An ontology for automated scenario-based training

Marieke M.M. Peeters*

Department of Intelligent Systems,
Delft University of Technology,
P.O. Box 5031, 2600 GA Delft, the Netherlands
Email: M.M.M.Peeters@tudelft.nl
*Corresponding author

Karel van den Bosch and Mark A. Neerincx

TNO (Dutch Organization for Applied Sciences),
Training & Performance Innovations/Perceptual & Cognitive Systems,
P.O. Box 23, 3769 ZG Soesterberg, the Netherlands
Email: karel.vandenbosch@tno.nl
Email: mark.neerincx@tno.nl

John-Jules Ch. Meyer

Department of Information and Computing Sciences,
Utrecht University,
P.O. Box 80.089, 3508 TB Utrecht, the Netherlands
Email: J.J.C.Meyer@uu.nl

Abstract: An intelligent system for automated scenario-based training (SBT) needs knowledge about the training domain, events taking place in the simulated environment, the behaviour of the participating characters, and teaching strategies for effective learning. This knowledge base should be theoretically sound and should represent the information in a generic, consistent, and unambiguous manner. Currently, there is no such knowledge base. This paper investigates the declarative knowledge needed for a system to reason about training and to make intelligent teaching decisions. A frame-based approach was used to model the identified knowledge in an ontology. The ontology specifies the core concepts of SBT and their relationships, and is applicable across training domains and applications. The ontology was used to develop a critical component of SBT: the scenario generator. It was found that the ontology enabled the scenario generator to develop scenarios that fitted the learning needs and skill level of the trainee. The presented work is an important step towards automated scenario-based training systems.

Keywords: ontology; scenario-based training; cognitive engineering; reusability; educational games; personalisation; knowledge representation.

Reference to this paper should be made as follows: Peeters, M.M.M., van den Bosch, K., Neerincx, M.A. and Meyer, J.J.Ch. (2014) 'An ontology for automated scenario-based training', *Int. J. Technology Enhanced Learning*, Vol. 6, No. 3, pp.195–211.

Biographical notes: Marieke M.M. Peeters (PhD) is a postdoc at Delft University of Technology, where she conducts research on cognitive systems and man-machine collaboration. After working on personalised educational games and technology-enhanced learning for four years, she is now extending her scope to the domain of autonomous technology in the military and healthcare domain.

As a senior researcher at TNO, Karel Van Den Bosch is involved in the development and deployment of intelligent software agents in simulation- or game-based training. The goal is to increase effectiveness of training by consistently eliciting the intended behaviour of the learner, and efficiency of training, e.g. team members no longer need to be present.

Mark A. Neerincx is a Professor in the Interactive Intelligence group at Delft University of Technology, and senior researcher at TNO Perceptual and Cognitive Systems. He has extensive experience in fundamental and applied research on human-computer interaction, among other things in the domains of health, security, defence, and space.

John-Jules Ch. Meyer is a full Professor of Computer Science at Utrecht University where he heads the Intelligent Systems Group. His current research interests include artificial intelligence, and intelligent agents in particular, from theory to applications. He is a Fellow of the European Coordinating Committee for Artificial Intelligence.

This paper is a revised and expanded version of a paper entitled 'An ontology for integrating didactics into a serious training game' presented at the '1st International Workshop on Pedagogically-driven Serious Games (PDSG 2012), during the European Conference for Technology Enhanced Learning (ECTEL)', Saarbrucken, 18–21 September 2012.

1 Introduction

To introduce the problem addressed in this paper, we start by presenting the reader with a case that will be used throughout the paper as an exemplary context.

Henry opens his laptop to participate in a training session of his First Aid Course. He is welcomed by his virtual instructor: "You did well last time, let's practice what you've learned about treating burns." Henry clicks 'start'. A game world opens up, showing a garage with virtual characters working on cars. Henry uses his mouse and keyboard to control his character in the game. The other characters are intelligent agents that act autonomously. Suddenly a hot soldiering iron falls on the arm of his virtual colleague, Marge. Henry hurries forward and inspects Marge's arm. The game zooms in on the scene, and Henry sees blisters developing. Marge is in pain and very upset. Henry instructs her to go to the water tap and cool the wound. He guides her and provides reassurance as they walk to the restrooms. While cooling the wound, they see smoke coming from the garage. The hot soldiering iron has ignited a fire. Henry leads Marge out of the building and calls the fire department. Then the Virtual Instructor appears: the scenario has ended. "How did it go, Henry? Can you tell me what went well and/or what you would do different were you to do it again? Please explain your answers." Henry enters his reflection into the system. Marge explains how she experienced the events and how her wound recovered after Henry's treatment. Virtual Instructor concludes: "Not bad Henry, but remember to pay attention to the safety of yourselves and your environment."

To produce the behaviour described above, this automated training system uses knowledge about: the domain and task of First Aid (e.g. how to treat burns); events taking place in the virtual environment; the behaviour of the participating characters; and teaching strategies (e.g. when and how to give feedback). This knowledge must be made available to the system in a correct, consistent, and unambiguous manner.

