A. Oulhaci, E. Tranvouez, S. Fournier, B. Espinasse, « A Multi-Agent System for Learner Assessment in Serious Games: application to learning processes in Crisis Management », 7th IEEE International Conference on Research Challenges in Information Science, IEEE-RCIS 2013, Paris, France, May 29-31, 2013.

A Multi-Agent System for Learner Assessment in Serious Games: Application to Learning processes in Crisis Management

M'hammed Ali Oulhaci, Erwan Tranvouez, Sébastien Fournier, Bernard Espinasse
Aix-Marseille Université (AMU), LSIS UMR CNRS 7296
38, Av. Normandie-Niemen 13397 Marseilles Cedex 20 FRANCE
Email: ali.oulhaci,erwan.tranvouez,sebastien.fournier,bernard.espinasse@lsis.org

***Abstract*—Serious games are more and more used for training in various domains, especially in crisis management domain. In the development of serious games in such domain, the learner assessment has an important place. Indeed, learner assessment is essential in monitoring and supporting learners, and that by analogy with the intelligent tutoring systems (ITS) which aim to individualisation of learning in virtual environment. In this paper, to enhance learning and pedagogy in virtual environment of collaborative serious games for crisis management, we propose a software solution based on a multi-agent system (MAS) supporting learner assessment. In the context of collaborative serious games for crisis management, this supported assessment has to consider individual and collective assessment. The software solution proposed is integrated in a serious game developed in the SIMFOR project dedicated to crisis management and an illustrative example is presented.**

***Keywords*-Learner assessment, multi-agent system, serious game, crisis management.**

I. Introduction

In France, crisis management is based on the ORSEC plan, ORSEC means "Organization of Civil Security Response". The plan is designed to mobilize and coordinate under the authority of the prefect, the actors of civil security beyond the prevalent level of response or daily services. The aim is to develop the preparedness of all actors, public or private, who may be involved in population protection. Each actor must take ownership of the tasks within its competence and transcribe them in its internal organization. Indeed, the only way to test these plans is to make exercises in real conditions, which can become very heavy in terms of organization and very expensive. To reduce the cost and saving time, computer tools are solicited like Serious Games (SG). SG provides a fun way to learn. With the digital age, many schools and organizations are using SG for training. Nevertheless, the learner assessment in the SG remains highly problematic [1]. In most cases, the design of serious games (or similar system), designers focus on the pedagogy and the adaptive aspect of the serious game, while the assessment is discussed in a simplified manner [2] or missing in some serious game [3].

Some people consider the term *"serious game"* as an oxymoron expression, because the two words are contradictory. Domain professionals define a SG as a game that focuses on education rather than entertainment [4]. There are other definitions in the computer field such as Zyda's definition [5] where a SG is a cerebral challenge, played with a computer which uses entertainment as an added value. There are also approaches from the field of psychology such as Tricot's definition [6] that focuses on the pedagogical scenario. In [7], Alvarez offers a unified definition of a serious game: *A computer application, aims to combine with consistency, both serious aspects such as non-exhaustive and non-exclusive, teaching, learning, communication, or the information, with playful springs from the video game, adding this association must be done by implementing a pedagogical scenario"*.

When designing a SG, a large part of resources is allocated to the fun aspect of the serious game (virtual environment, game mechanisms, virtual reality ...) while assessment and monitoring of learner inherits a small part of the resources [8], or ignored. This is due to the nature of SG, following the definition of a serious game. A serious game is foremost a video game designed to teach something [8]. In most SG, the assessment is done manually with a tutor, or via the learner himself (diagnostic assessment, section II).

The objective of the paper is to automate the evaluation (in the broad sense, including monitoring and support) learners in serious games. In the literature, there are tools that deal mainly with the assessment and support of learners as intelligent tutoring systems (ITS). ITS are a particular branch from the Technology Enhancing Learning (TEL) researchers community aiming the individualizing training. ITS can provide real-time support as a virtual tutor, also called pedagogical module. The virtual tutor monitor the learner during training, and it provides him the necessary support and corrects mistakes (depending on system configuration). The ITS also represents (explicitly or not) knowledge (declarative or procedural) from the domain under study [9] as well as knowledge to be acquired by the learner (its mental state)[10] during training session.

In this paper we present a multi-agent system (MAS) representing the different modules of an Intelligent Tutoring System (ITS) providing individual and collective assessment in a collaborative serious game. Combining serious game and ITS provide a fun aspect of training wile having a strong pedagogical support, adding to that an evaluation module for more flexibility and enhancing learning process in virtual environment.

Section II exposes our research problematic, which concerns learner assessment in SG. In section III, we briefly present SIMFOR project, a SG for training non professional for crisis management as well as the issues related to learners assessment in the SIMFOR project. In section IV, we present our solution to add player' assessment capabilities to SG based on the *"Evaluation Space"* concept. Then in section V, we detail the MAS architecture proposed to support learner assessment. This MAS is instantiated on an illustrative example in section VI. Finally, we conclude by drawing the future steps of our research.

II. Learner assessment in serious games

Learner assessment is an essential learning task for teachers and learners. There are several kinds of evaluation processes each characterised by specific objectives (and thus means) as well as time frame as summarized in table I [11]:

Table I
THE DIFFERENT KINDS OF ASSESSMENT.

Assessment	Objective	When
Prognostic	predict the level of the learner	before learning
Formative	inform and assess skill levels	just before learning
Trainer	inform and regulate the activity of the learner	during learning
Diagnostic	inform and enable learners to regulate their learning	during learning
Summative	certify the result of the learner	after learning

The work presented in this paper requires assessment during (trainer and diagnostic assessment) and after the training (summative assessment). The assessment during training aims to provide a real time support while after training assessment provide a diagnostic of learned skills and allow a post training debriefing between learners and monitors. This is typically characteristics of SG assessment needs.

Assessment in SG represent a specific challenge [12], which comes to turning a SG into an Intelligent Tutoring System (ITS). As one of ITS main purpose is assessing a learner level [13], we have analysed different works in SG and ITS literature [14][15]. We have retained three important requirements in the learners' assessment:

- Knowledge representation
- Inputs and learners outcomes
- Feedback strategy

The evaluation aims to certify the learning of the learner. This learning is to gain some knowledge or skill in a particular domain. As we evaluate skills and knowledge, we must represent this latter in a computer ways. Representation and manipulation of knowledge and skills constitutes in themselves a whole research discipline, with several modelling paradigm depending on the nature of the knowledge as well as the manipulation objectives [9]. In our specific research context, Tchétagni's work appeared particularly relevant. In [16], Tchétagni et al. present an hierarchical approach for knowledge representation for learners assessment. The hierarchical assessment consists in assessing the state of the learner's knowledge at various levels of granularity. We can have other knowledge representation like ontologies, in [17], Provitolo presents an ontology of the domain of risk and catastrophe. more complex with some semantic with ontologies for example. The Ontology can define concepts and relationships between the different domain models. We can also have a knowledge representation as a meta model as was the case in [18].

