG MDA Level</th>
<th>Educational Game Metamodeling Architecture</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>M3 – Metametamodel</td>
<td>The Meta Object Facilities (MOF)</td>
<td>The MOF is an OMG standard that defines a common, abstract language for the specification of metamodels. MOF is a metametamodel - the model of the metamodel, sometimes also called an ontology.</td>
</tr>
<tr>
<td>M2 – Metamodels</td>
<td>The Educational Game Metamodel (EGM)</td>
<td>The Educational Game Metamodel provides a common and standardized language about phenomena domains relevant to the design of educational games. It is called a metamodel as it is the abstraction (model) of platform specific models.</td>
</tr>
<tr>
<td>M1 – Models</td>
<td>Platform-specific Schemas (XHTML, WML Schemas...)</td>
<td>Platform specific models of educational game content.</td>
</tr>
<tr>
<td>M0 – Objects, data</td>
<td>Content data (XHTML, WML files...)</td>
<td>Instances of platform specific models.</td>
</tr>
</tbody>
</table>

Fig. 2. MDA model transformations. For example, the first PSM could be an XHTML schema, whereas the second model could be some of the speech synthesis platforms such as a Java SAPI.

we use it as a basis for model-driven design of educational games. In MDA, PIMs are initially expressed in a platform-independent modeling language and are later translated to PSMs by mapping the PIMs to some implementation platform using formal rules. Fig. 2 shows the basic idea of model-driven educational game transformation. Bridge transformations work directly with platform-specific metamodels (schemas). In MDA, however, the greatest possibilities lie in more general approaches. The transformation of the game models can be specified by a set of rules defined in terms of the corresponding higher level metamodels. The transformation engine itself may be built on any suitable technology such as Extensible Stylesheet Language Transformation (XSLT) tools.

Building of (meta) model-driven transformation tools is much more efficient than bridge building as the number of modules is linearly dependent on the number of target platforms. For each new platform, it is necessary to develop only two new modules that map a new format into the platform-independent form and vice versa. On the other hand, developing bridges is a difficult and costly process, as bridges must have detailed knowledge of specific platforms. If we want to allow broader interoperability among various platforms, the number of bridges can be up to $N^2 - N$, where $N$ is the number of target platforms [23]. Furthermore, the processing logic of a particular bridge is not necessarily reusable in the construction of other bridges, which greatly increases the development cost. Mechanisms for defining model conversions are divided into two groups: mappings and transformations. Mappings are used between abstract layers, whereas transformations are used within them (Fig. 2). For example, it is possible to define mappings between PSMs, such as XHTML, and corresponding higher level metamodels, such as the textual platform metamodel. Conversely, it is possible to define transformations between various presentation platform metamodels at the same level of abstraction, such as between textual and text-to-speech platform metamodels. The generic mechanisms defined in the EGM can serve as a foundation for more elaborate solutions and are not the focus of this correspondence.

Our approach is based on standard technologies such as the Unified Modeling Language (UML) and Extensible Markup Language (XML), which are familiar to many software practitioners and are well supported by tools. Therefore, it is not necessary to develop complex solutions from scratch and is possible to reuse existing model-driven solutions and experiences from other domains [24].

In realization of the approach, we rely on existing UML modeling tools, XML parsers, and software frameworks, developing only code that extends, customizes, and connects those components according to common and standardized language defined in the EGM.

A. EGM

The EGM provides a common and standardized language for sharing and reusing knowledge about phenomena from various domains relevant to the design of educational games. Our aim was to identify a basic set of concepts from each of the domains and the relations among them. Speaking practically, we wanted to achieve the following:

1) a set of precisely defined terms and structured definitions of educational game domain concepts;
2) high expressiveness, enabling us to efficiently describe each aspect of educational games;
3) coherence and interoperability of resulting knowledge bases, using standard modeling and storage technologies;
4) scalability of metamodels providing us with means for definition at different levels of abstraction.

We have used UML to describe the metamodel. UML is an open standard and has standard mechanisms for defining extensions for specific applications contexts. We have used these mechanisms extensively in our approach. In addition, UML is widely adopted and is familiar to many software practitioners, widely taught in undergraduate courses, and supported by many books and training courses.