Automated training systems gradually become more complex. As a result, they rely on ever larger knowledge bases. Consequently, a large part of the development time is spent on constructing the knowledge base. To reduce the development time, the knowledge base must be designed such that it is reusable across training domains and system designs. Therefore, the knowledge must be represented *explicitly* using a (preferably) well-known format.

1.1 Problem statement

In scenario-based training (SBT), learners engage in interactive storylines. In order for a system to support – theoretically and empirically founded – automated SBT, we need an unambiguous – i.e. correct, consistent, and transparent – knowledge representation for two reasons: (1) to integrate didactic principles into an automated training system; and (2) to explicitly specify which scenario variables and performance variables are used when deciding upon corresponding scenario adaptations.

To construct such a knowledge representation, theories and models of learning and instruction need to be formalised. Even though the training domain may differ, the concept of SBT and its didactic principles are generic. A knowledge representation for SBT should therefore be largely reusable across training domains, systems, and applications.

This paper uses the format of frame-based ontology to develop a knowledge representation for SBT (Minsky, 1975). Frame-based ontology is derived from semantic networks, and is based on psychological theories of human cognition. This approach is considered to be especially appropriate for representing knowledge involved in training, because a frame: (1) closely resembles the internal representations believed to be maintained by human experts; and (2) is used to represent declarative knowledge (in contrast to, for instance, rule-based systems and heuristics, which are more suitable for representing procedural knowledge).

1.2 Our contribution

A reusable (declarative) knowledge representation for automated SBT is currently lacking. This paper presents the construction and verification of such a knowledge representation using a well-known format: a frame-based ontology. The following research question is investigated:

What declarative knowledge is needed to support (automated) SBT and how can this best be represented in an ontology?

2 The knowledge to be modelled: about scenario-based training

Before knowledge can be represented, we need to know exactly *what* knowledge is needed to enable automated SBT and a given training domain. In order to acquire this knowledge, we conducted an investigation into the concept of SBT.

2.1 Obtaining the knowledge relevant to SBT

SBT relies on various reasoning processes, such as: assessing and interpreting a learner's task performance, deducing a learner's competencies from a given task performance, predicting what learning goal is suitable to challenge and motivate a learner to further develop his/her competencies, etc. To investigate what knowledge is relevant to support such reasoning, the literature about SBT, learning, and instruction was consulted. In addition, because the concept of SBT may differ from the practice of SBT, a series of interviews was conducted with domain experts, i.e. instructors who regularly train people by means of storylines within simulated environments.

The interviews used in this investigation consisted of: (1) a joint interview with four instructors from the *Helicopter Directive Operator* training programme of the Netherlands Royal Navy; (2) a joint interview with five *First Aid* instructors; (3) a joint interview with six instructors who provided training days for *In-company Emergency Management* teams; and (4) six individual interviews with *First Aid* instructors.

In addition, three SBT sessions were attended to experience the practice of SBT first-hand, i.e. (1) a training session as part of the *Command Center Officer* training programme of the Netherlands Royal Navy; (2) a training session as part of the *Platoon Officer* training programme of the Netherlands State Police; and (3) a training session as part of the *Emergency Care* training programme of the Utrecht University's Medical Centre.

The investigation covered a wide range of training domains. Therefore, the principles underlying SBT presented in this paper are applicable across training domains. However, for reasons of consistency, examples presented throughout this paper involve First Aid.

2.2 Results of the investigation into SBT

The results of the investigation are reported below. The report describes how SBT is employed, and how SBT supports the development of competencies relevant in professions characterised by complex tasks and decision making, such as police men, fire fighters, military, and emergency healthcare professionals.

2.2.1 The need for practical experience

Becoming proficient in a domain of expertise, e.g. 'First Aid', requires a learner to gain experience in solving problems that are typical and critical for the work situation, e.g. 'treat a burn'. Practical experience is considered to be especially important, because there is evidence that experts use previous experiences to recognise similarities with the task situation at hand (Klein, 2008). Based on recognised similarities, experts adopt a solution plan, e.g. 'cool the burn with running water'.

Situational experiences are stored in long-term memory in the form of patterns, sometimes referred to as *schemata* (Tannen, 1993). Schemata contain relevant cues, expectancies, goals, and responses related to a particular situation. For example, a schema of a burn injury may contain information about the type and size of the blisters, and the (either positive or negative) effects of cooling a burn with water, treating it with ointment, applying various types of bandages, etc.

Situation awareness is a person's awareness of the current situation in a dynamic environment (Endsley, 1995), such as being aware of the size and type of the blisters inflicted by a burn or being aware that a soldiering iron is still turned on. Provided that one knows what to look for, situation awareness allows for quick recognition of the current situation's distinguishing features for comparison with the schemata stored in long-term memory.

2.2.2 Scenario-based training to facilitate experiential learning

On-the-job learning is a way to gain the experience needed to become an expert professional. However, especially in high-risk professions, learning on-the-job has dangerous and often life-threatening consequences. Therefore, alternative forms of experiential learning and training are needed. An example is scenario-based training (SBT).