To assess learners, we must get his actions, and in general serious game, this is done in a virtual environment. The Input-Environment-Outcome model (IEO) [19] has been developed as a framework for assessment of higher education courses understanding level. The basis of this model is that the evaluation is not complete if the assessment does not include the learner input information, the learning environment and the result of learning. In the context of Serious Game, learner input are *players'* actions in the 3D environment as well as decision made through specific interfaces (input forms) which both constitute the learning environment, while results can be computed through simulation.

In [20] Schmidt et al. show that while learners immediate feedback help learners completing a training session significantly faster, allowing the player to somewhat waver by delaying feedback, learners show better retention of skills. Feedback mode can be adapted to the learner progression in a training scenario, based on his/her reaction speed or initial and current knowledge level.

To enhance assessment in SG, we have tried to combine different solutions from different disciplines. The SG come with a fun aspect for training [15] and we enhance the pedagogical aspect with the different modules of an Intelligent Tutoring System. We thus propose to add to a SG, an evaluation module that must compute real time evaluations and post training evaluations. The evaluation can be individual (for one learner) and collective (collaborative learning). We present our model in more details in section V. In the next section we present the SIMFOR project, a serious game for training non professional to crisis management, and expose the assessment needs raised by this project.

III. Improving assessment in serious games

In this section, we briefly present SIMFOR project, a SG for crisis management as well as the issues related to learners assessment in the SIMFOR project in particular the heterogeneous nature of the information or knowledge representation required for assessment.

A. SIMFOR project

SIMFOR (figure 1) is a serious game developed by SII[1] company in partnership with Pixxim[2] company, in response to serious gaming call for project launched by the French

[1]http://www.groupe-sii.com
[2]http://www.pixxim.fr

Secretary of State for Forward Planning and Development of the digital economy. SIMFOR provides a fun and original approach for learning risk management as a serious game. SIMFOR is adapted to actors' needs and enables learners to train for major risk management by integrating multi-stakeholder aspect. The project objective is to create a tool that provides to users a context of risk management in real-time and realistic in terms of environment, self-evolving scenarios and actors.

Figure 1. Screenshot from SIMFOR project

SIMFOR is a multi-player game and allows different people to learn skill (shared or specific) in the same game. This is possible because SIMFOR does not target the specialists in the field of risk management, but rather the non-professional. Managing a major crisis can mobilize several hundred stakeholders, from the regional Prefect in his office to the firefighter in the field. These stakeholders are required to communicate and work together in order to restore a normal situation.

Many works, in the ITS literature, addresses the learner support and assessment issues [21][22], but SIMFOR is a multi-actor game dealing with two types of evaluation: individual and collective. Solving the crisis requires the resolution of all procedures of the stakeholders, so individual evaluation can affect the collective evaluation, and the collective evaluation can affect the individual evaluation too. For example if a learner has successfully realized his procedures, but the main purpose was not reached (material and human loss for example), the learners must be evaluated on their individual and collective performance to infer the reason of failure (lack of communication, missing procedure of another learner, ...).

B. Heterogeneous assessment problematic

As a SG, SIMFOR aims at emerging players in a virtual world enabling them to pretend acting as they would (and should) do in a real emergency situation. Knowledge and skills involved in such situation are various in nature as well as in terms of evaluation means, but nonetheless must be all assessed in order to certify (or not) that players know their part of the job on which many lives may depend. To better understand the heterogeneous aspect of the assessment needs, let's consider a simplified example of emerging situation scenario. This scenario starts with a TDM (Transport of Dangerous Material) truck overturned after a traffic accident. The tank is damaged and hydrocarbon is spreading over the road. A witness to the accident gives the alarm by calling the CODIS (Departmental Center for Operational Fire and Rescue Services in french) which in turn must perform four missions consequently to the alert. First, CODIS has to send a fireman on the scene to retrieve information about the accident (*"send firefighter"*). Once information on the accident is received (transmitted by the firefighter in the ground) confirming an TDM accident has occurred, the CODIS must secondly gives instructions to an officer (firefighter) on the measures to be taken. Then thirdly, the CODIS must complete an information sheet on the disaster that passes later through a fax to the mayor, prefect and the sub-prefect (the sending order is not important). Finally the last mission is to inform the OCP officer (Operational Command Post) when it is sent by the prefect.

This incident scenario involves several actors (CODIS, Firefighter, ...) each with different procedures and knowledge to be certified as well as an interaction orchestration to be respected. As these "learning material" differs in nature and perimeter, each requires specific representation as well as manipulation in order to produce reliable assessment. For example, figure 2 illustrates through an UML activity diagram the CODIS procedural knowledge to be respected and defined by ordered missions to be performed by the actor playing the role of CODIS. The incident form completion mission itself requires precise description items of the incident (declarative knowledge). Additionally, interaction results can be assessed by the time spent between incident detection and its resolution as well as the final impact area (for ex. area destroyed by the incident). Each piece knowledge needed to be assessed may use different representation model as well as specific evaluation computation to produce a learning score validating or not the learning objective of one scenario.

Thus, players knowledge assessment requires to collect learners data from different sources and varies depending on the role incarnated in the scenario. For example, to assess the first action of the CODIS (phone), we need data from the pedagogical environment (3D environment accessible through Human-Machine Interface - HMI) to get the learner interactions with the software. We also need learner actions history to compare with what sequence of actions was expected as defined in the domain model (as defined in fig. 2). On a lower (software) level, these data are not from the same type. The data collected from HMI (such as waiting time or action execution time) are represented numerically , while data collected from the domain model and the learner model requires Artificial Intelligence (AI) representation models (for

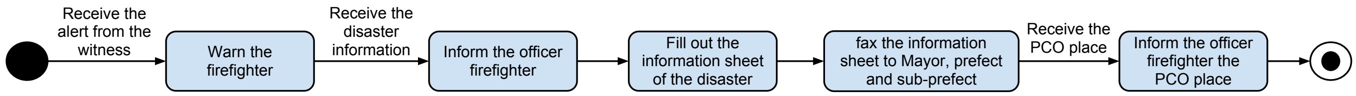

Figure 2. An activity diagram example of an actor procedures (CODIS)

example rule based). If knowledge modelling can be shared, its content itself depending on the players role and responsibility in the scenario.

In short, for the global assessment of the learner CODIS, we need data from his learner model to get his actions history and his knowledge, we need data from the domain model to identify the procedure to be followed, we need also learner's interactions to get information exchange and communication time and finally the information of his performance (environment information to get damage cost). In order to rationalise these data management we have have defined three data category : Behaviours data, Learners interactions and Simulation data.