Defining educational game models requires a vocabulary of modeling primitives [25]. Therefore, our metamodel describes basic educational game concepts. Fig. 3 shows a simplified EGM.
therefore, we introduce the concept of EduGameKnowledge which is needed for managing complexities of the underlying knowledge. Educational content needs to be presented in some form to the user; expertise in science area is needed for conveying to players in the learning process. EduGameKnowledge introduced as a basis for defining other concepts of an educational game.

The metamodel’s main concept is EduGameElement which is introduced as a basis for defining other concepts of an educational game. EduGameKnowledge defines educational content that is aimed to convey to players in the learning process. Expertise in science area is needed for managing complexities of the underlying knowledge. Educational content needs to be presented in some form to the user; therefore, we introduce the concept of EduGameEngine. It describes the mechanism used to present knowledge, which, for example, might be the learning tool to generate and answer questions that guide a learner through the exploration and discovery of the required science area. The GameInteraction concept describes communication between players and games. The concept describes interaction at a high level of abstraction, regardless of specific manifestations. Interaction is established through multiple channels of communications [2]. EduGameLevel comprises previous modeling primitives in order to provide an inherent and natural evaluation mechanism in which levels are grades. It also allows creating games at multiple scales of knowledge and skills.

B. Game Interaction Metamodel

Interaction between players and games is realized using multiple channels of communication. In order to describe relevant concepts of communication between players and games, we propose a metamodel as shown in Fig. 4.

The main concept in the metamodel of communication between players and games is GameModality. It is defined as a form of interaction between players and games, which engages human capabilities to produce some effects on users. The presented concept of GameModality is derived from our existing metamodel of multimodal HCI [2]. Multimodal interaction can be established between multiple players (GamePlayer) and multiple game platforms (GamePlatform). GameContext defines a set of conditions and facts that are pertinent to a specific situation and can affect interaction between players and games. GameContext is additionally classified into the following:

1) physical context—defines environment conditions such as kind of space in which interaction is established, temperature value, luminance level, noise, and humidity;
2) situational context—defines a current situation in the environment from the point of view of the task that a user has to accomplish;
3) common knowledge—represents a set of facts that are understood by humans and computers;
4) social context—defines social environment of the interaction.

Each modality engages human capabilities, producing some effect on the user. Effects are classified into four main categories [2].

1) Sensory effects describe the human sensory apparatus’ processing of stimuli.
2) Perceptual effects result from the human perceptual system’s analysis of sensor data.
3) Motor effects describe human mechanical actions, such as head movement or pressure.
4) Cognitive effects occur at higher levels of human information processing and include memory, attention, and curiosity. In the design of educational games, these are the key assets.

In this way, user interface can be described in terms of messages (effects) that a designer sends to the user. Effects are interconnected (Fig. 7). For example, all perceptual effects result from sensory effects. These relationships enable designers predicting the result of using some effects.

C. Game Player Profile Metamodel

We have classified players/learners according to multiple intelligence theory [26].

The fact that some people impose multiple types of intelligences justifies the introduction of Complex type (Fig. 5). Gardner [26] states that most schools focus on the linguistic and logical/mathematical intelligences. In this way, individuals exposing gifts in the other intelligences—the artists, architects, musicians, naturalists, designers, dancers, therapists, and entrepreneurs—are learning confined by traditional educational processes. Unfortunately, many children who have these gifts do not receive much reinforcement for them in school. McCue [27] describes how computer games can provide a multi-intelligence approach to learning by exemplifying particular intelligences. In our approach, we are reusing multiple intelligence concepts in order to comprise users with diverse learning preferences inherently predefined by particular intelligences and identify the dominant content of learning interface according to intelligence type as follows [28]: words (linguistic intelligence), numbers or logic (logical/mathematical intelligence), pictures (spatial intelligence), music (musical intelligence), self-reflection (intrapersonal intelligence), a physical experience (bodily kinesthetic intelligence), a social experience (interpersonal intelligence), and an experience in the natural world (naturalist experience).

D. Game Player State Metamodel

In the field of psychology, cognitivism has renewed the study of the human mind on various aspects: language, emotions, memory, or motivation. If conceptual models and their associated semantics have bloomed during the second half of the 20th century, no global theory yet gathers the various motivational phenomena [29].