SBT facilitates competency development through experience: learners participate in *scenarios*, such as the scenario described in the introduction, i.e. re-enactments of real-life situations that are representative for a learner's future profession (Oser et al., 1999; Van den Bosch and Riemersma, 2004). The interactive nature of SBT offers learners a sense of control over the learning experience, thereby increasing motivation and performance (Cairncross and Mannion, 2001; Mayes and Fowler, 1999; Corbalan Perez et al., 2009).

Scenarios are typically staged in a simulated environment. To ensure competency development during training, the simulated environment must allow the learner to learn the relevant cues, expectancies, goals, and responses, so that these can be stored in schemata (Baldwin and Ford, 1988; Young, 1993; Yamnill and McLean, 2001; Houtkamp, 2012). To support learners in developing generally applicable schemata, it is important to offer them a large variety of situations (Atkinson et al., 2000; Baldwin and Ford, 1988; Yamnill and McLean, 2001).

SBT is more than just a simulated environment: behind the scenes, the instructor and role players (i.e. staff members) reason about the didactic value of the current training situation. Based on their assessments, they decide whether or not adjustments in the scenario are needed to ensure a situation that allows the learner to accomplish the learning goals.

2.2.3 Automating scenario-based training

SBT is a successful training form. However, training opportunities in SBT are often relatively scarce due to elaborate preparations. One way to increase the available training opportunities is by (partially) automating SBT with the use of modern technology. Therefore, researchers have been studying the use of virtual environments and the automation of the role players in the exercises (Cannon-Bowers et al., 1998; Oser et al., 1999). Yet the availability of configurable virtual environments and artificial NPCs does not by itself guarantee good training.

To automatically support SBT, there are various didactic reasoning processes (i.e. *procedural knowledge*) that need to be automated to support effective instruction. For example, the system should be able to select an appropriate learning task (or scenario) given a learner's prior knowledge structures (Salden et al., 2006; Kalyuga and Sweller, 2005). Other examples are: reducing the amount of unnecessary cognitive load on the part of the learner (Mayer, 2008); providing learners gradually decreasing support (i.e. scaffolds) (Sweller et al., 2011); and asking the learner to reflect on decisions, actions, and the results thereof (Chi et al., 2001; Hattie and Timperley, 2007). To support these reasoning processes, the system needs an internal knowledge representation (about SBT, the learner, and the training domain) (Hoekstra, 2009).

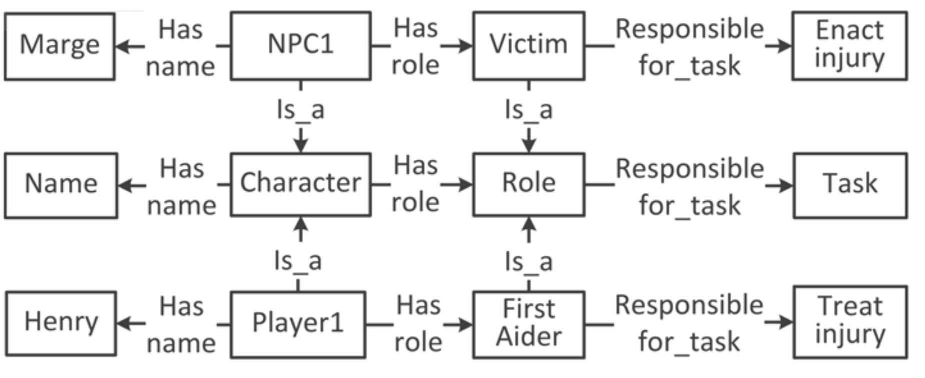
3 Representing knowledge

This paper proposes the use of a frame-based ontology to represent the declarative knowledge required for automated SBT in a given training domain.

3.1 Frame-based ontology

A frame-based ontology provides an explicit, structured, and semantically rich representation of *declarative* knowledge (Hoekstra, 2009); procedural rules and heuristics are not part of an ontology. A frame-based ontology consists of concepts and instances of those concepts. For example, the concept *Character* has the NPC *Marge* as its instance (see Figure 1).

Figure 1 An example of a frame



Frames represent *stereotyped situations* in the form of a group of interrelated concepts with a fixed structure (Minsky, 1975). Frames can in fact be regarded as schemata: they represent *structures of expectation* constructed from previous experience (Tannen, 1993). Because of this direct mapping, a frame-based ontology is considered appropriate to represent knowledge for the application of our research.

The frame in Figure 1 describes the concept ‘Character’, which has two attributes (slots): a name and a role. In turn, the concept ‘Role’ also has a slot: being responsible for performing a particular task. These concepts form the ‘top’ level of the frame; the top level represents knowledge that is considered to be true for any situation modelled according to this stereotype. Slots represent features of a concept that are generally applicable for the stereotypical situation.