In the next section we present the *Space* concept, our solution to manage the heterogeneous nature of assessment data, whereas section V propose a multi-agent architecture dealing with the Data source heterogeneity.

IV. Assessment in Serious Game : the evaluation space concept

This section develops a modelling contribution proposing to add players' assessment capabilities to Serious Game taking into account the various nature and origin of elements required to produce these assessments. These players' assessment capabilities are introduced in the SG, and an illustration is given in the SIMFOR project. We present also different sort of indicators to cover all kinds of assessment.

A. Evaluation space concept

As seen in the preceding section, adding assessment to a Serious Game (SG) implies dealing with knowledge, information or data produced/transformed continually or at the end of a game scenario. Each piece of information requires specific manipulation in order to extract evaluation material. A natural way to deal with the complexity of the management of these information (in a broad meaning) is to divide and to organise these information in homogeneous set where dedicated primitives can be used to produce evaluations. The *Evaluation Space* concept follows this approach by including all the elements required to produce assessments.

The concept of *"Space"* was inspired by the MASQ model (Multi-Agent Systems based on Quadrants) [23]. The MASQ model generalises previous work by having an agent behaviour projected on several (physical) Space in which agents are embodied through their preceptors and actuators, each responding and acting accordingly to the space's laws (physics law in relation to the space layout for example). We follow the same idea by considering a SG scenario through different view, each corresponding to a particular evaluation objective. An *"Evaluation Space"* thus gathers (homogeneous) information and primitives to manipulate these information in order to produce assessments. A space is defined by a set of the following element:

$$Space = \{Kw, I, M, AM\} \quad (1)$$

- *Knowledge representation model* (Kw): As seen in section III-B, there is different kind of knowledge, each based on a specific modelling paradigm (data modelling, rule based, bayesian networks,...). Thus, to ensure homogeneity each space will have a set of similar knowledge representation language. This homogeneity simplify interoperability and knowledge processing .
- *Indicators* (I): An indicator is usually defined as selected information associated with a phenomenon and designed to observe periodic changes by the light of objectives. Therefore, it is a quantitative data that characterizes an evolving situation (an action or consequences of an action) in order to evaluate and to compare their status [24]. Indicators will feed assessment process while the question on how to compute and maintain up to date these indicators will be dealt at a software level.
- *Metrics* (M): The metrics represents the methods and unit of measure used to compare expected results following actors' behaviour/decision to their actual doings. The metric is used to quantify the indicator, in other term, give a value to the indicator to compute an assessment.
- *Assessment model* (AM): There is different model of assessment, depending on the space and his knowledge representation. The assessment can relate to an action or a procedure or a global assessment (we will see the different kind of assessment in section V). An indicator computation relies on a specific assessment model according to its associated metric.

This definition of Evaluation Space can also be seen as a modelling guide for SG designers in order to help knowledge extraction from domain expert when defining domain model and evaluation expectations. This could open a way for a SG designing method, however not addressed in this paper. To better show how this concept can be instantiated in a real world case, the next section exposes its application to the SOMFOR project (see section III).

B. Illustration in the SIMFOR project

Section III-B has shown the heterogeneous nature of assessment in a SIMFOR game scenario. In order to better grasp its extent, we have defined a general domain model modelled as an ontology (Figure 3). The ontology is used to represent the different concepts of risk management in a generic and comprehensive way and describes the general concepts involved in a scenario. This ontology allow to represent different kinds of knowledge: the procedural knowledge (missions and actions), the environment entities (actors avatar, disaster, means, infrastructures, ...) and the different interaction between element (interactions). This analysis confirmed the heterogeneity of nature and source of the knowledge processed during the enactment of a game scenario.

In order to deal with the heterogeneity, three different *Evaluation Spaces* have been defined:

- *Behaviour Space*: The behaviour space includes actors actions and knowledges, as well as different information on skills and procedures to learn. This kind of "data" corresponds in an Intelligent Tutoring System to the domain model and the learner model. The domain model is static and is defined by a domain expert. The learner model is dynamic (evolves along the learner game experience) and is powered by learner actions performed during the game as well as knowledge acquired. Assessment will require reasoning on Intelligent Artificial related models.
- *Physic Space*: The physic space represent the virtual word, with actors avatar, means (cars, phone, fax, ...) and environment (building, road, trees, ...). This data can evolve in time like disaster state, actor avatar position, ... The virtual world is represented by the serious game 3D interface. The data handling is performed by the game simulation and the different mechanisms of the game engine (interaction, animation, ...). Assessment will consist in data aggregation through mathematical expressions.
- *Social Space*: The social space represent the social interaction between different actors. We represent the social space with graphs that records each interaction between actors and compute the strength of interaction between each actor and the network coupling. Assessment will be based on these measures.

Table II summarizes the different spaces used in Simfor project with the knowledge representation, the data sources and a few example of indicators.

C. Assessment Indicators

To cover all kinds of assessment (real time, final and collective assessment) we have defined different sorts of indicators.

1) Real time assessment: This assessment relates to a diagnostic (trainer) assessment evolving along the game scenario enactment. Real time assessment involves evaluation of action or mission.

- *Action assessment*: depending of kind action, the evaluation agent compute a score with the corresponding indicator (*action indicator*). For example, in the "Fax" action in 6, evaluation will check that no interlocutor has been forgotten.
- *Mission assessment*: players endorsing a role (CODIS, firefighter...) have to realize a set of missions, where a mission can be described as a set of (eventually) ordered actions responding to an objective (see figure 2). A mission assessment includes six different parameter, Table III shows the various indicators used as well as the formulas associated. The score of the mission is the mean of the six scores produced by these indicators. Indicators are computed either regularly (time or event directed), and are based on variables which themselves may require specific computing. For example, in the *Mission Preconditions* indicator, checking the respect of precondition may require inference rules to be run, whereas in the *Idle time* indicator, simpler data acquisition suffices.

2) Final Individual assessment: The final individual assessment corresponds to a summative evaluation that assesses and certifies the learning of the learner at the end of a game scenario. This evaluation compares the final state of the learner model (learner actions and knowledge) with the domain model and establish a diagnosis on the skills learned and skills that remains to be learned.

3) Collective assessment: Using the social space, an interaction graph can be built, representing the different communications and interactions between the different actors (learners and simulated actors). With this graph, we know who contacted who, when, and for how long time as well as the information exchanged. Combining information from different spaces, we have the possibility to infer a causality link between the actors procedures (Actor A was failed in his mission because actor B has not sent the correct information) and we thus obtain a complete and accurate assessment of all actors.

Global assessment, which can only be determined at the end of a game session, will integrate both individual and collective assessment. Certification of a learner skill or knowledge can be obtained through negotiation as in [25].