In this correspondence, we have chosen to look at motivation through Deci and Ryan’s self-determination theory (SDT) [5], since it is particularly expressive in the fields of education and gaming. Within SDT, the humans’ social development is driven by the satisfaction of innate psychological needs for competence, autonomy, and relatedness. This general framework of human motivation has
to be coupled to one’s innermost interests in order to explain self-determined behaviors encountered when people act with a full sense of volition and choice. High degrees of motivation require satisfaction of innate psychological needs and are directed toward what people find interesting or important. SDT is related to the following classification of motivation qualities:

1) **intrinsic motivation**—refers to doing an activity for the inherent satisfaction of the activity itself;
2) **extrinsic motivation**—refers to the performance of an activity due to factors that are external to the activity itself, like reward or threat;
3) **amotivation**—denotes the absence of motivation.

Motivation appears to be very effective in video games since they deal with fun, a potent source of intrinsic motivation [30].

In describing a player’s states, we are building on work by Denis and Jouvelot [20]. They have introduced an organization of fun factors in two poles—pleasure and desire—and their resulting ludic tension.

1) **Pleasure** comes from fantasy [31], control [32], power, creation, social interaction, immersion and comedy [33], direct system response, and experience of effectance [34].
2) **Desire** comes from challenge and curiosity [31], problem solving and competition [33], and competence [34].
3) **The ludic tension** comes from discovery [33], conflict, suspense and relief [34], and narration [35].

Looking at motivation through the impact of challenges and skills, the notion of parallelism between fun and learning is introduced: curiosity and proficiency as cognitive desire and pleasure (Fig. 6).

Ludic tension describes unstable whirlwind existing between pleasure and desire. Immersion is defined as the experience of feeling part of the synthetic experience and is closely related to the state of flow which occurs when a player is engaged in an activity (physical, mental, or both) at a level of immersion that causes the player to lose track of time and the outside world [36]. Intrinsic motivation encompasses flow, arousal, and control. Arousal and control are temporary regulation phases that reflect the dynamics of a learning curve, which has to be constantly renewed and balanced to maintain the player’s engagement. In this way, educational games can be seen as examples of user-centered designs that motivate through learning, arousing players’ interest (desire to act), and giving them the power to express themselves (pleasure to act).

Thus, the case for games in education considers fun as highly motivated learning, where evolving desires are nurtured by effective pleasure.

The overall goal of educational games is to draw the learner learning curve dynamics nearer to the zone of proximal development. The zone of proximal development is defined as “the distance between the actual development level as determined by independent problem solving and the level of potential development as determined through problem solving under adult guidance or in collaboration with more capable peers” [37].

According to previously described concepts, Fig. 7 shows a simplified UML description of an intrinsically motivated player’s states that are pertinent to the learning process.

Table II shows intrinsically motivated states associated with corresponding cognitive effects. The proposed classification of a learner’s states with associated messages that trigger the state entry gives the designers an opportunity to predict results of using some effects early in the design phase.

As pointed out in the previous section, GameModality engages human capabilities to produce effects on users. Concrete effect will cause transition of the player to the desired state. We defined input and output types of a GameModality using the computer as a reference point.

Fig. 8 shows a closer look at communication between players and games.
which defines all elements of one scene in educational games but without binding to a concrete user interface specification. Then, *GameOutputModality* produces concrete *InterfacePresentation* based on given *EduGameScene* and concrete *EduGamePlatform*. The selection of a concrete *InterfacePresentation* instance depends on the type of the *GamePlayer*. The player type can be determined at the beginning of the game and can be customized during the game, depending on a player’s actions. In order to identify the initial type, we can use psychological tests or try to capture it automatically, if possible, by analyzing a player’s actions produced during the game training phase.

The major benefits that the framework brings are adaptation of interface presentation to a player type and guidance of communication in terms of selected effects that will shift the player to a target state. A clear separation between concepts relating educational game logic and content on one side and flow of interaction on another gives us more flexibility in game design. Extraction of interaction capabilities into separated abstraction (GameInteraction) improves reusability, enabling to preserve game interaction while modifying educational content.

VI. DESIGN EXAMPLE

Table III gives a survey of possible mappings of some high-level cognitive concepts, described in Table II, into concrete multimodal primitives offered by visual or audio user interface platforms. A typical game enforces different communication modes (GameModalities), and each of them is engaged to some extent, since a typical game player profile is of complex type. Thus, every learner has his/her own proportion of intelligences, which further implies appropriate educational game interface. Extraction of high-level EGM concepts and defining model transformations give us an opportunity to specify game interfaces adaptable to concrete learner/user which will produce cognitive effects (messages) causing learner transition into the desired state.