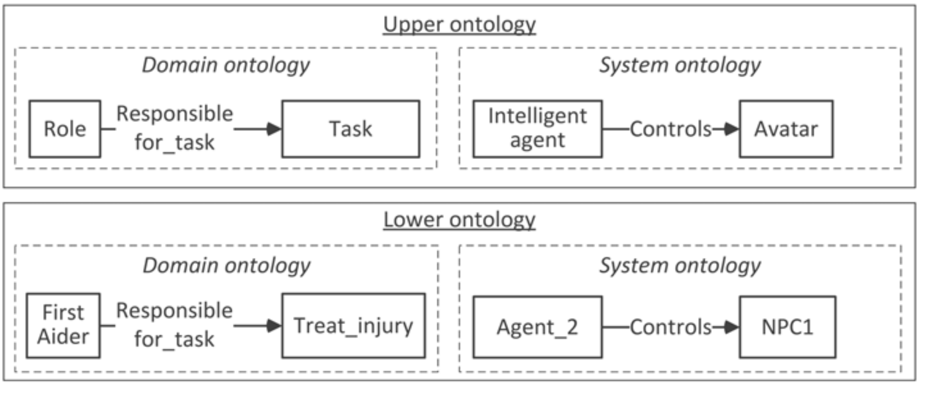
Instances of a general stereotype have been assigned concrete values. In Figure 1, there are two instances of the ‘Character’ concept: NPC1 and Player1. When creating an instance of a concept, the slots of the concept are inherited and must be specified for that particular instance. For example, the concept ‘character’ has the attribute that it has a name. However, the *actual name* of a given character is not yet specified, because if it was specified at the concept level, then all instances would have the same name. The slots of an instance must be specified, e.g. when adding an instance of a character to the ontology, the name of that character must be specified. NPC1 is named ‘Marge’.

It is possible to specify *default values*: overridable slot values that are automatically assumed for any given instance belonging to the concept, unless otherwise specified. Default values facilitate reasoning with incomplete information. For example, the default role of a player may be the First Aider role.

3.2 Various types of ontology

The ontology of automated SBT comprises a combined ontology. Within the ontology for automated SBT we distinguish four ontology parts based on two features: (1) level, i.e. upper vs. lower, and (2) type, i.e. domain vs. system (also see Figure 2).

Figure 2 Various types of ontology



3.2.1 Upper vs. lower ontology

An upper ontology describes the top level concepts: the stereotypical information needed to reason about general, non-specific, situations. The structure of the upper level ontology is fixed and considered to be applicable across situations (e.g. training domains) without the need for alterations. For our purposes, the upper level ontology provides generic knowledge about SBT, task performance, and competency development. Figure 2 provides an example: the upper domain ontology states that a role is responsible for a task. The descriptions of role and task are regarded to be applicable, hence reusable, across training domains.

A lower ontology describes specific situations (e.g. training domains) that adhere to the upper level structure (e.g. a generic description of training domains in general). In Figure 2, the concepts and relations contained in the upper ontology are applied to the training domain of First Aid: a First Aider is responsible for treating an injury.

Distinguishing between an upper and lower ontology allows for reuse across training domains. The system's reasoning rules should be developed such that they use the upper ontology structure to reason about training and task performance *in general*. When the ontology is applied to a particular training domain, the reasoning rules use the upper level structure to request the lower level, situation-specific, information where necessary. This allows for reusability of the reasoning rules as well as the upper level ontology across training domains. Furthermore, it reduces development time by requiring only for a specification of the knowledge that is typical for the new training domain. This paper presents only the contents of the upper ontology in detail. The upper ontology describes the concepts of SBT in general, i.e. independent of the training domain.

3.2.2 Domain vs. system ontology

A domain ontology contains the declarative knowledge relevant to the domain in which the system will operate, regardless of that system's design. In the example (Figure 2), the domain ontology describes the relation between, e.g., role and task (upper), First Aider and treat injury (lower), thereby describing a part of the domain of SBT and First Aid.

A system ontology contains concepts relevant to the system's design. In the example (Figure 2), the system ontology describes the relation between, e.g., intelligent agent and avatar (upper), agent2 and NPC1 (lower). These concepts and instances describe the system's design and/or a given state of the system.

Separating the domain knowledge from the system knowledge is especially useful in knowledge-based systems, because in such systems the body of knowledge concerning the domain should be made reusable across various system designs. *This paper presents only the domain ontology, describing SBT independent of the system's design.*

4 Construction of the ontology

The ontology was constructed as outlined below.

4.1 Ontology construction procedure

The ontology for SBT was developed using the procedure proposed by Noy and McGuinness (2001):

1 Determine domain and scope of ontology: Automated SBT (for First Aid)

2 Consider reuse of existing ontology: The 'task analysis'-ontology by Van Welie et al. (1998) (see below).

3 Enumerate concepts in ontology: Concepts relevant to SBT

4 Define classes and hierarchy: Results described in Section 2.2 were used.

5 Define properties of classes, called slots: Results described in Section 2.2 were used.

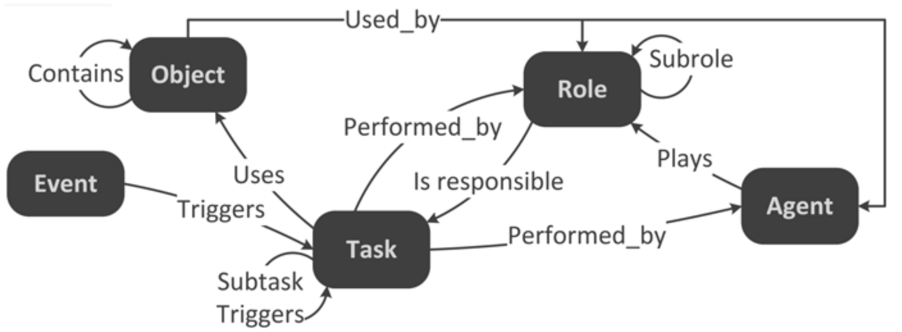
6 Define constraints of slots and create Instances: Results described in Section 2.2 (including results typical for First Aid) were used.