V. A MULTI-AGENT SYSTEM FOR LEARNER ASSESSMENT

The previous sections have detailed what information (in a broad sense) was required to produce the players assessment in the SIMFOR Serious Game (SG). In order to develop SIMFOR into a ITS SG, the software architecture must be adaptive to the different software components (SIMFOR 3D environment, Multi-agent Simulation, environment, simulating human interactions, Database tools, etc.) while keeping it open to the heterogeneous knowledge and assessment needs. The distributed and heterogeneous aspect of the problem has led us to opt for a multi-agent architecture for the implementation of the solution. An agent is a computer program located in a certain environment and able to perform actions autonomously in this environment to achieve its objective [26]. The multi-agent system allows a decentralized and distributed architecture and have social abilities (communication between agents). Our goal is to combine the different technologies of serious game (3D, game mechanism, animation, ...) and ITS (knowledge representation, pedagogy, ...) to get an optimal learning tools, thanks to a MAS architecture. These agents have for mission to collect learners' data, process and evaluate learners data and provide support to learners. To achieve these missions, we have defined five kind of agent:

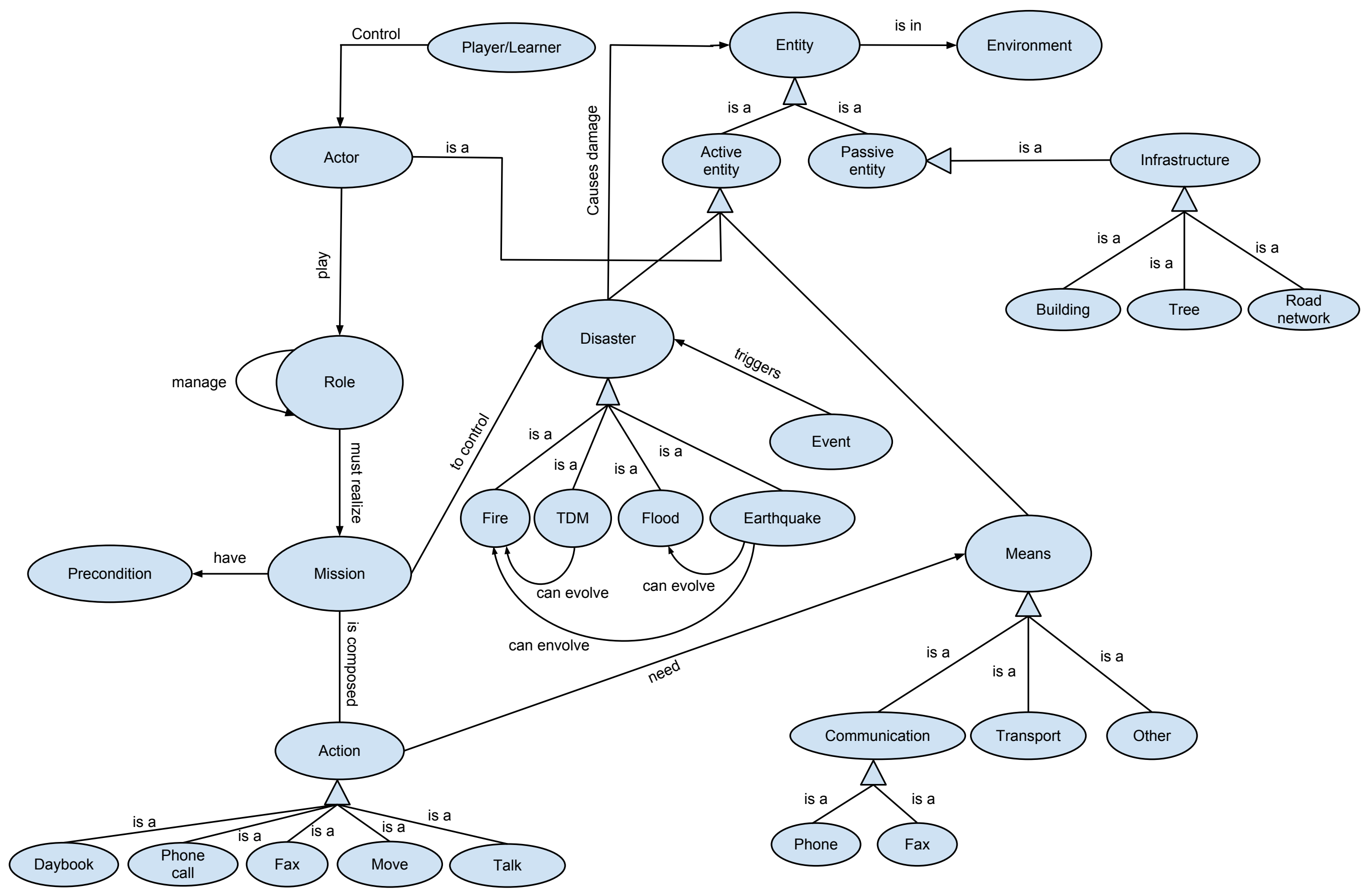

Figure 3. The ontology used in the SIMFOR project

Table II
CHARACTERISTICS OF THE DIFFERENT EVALUATION SPACES.

Space type	Knowledge representation	Example of indicators	Data sources
Physic space	Data structure	Damage cost, means used, ...	Simfor 3D environment
Social space	Interaction graph	communication time, information exchange	Simfor, MultiAgent System (Behaviours simulation)
Behaviours space	Knowledge Base, UML Activity Diagram	mission execution time, action efficiency	Domain model and Learner models (via agents)

- Data Source Agent: responsible of data collection.
- Indicator Agent: aggregate/transform data into indicator for learners assessment.
- Evaluation Agent: compute a learner evaluation
- Learner Agent: collect learner knowledge and actions and store them into a learner model.
- Pedagogical Agent: analyses the game situation and select the adequate strategy to help learners.

In addition, to simulate actor's avatar that are not played by learner, we have added a MAS that simulate human behaviour. For each ”non playing character” (NPC), we must associated an agent (called Game Agent) which is designed to reproduce the behaviour of the actor simulated (role incarnated). The Game Agents are based on BDI architecture[27] and we have brought some modification to the BDI architecture to provide more flexibilities to the scenario designer.

The global and collective assessment can be done by a collaborative process (or negotiation process) between the different evaluation agent.

To better understand how the agents enable the distributed assessment approach presented in section IV, figure 4 sums up the gradual transformation process from game data to learner assessment. These information are transformed and aggregated progressively but each step is ensured by dedicated agent. This figure thus illustrates the heterogeneous nature of these information as well as their source, and how agents fills in the gap from a knowledge and software level.

Figure 5 present a general architecture of the system. The use of agents allow a distributed task and make the system generic for other use of the system. For example, adding new

Table III
ASSESSMENT INDICATORS.