Based on the mapping described previously, Table IV illustrates the transformations of an educational game’s high-level PIM fragment into textual, visual, and speech representations, respectively. The given example transforms an object of a strategic game, action named ATTACK, which denotes the fact that the player’s troupes are currently under attack. The action further employs cognitive message “Direct system response.” Three basic representations are shown: text, animated pictures, and speech. Platform-specific technologies for textual, visual, and speech representations are XHTML, SWIXml, and Microsoft Speech API (SAPI), respectively. Transformations are implemented using XSLT technology. In addition, the ATTACK action is a part of more complex concept SCENE, which describes a complete strategic game scenario. The given mappings, with supporting transformations, allow us to personalize the learning content to a specific learner profile to some extent.

High-level models may be used for automation of some phases of the design of educational games’ interfaces. We used the extension mechanisms (stereotypes) of the UML to describe these models. In order to validate the framework, we have been developing tools for generation of Java-based educational games’ interfaces. These tools take as an input the XML description of a high-level model exported from UML models, parse it, and produce Java code files with an abstract Java framework that represents a skeleton of a designed user interface. A designer can make use of generic mechanisms supported in the framework and extend this framework with platform-specific implementations of games’ interfaces.
TABLE IV
EXAMPLE OF MDA TRANSFORMATIONS FOR EFFECT DIRECT SYSTEM RESPONSE, FROM HIGH-LEVEL PLATFORM-INDEPENDENT GAME ACTION CONCEPT INTO PLATFORM-SPECIFIC GAME ACTION REALIZED WITH XHTML, SWIXML, AND MICROSOFT SAPI XML

<table>
<thead>
<tr>
<th>PIM game definition fragment</th>
<th>Animated pictures representation</th>
<th>Speech representation</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>XSL Transformations</strong></td>
<td>&lt;xml:template match=&quot;ACTION [name]&quot;&gt;</td>
<td>&lt;xml:template match=&quot;ACTION [lang]&quot;&gt;</td>
</tr>
<tr>
<td></td>
<td>&lt;panel</td>
<td>&lt;SAPI&gt;</td>
</tr>
<tr>
<td></td>
<td>constraints=&quot;BorderLayout.CENTER&quot;&gt;</td>
<td>&lt;LANG langid=&quot;409&quot;&gt;</td>
</tr>
<tr>
<td></td>
<td>&lt;html&gt;</td>
<td>&lt;label font=&quot;Georgia-BOLD-12&quot; foreground=&quot;blue&quot;&gt;</td>
</tr>
<tr>
<td></td>
<td>&lt;xsl:attribute name=&quot;text&quot;&gt;</td>
<td>&lt;xsl:attribute name=&quot;text&quot;&gt;</td>
</tr>
<tr>
<td></td>
<td>&lt;xsl:value-of select=&quot;desc&quot;/&gt;</td>
<td>&lt;xsl:value-of select=&quot;desc&quot;/&gt;</td>
</tr>
<tr>
<td></td>
<td>&lt;/xsl:template&gt;</td>
<td>&lt;/html&gt;</td>
</tr>
<tr>
<td></td>
<td>&lt;html&gt;</td>
<td>&lt;label font=&quot;Georgia-BOLD-12&quot; foreground=&quot;blue&quot;&gt;</td>
</tr>
<tr>
<td></td>
<td>&lt;xsl:attribute name=&quot;text&quot;&gt;</td>
<td>&lt;xsl:attribute name=&quot;text&quot;&gt;</td>
</tr>
<tr>
<td></td>
<td>&lt;xsl:value-of select=&quot;desc&quot;/&gt;</td>
<td>&lt;xsl:value-of select=&quot;desc&quot;/&gt;</td>
</tr>
<tr>
<td></td>
<td>&lt;/xsl:template&gt;</td>
<td>&lt;/html&gt;</td>
</tr>
<tr>
<td><strong>PSM XML fragment</strong></td>
<td>&lt;panel</td>
<td>&lt;label font=&quot;Georgia-BOLD-12&quot; foreground=&quot;blue&quot;&gt;</td>
</tr>
<tr>
<td></td>
<td>constraints=&quot;BorderLayout.CENTER&quot;&gt;</td>
<td>&lt;xsl:attribute name=&quot;text&quot;&gt;</td>
</tr>
<tr>
<td></td>
<td>&lt;label font=&quot;Georgia-BOLD-12&quot; foreground=&quot;blue&quot;&gt; &quot;Your army is under attack!&quot;/&gt;</td>
<td>&lt;xsl:value-of select=&quot;desc&quot;/&gt;</td>
</tr>
<tr>
<td></td>
<td>&lt;xsl:attribute name=&quot;text&quot;&gt;</td>
<td>&lt;/xsl:attribute-name=&quot;image&quot;&gt;</td>
</tr>
<tr>
<td></td>
<td>&lt;xsl:value-of select=&quot;desc&quot;/&gt;</td>
<td>&lt;/xsl:attribute-name=&quot;image&quot;&gt;</td>
</tr>
<tr>
<td></td>
<td>&lt;/xsl:template&gt;</td>
<td>&lt;/script&gt;</td>
</tr>
</tbody>
</table>