Constructing an ontology is a dynamic process: each step results in new knowledge about the concepts, attributes, relations, and their definitions. It is often necessary to move back and forth between these steps to check the consistency of new additions or alterations with the rest of the ontology.

4.2 Reused ontology and extension

When developing the ontology for SBT, the 'task analysis'-ontology by Van Welie et al. (1998) (see Figure 3) was reused.

Figure 3 An ontology for task analysis by Van Welie et al. (1998)



The ontology by Van Welie et al. (1998) models the domain of 'Task Analysis' (Annett, 2004; Clark and Estes, 1996) and can be used to describe relations between, e.g. tasks, situations, events, procedures, and tools within a given task domain. In SBT, learners aim to develop the competencies required to perform a particular *task*. Therefore, the ontology by Van Welie et al. (1998) provided an appropriate start for constructing an ontology for SBT. However, task analysis is a different application than SBT. Therefore, the ontology of Van Welie et al. (1998) needed to be extended. For example:

1 The 'task-analysis'-ontology can be used to describe the way a task should be performed. However, it does not allow for any deductions about a person's competency to perform the task correctly, nor how to support a learner in developing competency in performing a task. For this, the ontology needed to be extended with concepts that describe the relation between competency development in a learner and performance outcomes observed in the training environment.

2 The 'task-analysis'-ontology provides an abstract upper level ontology, e.g. tasks are performed by roles with the use of objects. Yet SBT concerns tasks staged within the training environment in the form of storylines. Tasks are structured, contextualised, and coordinated with the use of scenarios. Therefore, the knowledge representation, i.e. the ontology, for SBT must include concepts needed for reasoning about the storyline in terms of task performance, e.g. actions, object features, characters, or situational information.

5 An ontology for automated SBT

Figure 4 presents the upper domain ontology for SBT. It has been divided into three layers: (1) 'Training Environment', (2) 'Task Domain', and (3) 'Didactics'. These three layers must be coordinated and integrated to support the design and delivery of automated SBT. The three layers are discussed in more detail below.

The ontology description starts with Layer 2 concerning the Task Domain. A task requires particular competencies. This is depicted in Figure 5: the concept 'task' in the centre of the figure is connected to the big green area on the left named 'Competency' by means of an arrow that says 'requires'. There are three types of competencies: skill, attitude, and knowledge. Knowledge can be categorised into three types: (1) assumed knowledge – knowledge that the learner is assumed to possess before entering the training, (2) domain knowledge – knowledge the learner should acquire during training, and (3) situation awareness – knowledge that is contextual to the task performance and only becomes available once the exact task situation can be observed and assessed. Situation awareness is specifically important because it is the learner's awareness of information that is relevant to the way he/she should perform the task at hand; based on the learner's situation awareness, he/she is able to determine what procedure is applicable to perform.