Indicator	Objective	Formula	Variable
Mission Preconditions	represents the respect of mission preconditions	$S_{prec} = \frac{NbRespectedPrec}{NbPrec}$ (2)	*NbRespectedPrec*: respected preconditions, *NbPrec*: mission preconditions
Action order	number of action done in order	$S_{order} = \frac{NbOrderedAction}{NbAction}$ (3)	*NbOrderedAction*: number of action done in order, *NbAction*: number of mission actions
Action count	number of action done (attempt or additional)	$S_{order} = \frac{NbOrderedAction}{NbAction}$ (4)	*NbActionPerformed*: number of action done (attempt or additional) *NbAction*: number of mission actions
Duration time	speed to perform the mission	$S_{Time} = \frac{1}{1+\exp^{-(\frac{plannedTime}{takenTime})}}$ (5)	*plannedTime*: time estimated by the expert, *takenTime*: time taken by the learner to achieve the mission.
Idle time	number of mission actions	$S_{idleTime} = 2 * (1 - \frac{1}{1+\exp^{-idleTime}})$ (6)	*idleTime*: idle time of the learner.
Actions efficiency	efficiency of action performed	$S_{Efficiency} = \frac{\sum_{i=0}^{nbAction} S_{action_i}}{NbAction}$ (7)	S_{action_i}: efficiency of action i, *NbAction*: number of mission actions

knowledge to be assessed or new information source, will not disrupt the whole architecture. This section details the assessment process and its functioning. For each agent we describe the knowledge used and its behaviours.

A. Data Source Agent

For each "information" provider component (a Database Management System for example), a *Data Source agent* is associated, whose mission is to collect the information needed for the monitoring process. This information may be related to serious gaming, such as data simulation like time or disaster evolution. This information may also be related to the geographic database such as the location of fire hydrants, the number of inhabitants in a building ... This information may also be related to agents behaviour (MAS responsible of simulation of non played actor), eg the state of an agent (moving, communication, ...) or its current action. The data source agent must retrieve data from the different *Evaluation Space*(figure 4).

1) Behaviour: The Data source agent (DSAg) receives continuously data from its linked component (Simfor, GIS database, Game Agent MAS). According to the data received, it is updated if it already exists, such as the position of an actor, or added as a new entry (eg an action performed by an actor). The DSAg has the following behaviours:

- Data dissemination: A agent or service can subscribe to the DSAg to follow a certain data. When the data in question is updated, the DSAg broadcasts the data to the concerned subscribers.
- Respond to a data query: The DSAg can receive a query for specific data, if DSAg has the value of this data, then DSAg sends it to the applicant.

2) Knowledge: The DSAg handles the following knowledge:

- Data list: Data is represented by a data structure depending on the component that generates the data. We defined three kind of data: data related to actors, data related to actions and knowledge, and finally the general information data (simulation data).
- List of subscribers: This list contains agents id (or service) that subscribe to DSAg for following some data.

B. Indicator Agent

The Indicator Agent (IAg) has to compute or to select the appropriate information for learner evaluation.

1) Behaviour: The IAg has the following behaviour:

- Compute indicator: according to the type of assessment (depending on the request of the evaluation agent) and the available indicators, the indicator agent selects the appropriate indicator, and sent it to the evaluation agent.
- Data Request: depending on the selected indicator is the indicator agent asked the agent data source for the data required for the evaluation.

2) Knowledge: The indicator agent handle the following knowledge:

- Indicators: an indicator is used to compute an assessment. This assessment may relate an action, a mission or a

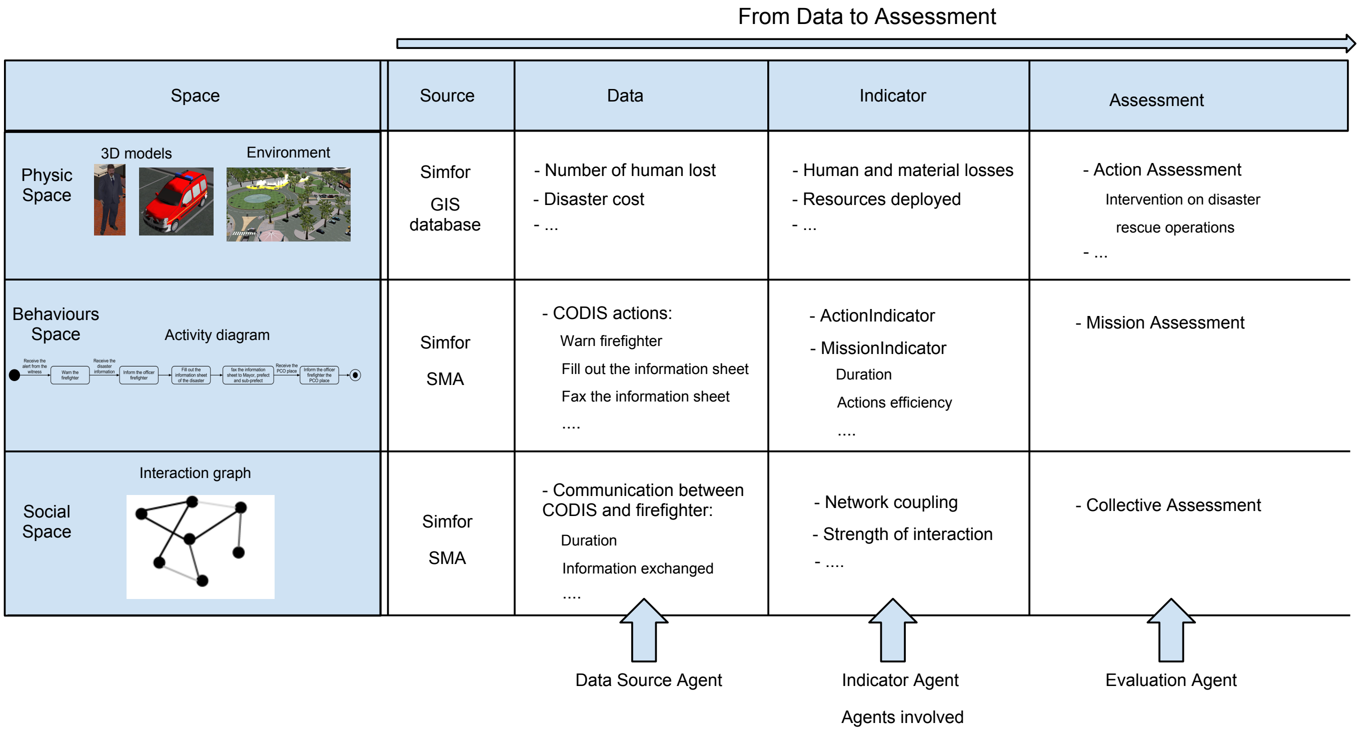

Figure 4. From data to assessment

collective assessment. Indicators and their relevance are defined by the domain expert.
Indicators are extended with specific attributes to describe specialised actions or specific missions.

- Data list: this list of data (the same structure used by data source agent) is used by indicator agent to generate indicators.