V-Strat is a pilot project educational game framework. It is an experimental prototype developed according to the proposed design issues for educational games. Fig. 9 shows an example of a generated educational game’s interface. Scientific knowledge facts are embedded into game rules (answering questions instead of throwing the dice). The game’s interface dominant content is aimed for complex intelligence type. A chat room emphasizes the game’s social component, enabling to take advantage of all existing communication types for collaborating, negotiating, plotting, and competing among players. V-Strat is built upon Java platform and is integrated into Moodle Courseware Management System—based on Linux/Apache/MySQL platform.

MDA is based on standard technologies, which are familiar to many software practitioners and are well supported by tools. Formalizing concepts of educational games using standard UML enables incorporating educational game development into standard software development techniques and methodologies.

In order to take into account the type of learner, game level mastery, domain knowledge characteristics, and environment, our method allows definition of different models of interfaces at different levels of abstraction. Models can be organized hierarchically and grouped according to different aspects. The models can be reused, which reduces the effort. In our experience, creation of the initial EGM is the most time-consuming effort.

The EGM can be viewed as an ontology that offers a uniform view on important relations among educational game concepts. The unified ontology of educational game concepts can provide the context where we could perceive many relations that are not always obvious. In this way, the EGM could be used as a basis for collaborative development of knowledge about phenomena in educational games [38]. Currently, the main obstacle for introducing ontology-based knowledge management applications into commercial environments is the effort needed for ontology modeling and metadata creation [39]. One approach to learning ontologies and creating metadata is to supplement the results of semiautomatic ontology-learning and metadata-generation techniques with information drawn from the context in which the user is working. This is an area for research both into formal mechanisms for establishing trust and into the human and psychological aspects of how we can use such mechanisms. This is where we see the potential role of our approach for the realization of the Semantic Web vision.

If an educational game is well suited to the values that it conveys, the player’s progress should reflect his development regarding the learning material. This way, games provide a natural and seamless evaluation mechanism in which levels are grades. This automatically removes the negative aspects of external rewards on intrinsic motivation observed
by Deci and Ryan [29], since they deviate the players’ focus from skill acquisition per se.

VII. CONCLUSION

This correspondence has described an approach to modeling educational games that incorporates principles of motivation theory as well as multimodal HCI. The key challenge is to increase players’ motivation for learning. The proposed modeling framework is aimed to design educational games that motivate users with diverse learning abilities and preferences to get actually involved in the learning process. This is achieved by classifying motivational states that are pertinent to the learning process with messages that cause transitions to these states and identifying game player profiles according to player learning abilities and preferences.

The model-driven approach and proposed design solutions may be very useful for designers and researchers in the field of games for learning. The unified metamodel of educational games can provide them with a context of educational game concepts where they can perceive many relations that are not always obvious. The metamodel could also be used as a basis for collaborative creation of broader knowledge about phenomena in the field of games for learning. This is the ultimate goal of building ontologies as the means mainly used for sharing a common understanding of the structure of information among human and software agents, so humans and computers can understand the models [40].

Our approach enables motivating players by emphasizing the games’ social component. However, the games should take user preferences into account to create a more enjoyable gaming experience. Because a shared representation—understandable by and agreed upon by multiple human actors—is at an ontology’s heart, such games also show a promising fit for the Semantic Web. We can even extend this to a combination of user input and machine learning, leading to a true combination of human and computer intelligences.

At the current stage of development, the EGM framework provides a set of model transformations that facilitate design activity at every stage, although some manual intervention by designers is always needed. Our future research agenda will concentrate upon empowering and incorporating our design abstractions in standard software development processes in order to improve and simplify design and evaluation mechanisms of educational games.

REFERENCES