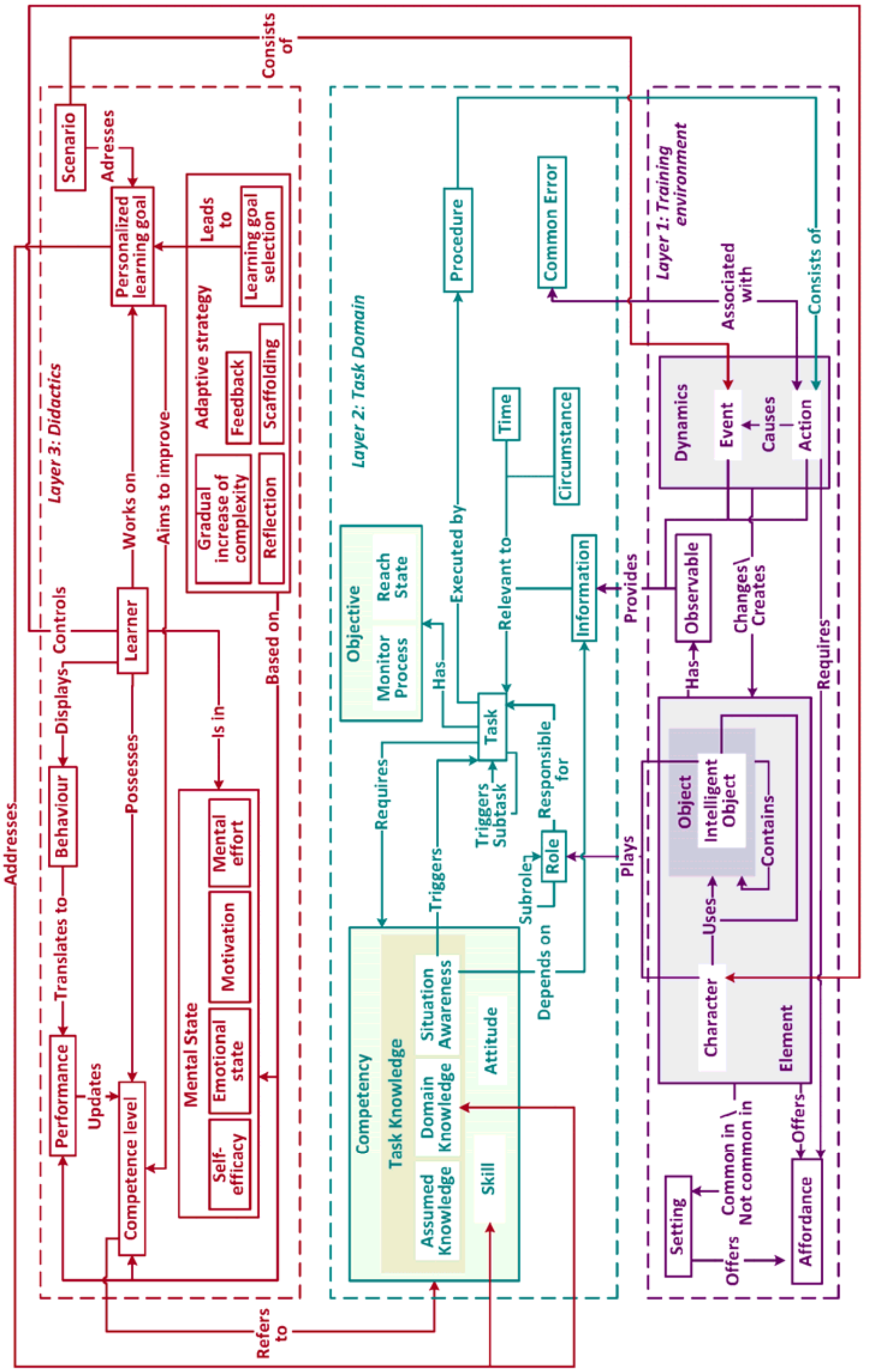
A task typically has an objective. This objective can be monitoring a process or establishing some situation. The way in which a task is to be performed depends on the circumstances under which it is performed, e.g. danger, complications, etc. In addition, time (e.g. duration, timing) is often relevant to the task performance. Furthermore, a task usually requires information that is conveyed by the world. Because the world in SBT is a simulated one, it is important that it provides the information required to enable the learner to act. Therefore, this information should be made explicit and must be actively added to the training environment. A task can contain subtasks until eventually the subtask breaks down into a procedure (i.e. an action sequence).

When performing a task, the learner may make errors. By identifying and explicitly representing errors that are known to occur, the system will be able to recognise them and, in turn, to deliver error-specific feedback. For this reason, errors are specifically specified as alternatives to the correct actions in a procedure.

Tasks are the responsibility of roles, which in turn are impersonated by characters. If a character (the learner, or an NPC) has a particular role, he/she is responsible for the tasks described in the task specification of that role.

We now move on to Layer 3 of the ontology: the didactics. This breaks down into two parts: (1) a description of the learner's experience and performance from a didactical point of view, and (2) the role of the scenario within the training program or curriculum. Let us start again with the central concept of this layer: the learner. The learner controls one of the scenario's characters and, as such, plays one of the roles. As the learner engages in task performance in the virtual world, he/she displays particular behaviour. This behaviour is interpreted in terms of task performance. The resulting performance assessment is used to update the learner's competency levels. In addition, the learner's mental state can be analysed and estimated. The ontology distinguishes four components: the learner's self-efficacy (confidence in one's own ability), motivation, emotional state, and the mental effort invested during task performance. Furthermore, the learner is working on a personalised learning goal, which aims to improve a particular competency level.

Figure 4 A graphical representation of the upper domain ontology (automated SBT)



The personalised learning goal is selected by applying an adaptive strategy called 'learning goal selection' and results in a subset of competencies that are currently suitable for the learner to develop during training. (How exactly this is determined depends on the procedural knowledge of the system.) The learning goal is addressed in the scenario offered to the learner. There are several other adaptive strategies that can be applied to increase the didactic effectiveness of the scenario: providing feedback, which may be specifically targeted at some common error. In addition, the automated system for SBT may apply scaffolds, and stimulate the learner to reflect on his/her performance. Whether and how these strategies are applied depends on heuristics and procedural rules, which are not part of the ontology.

The final layer of the ontology is Layer 1: 'Training Environment'. We start with the most central concept 'Character'. A character is an avatar in the scenario that is controlled by the learner or a non-player (in that case it is called a non-player character, or NPC). Characters in the scenario execute actions that are consistent with their role and tasks. There are also other elements available in the environment. These are called objects. A special type of objects is the intelligent object (i.e. controlled by an intelligent agent). It is an object that can exhibit intentional goal-directed behaviour, such as a dog, a robot, a vehicle, or a fire. A setting is a scenery, a static background for the scenario, such as a kitchen or a park. Elements can be common or not common within certain settings, e.g., a stove is common in a kitchen, but not in a park; a tree is common in a park, but not in a kitchen.