C. *Evaluation Agent*

The evaluation agent's mission is to provide a real-time individual assessment and a final evaluation at the end of the game. The collective assessment is carried out at the end of the game through negotiation between all evaluation agents (The assessment is calculated by a supervisor agent that acts as an arbitrator using the shared global knowledge of all evaluation agents). Figure 4 shows the interaction between the different agents as well as the different spaces (social space, physique space, behaviour space), this figure describes the process of a real time assessment for a specific learner.

1) Behaviour: The evaluation agent has the following behaviours:

- Real time learner assessment: the agent maintain up to date learners' evaluation along the game scenario enactment (indicators change of value will be declared by the Indicator agent thanks to its subscription list).
- Final assessment: the final evaluation of the learners behaviour during the risk management scenario will synthesise the final value of the "real time" assessments combined with information about the final campaign result (extent of destroyed are for ex.).
- Information Request for indicator: depending of the type of assessment (action, mission, real-time evaluation, final ...), the evaluator agent sends a request to the dedicated Indicator agent to get the necessary indicator value.
- Respond to an assessment request: when the evaluation agent receive a request of assessment, the evaluation agent compute a real time assessment and sends the result to the applicant.

2) Knowledge: The evaluation agent handles the following knowledge:

- Domain Model: the domain model represents the different concepts of the domain studied (see 3). The latter is segmented into several parts where each part represents a role or skill to learn. The domain model is encoded in the form of an ontology.
- Learner Model: for each learner or game agent, a learning model is associated. The learner model represents the mental state of an actor at time t. The learner model is encoded as domain model and it is encoded in the form of an ontology.
- Indicators: the evaluation agent use indicators (the same structure used by indicator agent) to compute an assessment.
- Evaluations: the evaluation at time t is represented by the global score well as well the six other scores used to compute the global score. This multicriteria approach

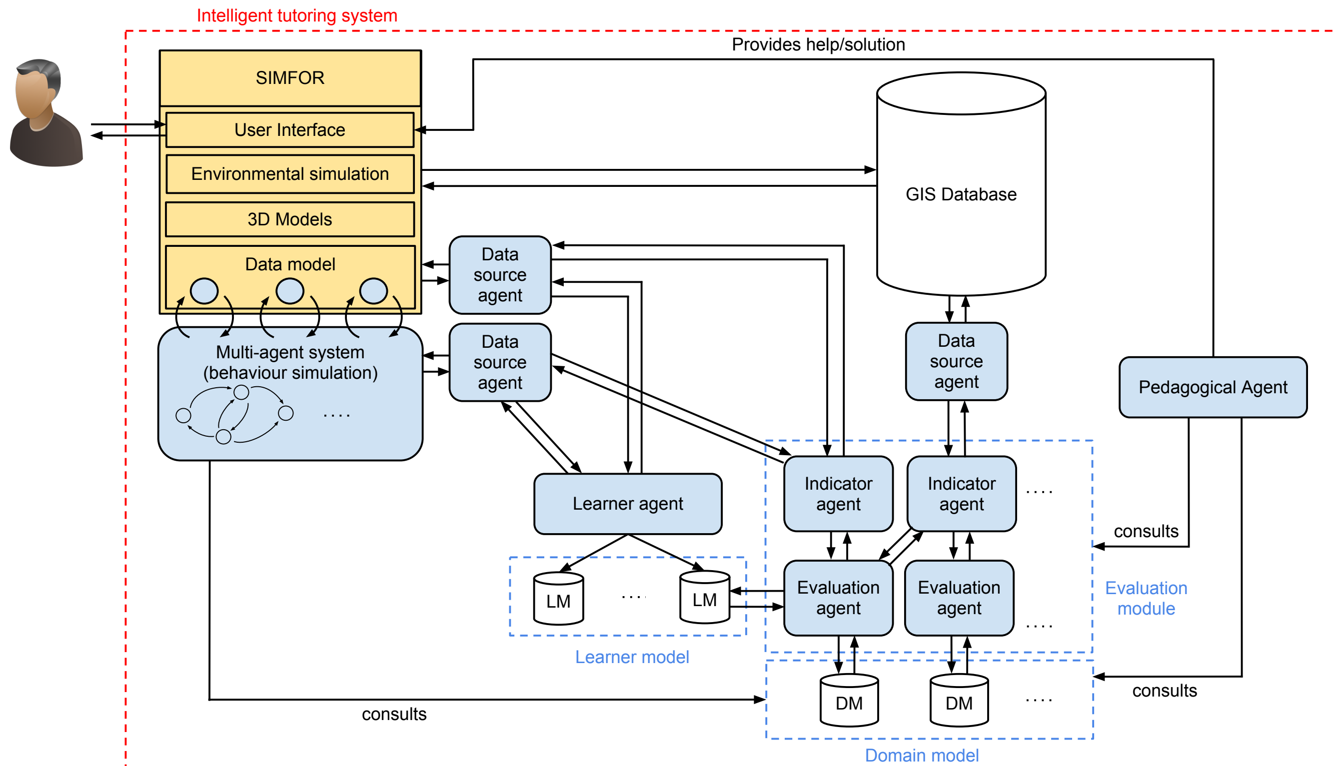

Figure 5. General architecture of the system.

allows embracing the various facets of skill evaluation.

D. Learner Agent

For each learner or agent, a learner model is associated. This model represents the mental state of actors at a time t. The Learner Agent (LAg) collects learner actions and knowledge, and stores them in the learner model. The learner model and the LAg represent the learner module of an ITS.

1) Knowledge:

- Learner model: the data structure of learner model is the same structure of the domain model, and it is an ontology, in order to use the knowledge overlap by the evaluation agent.
- Data: learner agent receive learner knowledge and action from data source agent.

2) Behaviour: The learner agent have the following behaviours:

- subscribe to data source agent: LAg subscribe to DSAg to get learner's update (knowledge and action).
- Update learner model: when LAg receive data from DSAg, the LAg update the corresponding learner model.

E. Pedagogical Agent

The pedagogical agent (PAg) plays the role of a virtual tutor that monitors learners in their training. The PAg provides support and assistance to learners to optimize learning in the virtual environment. The PAg analyses the situation (assessment results, domain model, learner model) and selects the appropriate strategy (proposing an action to perform, display help, correct, ...).

1) Behaviour: The PAg has the following behaviours:

- Assessment request: on a new event (action or environment change), the pedagogical agent request for a real time assessment.
- Select a support strategy: once the assessment results received, the PAg must select a strategy. This strategy may vary depending on the outcome of the assessment and the level of the learner and the difficulty level of the game. We define four strategies:
  - *Let the learner performs the action*: the study conducted by [20] have shown that learners who received delayed feedback have better retention of skills over time. If the student is experimented, the pedagogical agent let the learner find solution by himself.
  - *Give a clue*: if the learner is a novice, the pedagogical agent begins by giving clues about the procedure to follow.
  - *Propose action*: if the learner has difficulties to perform the procedure (time attributed to the action exceeded), the pedagogical agent propose action to realize (in the case of CODIS, call firefighter).
  - *Do the action in place of the learner*: if the learner does not know how to do the action, the pedagogical agent performs the action in his place while explain-

ing how to do it.

2) Knowledge: The PAg handle the following knowledge:

- Learner model: to provide a support, the PAg need to access to learner model. The learner model provides the history of learner actions and learner knowledge.
- Domain model: the domain model provides to the PAg the training step of the learner, this allows the PAg to determines the next action to be performed by the learner.
- Evaluation: the result of the assessment allows to identify the gaps of learners through the six different scores.

VI. General working of the system: A scenario example

To illustrate learner assessment, we present an example of scenario defined with the help of a domain expert that describes the interaction between the different agents. We present a simplified example of the missions to be performed by the actor playing the role of CODIS (Departmental Operational Fire and Rescue Services Centre) for a TDM scenario (Transportation of Dangerous Material). The scenario begins with a TDM truck which has spilled due to a traffic accident. The tank is damaged and the fuel is spreading over the road. A witness to the accident gave the alarm by calling the CODIS.

Figure 6 describe agents interaction with UML sequence diagram during the assessment process. The *Learner agent* has for mission to update learner model (a *Learner agent* is associated to each actor), for that the learner agent subscribes to *Data Source agent*(DSAg) (arrow 0, figure 6) to get learner's actions and his knowledge.

When the learner performs action (in this example the CODIS actor calls a Firefighter actor to warn him of the accident), the DSAg sends action data to the learner agent (1) (id actor, target, ...). Once the data is received, the *Learner agent* updates the learner model and informs the *Pedagogical agent* (PAg) (2) of a new event (new action performed). Once the event is received, the *PAg* begins a learner support process. First, the PAg request for learner assessment (3). The *Evaluation agent* (EAg) retrieves the relevant informations from the learner model (4) (level, role, pending procedure, ...), then requests the Indicator Agent (IAg) to get the indicators' value pertaining the evaluation (5). In the example, the learner performs an phone action, therefore the EAg performs an action's assessment. To compute action's efficiency, the EAg receive actionPhoneIndicator with the necessary information (6). This indicator is specific to phone action and has for parameters: action's execution time and the information exchange during the call. Once the assessment computed, the EAg sends the result to PAg (7). The PAg analyses the situation (learner assessment, domain model and learner model) and selects a strategy for learner support.

When EAg detect an mission complete, the EAg proceeds to a mission assessment. For that, the EAg request IAg to get adequate indicators. Depending of mission, the IAg request DSAg to get more data to complete the indicators, for example: to assess a firefigther for mission *"intervention on disaster"*, the IAg needs *material and human loss* and *time to master the fire*. As mentioned in V-C, the EAg compute an assessment score combining different scores using data form the three different space (physic space, social space and behaviour space).

VII. Conclusion

With the growing interest in SG for training purpose, the evaluation of players (learners) is increasingly relevant. In this paper we present different characteristics and problems of learners assessment. Then we propose to add players' assessment capabilities to SG using on the *"evaluation space"* concept. Then we propose a MAS architecture to support learner assessment based on this *"Evaluation Space"* concept and integrated to the SIMFOR project, a serious game for training non-professional to risk management. This integration to SIMFOR permits to enhance its pedagogical capabilities by proposing a heterogeneous assessment system (assessing different trades and skills: fireman, policeman, prefect ...) and distributed (three different evaluation space, physical, social and behavioural), based on an Intelligent Tutoring System (ITS). The ITS allows individual and collective assessment in real time, and an assessment at the end of the game that allows an individual and collaborative tasks diagnostic.

Our future work in the SIMFOR project is to focus on the collaborative aspects in the field of crisis management, based on an analysis of the interaction graph permitting real-time interpretation for better pedagogical support.

References

[1] G. Durand and C. Martel, "To scenarize the assessment of an educational activity," in *World Conference on Educational Multimedia, Hypermedia and Telecommunications*, vol. 2006, no. 1, 2006, pp. 845–852.

[2] W. Johnson and S. Wu, "Assessing aptitude for learning with a serious game for foreign language and culture," in *Intelligent tutoring systems*. Springer, 2008, pp. 520–529.

[3] A. Vidani and L. Chittaro, "Using a task modeling formalism in the design of serious games for emergency medical procedures," in *Games and Virtual Worlds for Serious Applications, 2009. VS-GAMES'09. Conference in*. IEEE, 2009, pp. 95–102.

[4] D. R. Michael and S. L. Chen, *Serious Games: Games That Educate, Train, and Inform*. Muska & Lipman/Premier-Trade, 2005.

[5] M. Zyda, A. Mayberry, C. Wardynski, R. Shilling, and M. Davis, "The moves institute's america's army operations game," in *Proceedings of the 2003 symposium on Interactive 3D graphics*, ser. I3D '03. New York, NY, USA: ACM, 2003, pp. 219–220.

[6] A. Tricot and R. Alain, "Modalités et scénarios d'interaction dans des hypermédias d'apprentissage," *Revue des sciences de l'éducation*, vol. 25, no. 1, pp. 105–129, 1999.

[7] J. Alvarez and D. Djaouti, "An introduction to serious game - definitions and concepts," in *Proceedings of Serious Games & Simulation for Risks Management Workshop*. LARSEN Science, December 2011.

[8] R. Stone, "Serious games: virtual realitys second coming?" *Virtual reality*, vol. 13, no. 1, pp. 1–2, 2009.

[9] J. Anderson, "The expert module," *Foundations of intelligent tutoring systems*, pp. 21–53, 1988.

[10] K. VanLehn, "Student modeling," *Foundations of intelligent tutoring systems*, pp. 55–78, 1988.

[11] J. Dintilhac and R. Ignace, "Evaluation de la technologie en collége, académie de montpellier," ACADEMIE DE MONTPELLIER, Tech. Rep., 2005.

[12] D. Nieborg, "America's army: More than a game," in *In the Proceedings of 35th Annual Conference of the International Simulation And Gaming Association (ISAGA) and Conjoint Conference of SAGSAGA*, 2004.