Elements can offer particular affordances, i.e. possibilities for a person to engage in certain interactions with the environment (Gibson, 1977). Affordances can be exploited while executing an action. An example is the following: when a character wants to break a limb (because this is part of the storyline), it needs to construct a plan to do so, such as 'fall from height'. The object 'stair' can provide the affordance 'fall from height' and enables the character to fall and break a limb. By assigning affordances to objects in the ontology, affordances specify the way in which objects can be used to perform particular actions.

The dynamics consist of actions and events. Actions are performed by characters (or intelligent objects) because they are part of a procedure to perform a task and that task is a character's responsibility because of the role it has adopted. Actions cause events, which in turn change the world by changing or creating elements. Events can also take place without being caused by a deliberate action, for instance, a tree may fall down simply from being old and rotten, but it may also come down because someone chopped it down with an axe. Once an event takes place, the world changes, leading to new tasks becoming relevant. For example, if indeed a tree comes down, it might be one's task to clear it from the road. As a result, elements and dynamics provide observables. However, these observables only lead to information and situation awareness if the learner picks them up accordingly.

As an exemplary case, the ontology pertains to the training domain of First Aid. The lower ontology contains domain-specific concepts related to First Aid, such as tasks, objects, roles, actions, and competencies. Protégé Frames 3.5 (http://protege.stanford.edu), an ontology editor for frame-based ontology, was used to implement the ontology (Friedman Noy et al., 2000). The complete ontology, containing the upper and lower ontology, is available for download at http://mariekepeeters.com/indigo. This ontology also includes the definitions for all concepts and relations in the ontology.

6 Evaluating the knowledge representation

The objective of the upper ontology described above is to provide a structure representing the knowledge that is needed to deliver automated SBT. To test the validity and usefulness of the upper ontology, it was applied to the First Aid training domain, for which a lower ontology, containing the domain specific information, was developed. The combined ontology (i.e. upper and lower) was tested with the use of domain experts.

A total of eight experts were interviewed. Three types of experts were used, in the fields of education, serious games, and First Aid instruction, respectively. Experts were asked to evaluate the combined ontology with respect to the following criteria: clarity, coherence, representativeness, accuracy, completeness, consistency, and conciseness (Gruber, 1995; Shanks et al., 2003).

The layers 'Training Environment' and 'Task Domain' of the upper ontology were reviewed by two researchers in the field of serious gaming. Two other researchers, from the field of learning and instruction, verified the 'Trainee and Didactics' area. All four researchers were recruited Human Factors Researchers employed at the Netherlands Organisation for Applied Scientific Research (TNO). The experts were asked to critically verify each concept, definition, relation, and instance of the ontology. In addition, they were asked whether the ontology accommodates the information required for automated SBT. The assistant of the session host processed suggestions on the spot. The updated graphical display of the ontology was presented to the researchers for further comments, alterations, additions or deletions. This was repeated until participants were satisfied with the result. The lower ontology was reviewed by four First Aid instructors, recruited through their First Aid associations. The followed procedure was comparable to the one explained above.

Results of the verification sessions were used to produce a new and updated version of the combined ontology. Examples of these adjustments are:

- The addition of 'adaptive strategy' as a superclass of feedback, scaffolding, etc.
- Changing 'feature' to 'observable' to describe distinctive characteristics of an element.
- The addition of 'attitude' to the types of competencies trained.
- The omission of 'closed task' and 'open task' as subclasses for tasks.
- Several First Aid procedures were refined with minor corrections.

7 Using the ontology in an application

Developing a fully automated SBT system in First Aid was beyond the scope of the project. However, we tested the usefulness of the upper and lower ontology for one of the important components of SBT: automated scenario generation. Automated SBT should be able to select or design a scenario that fits the learning needs and competency level of the trainee. Furthermore, scenarios should provide realistic situations sequenced into a coherent storyline. In addition, there should be sufficient variability across scenarios to make sure that the trainee is (1) motivated to pursue training and (2) able to recognise generic features and regularities to develop suitable schemata. The ontology was used to develop a scenario generator. The behaviour of the scenario generator, and the way it uses the ontology to produce that behaviour, is explained below.

First, the scenario generator generates an action sequence that should be performed by the learner. For example, the generator may produce the following action sequence for the learner: [Check_safety, turn_off_hot_device, ask_victim_mental_state, calm_victim, ask_victim_physical_state, check_burn, cool_burn, bandage_burn, instruct_on_aftercare].

To create this action sequence, the scenario generator uses two inputs that it receives from other system components: the learning goal and the desired difficulty level of the scenario (easy-medium-difficult). For example, the learning goal may be 'the diagnosis and treatment of a burn'. The difficulty level is set to 'easy' as the trainee is a novice and has no previous knowledge of treating this type of wounds. The scenario generator produces an action sequence by using reasoning rules that define how to decompose tasks into action plans and actions. The reasoning rules are not part of the ontology.

Once the action sequence has been established, the scenario generator uses annotations specified in the ontology to create a back story consisting of events taking place prior to the injury. For example, enabling the performance of the action sequence specified above requires a back story that involves some character inflicting a burn injury from a hot device.