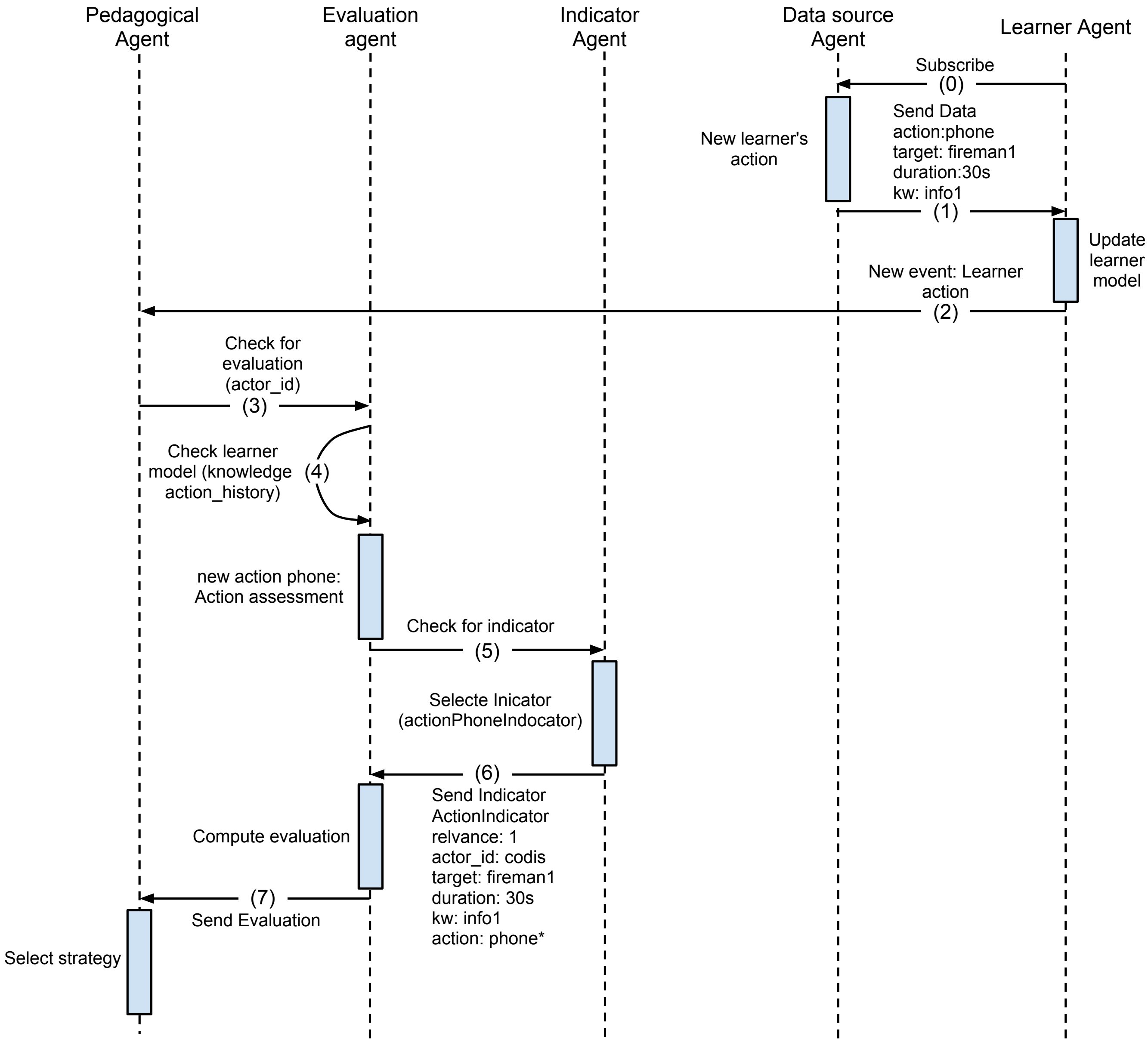

Figure 6. UML sequence diagram describing agents interaction.

[13] H. Burns and C. Capps, "Foundations of intelligent tutoring systems: An introduction," *Foundations of intelligent tutoring systems*, pp. 1–18, 1989.

[14] T. Murray, "Authoring intelligent tutoring systems: An analysis of the state of the art," *International Journal of Artificial Intelligence in Education (IJAIED)*, vol. 10, pp. 98–129, 1999.

[15] M. Ma, A. Oikonomou, and L. Jain, *Serious games and edutainment applications*. Springer, 2011.

[16] J. Tchétagni and R. Nkambou, "Hierarchical representation and evaluation of the student in an intelligent tutoring system," in *Intelligent Tutoring Systems*. Springer, 2002, pp. 708–717.

[17] D. Provitolo, J. Müller, and E. Dubos-Paillard, "Validation of an ontology of risk and disaster through a case study of the 1923 great kanto earthquake," *ICCSA Proceedings Special Sessions*, pp. 140–150, 2009.

[18] C. Buche, C. Bossard, R. Querrec, and P. Chevaillier, "Pegase: A generic and adaptable intelligent system for virtual reality learning environments," *International Journal of Virtual Reality*, vol. 9, no. 2, p. 73, 2010.

[19] A. Astin *et al.*, *Assessment for excellence: The philosophy and practice of assessment and evaluation in higher education*. Rowman & Littlefield Publishers, 2012.

[20] R. Schmidt and R. Bjork, "New conceptualizations of practice: Common principles in three paradigms suggest new concepts for training," *Psychological science*, vol. 3, no. 4, pp. 207–217, 1992.

[21] W. Johnson, J. Rickel, R. Stiles, and A. Munro, "Integrating pedagogical agents into virtual environments," *Presence*, vol. 7, no. 6, pp. 523–546, 1998.

[22] D. Lourdeaux, J. Burkhardt, F. Bernard, and P. Fuchs, "Relevance of an intelligent agent for virtual reality training," *International Journal of Continuous Engineering and Life-long Learning*, vol. 12, no. 1/2/3/4, pp. 131–143, 2002.

[23] T. Stratulat, J. Ferber, and J. Tranier, "Masq: towards an integral approach to interaction," in *Proceedings of The 8th International Conference on Autonomous Agents and Multiagent Systems - Volume*

2, ser. AAMAS '09. Richland, SC: International Foundation for Autonomous Agents and Multiagent Systems, 2009, pp. 813–820. [Online]. Available: http://dl.acm.org/citation.cfm?id=1558109.1558124

[24] K. Mustapha, E. Tranvouez, B. Espinasse, and A. Ferrarini, "An organization-oriented methodological framework for agent-based supply chain simulation," in *Research Challenges in Information Science (RCIS), 2010 Fourth International Conference on*. IEEE, 2010, pp. 353–364.

[25] A. Chadli, E. Tranvouez, and F. Bendella, "Combining agent-based participatory simulation and technology enhanced learning for rodent control," in *Proceedings of the Summer Computer Simulation Conference*. Society for Modeling & Simulation International, 2009, pp. 377–382.

[26] M. Wooldridge, N. Jennings *et al.*, "Intelligent agents: Theory and practice," *Knowledge engineering review*, vol. 10, no. 2, pp. 115–152, 1995.

[27] A. Rao, M. Georgeff *et al.*, "Bdi agents, from theory to practice," in *Proceedings of the first international conference on multi-agent systems (ICMAS-95)*. San Francisco, 1995, pp. 312–319.