Based on the back story and the action sequence, the scenario generator specifies requirements for the automated construction of a training environment. These requirements are constructed such that the resulting environment: (1) enables the virtual characters involved in the storyline to enact the back story and (2) enables the learner to perform the target task, e.g. a garage setting with virtual characters working on cars, a work bench with a soldiering iron for Marge to use, and a restroom with water taps.

The scenario generator uses declarative knowledge that is stored in the ontology to produce the behaviour described above. The ontology specifies the relationships between high level tasks, subtasks, procedures, actions, and their preconditions with regard to the virtual world (e.g. the action 'check_burn' requires the presence of a 'burn' on some 'character' and a 'burn' can only be acquired by coming into contact with an object that offers the interaction possibility of 'burn_skin'). Objects that are annotated with their interaction possibilities are called 'Smart Objects' (Kallmann and Thalmann, 1998; Gibson, 1977).

The ontology provides the declarative knowledge needed to produce a scenario plan. The scenario generator's reasoning rules use information from the upper ontology to reason with concepts like 'object', 'affordance', 'ask', and 'task decomposition'. As a result, new instances of tasks, objects, and affordances can be added to the ontology, and the scenario generator is immediately able to use these new instances in the scenario generation process.

A prototype of the scenario generator was developed and its output evaluated in an experiment. In this experiment, First Aid instructors were asked to evaluate the quality of scenario plans based on the scenario plan's (1) representativeness for real-life situations, and (2) suitability for a given learning goal and (3) difficulty level. The scenario plans were obtained from three different sources: laymen, experts, and the scenario generator. The instructors were naive about the differences between the scenario plans. The outcomes of the experiment showed that the scenario generator performed at least as good as the laymen, and experts outperformed the laymen and the scenario generator.

8 Discussion and concluding remarks

The ontology presented in this paper provides a concise and coherent specification of the core concepts in SBT along with their relationships. In addition, the ontology supports the integration of didactic principles into the automated training system by explicitly specifying the scenario variables and performance variables to be used for generating and adapting personalised scenarios. How the ontology supports the development of automated SBT was shown in an example of automated scenario generation and adaptation. The design of the upper ontology makes it applicable across training domains and system designs.

In addition to representing the knowledge required by an automated system for SBT, the ontology also facilitates consistency and clarity of terminology, thus providing a common language among stakeholders, designers, and users. When a team develops system components in parallel, the ontology provides conceptual and terminological agreement among its members. All developers within the project employ the representation provided by the ontology when describing the architecture, programming code, and so on. As a result, documents and programming code are easier to exchange, read, understand, and integrate.

The advantage of a frame-based ontology is that the intuitive structure of this format allows non-programmers to add new instances and/or concepts to the domain knowledge ontology, especially when they are able to work with user-friendly interfaces. The ontology can be easily extended with new instances (e.g. tasks, actions, roles, objects) (Muller et al., 2012). By placing them within the structure provided by the upper ontology, the system is able to reason about those new concepts and use them in the right way. This makes the ontology a user-friendly method to author the system's content. Future user-based studies with instructors may reveal whether this feature of the ontology is actually achieved.

The proposed ontology separates procedural knowledge from declarative knowledge, generic knowledge from domain-specific knowledge, and domain knowledge from design knowledge. This makes that the ontology can be reused across training domains, applications, and system designs. The initial version of the ontology was reviewed by domain experts and subsequently adjusted. For application investigated in this paper, the ontology is deemed appropriate for use in intelligent automated scenario-based training systems. Through the employment of the ontology in future applications, its usefulness can be investigated in more detail, most likely resulting in additional refinements and extensions of the current version.

References

Annett, J. (2004) 'Hierarchical task analysis', in Diaper, D. (Ed.): *Handbook of Task Analysis for Human-Computer Interaction*, CRC Press.

Atkinson, R.K., Derry, S.J., Renkl, A. and Wortham, D. (2000) 'Learning from examples: instructional principles from the worked examples research', *Review of Educational Research*, Vol. 70, No. 2, pp.181–214.

Baldwin, T.T. and Ford, J.K. (1988) 'Transfer of training: a review and directions for future research', *Personnel Psychology*, Vol. 41, No. 1, pp.63–105.

Cairncross, S. and Mannion, M. (2001) 'Interactive multimedia and learning: realizing the benefits', *Innovations in Education and Teaching International*, Vol. 38, No. 2, pp.156–164.

Cannon-Bowers, J., Burns, J., Salas, E. and Pruitt, J. (1998) 'Advanced technology in scenario-based training', in Cannon-Bowers, J. and Salas, E. (Eds): *Making Decisions Under Stress*, APA, pp.365–374.

Chi, M.T.H., Siler, S.A., Jeong, H., Yamauchi, T. and Hausmann, R.G. (2001) 'Learning from human tutoring', *Cognitive Science*, Vol. 25, No. 4, pp.471–533.

Clark, R.E. and Estes, F. (1996) 'Cognitive task analysis for training', *International Journal of Educational Research*, Vol. 25, No. 5, pp.403–417.

Corbalan Perez, G., Kester, L. and Van Merriënboer, J.J.G. (2009) 'Combining shared control with variability over surface features: effects on transfer test performance and task involvement', *Computers in Human Behavior*, Vol. 25, No. 2, pp.290–298.

Endsley, M.R. (1995) 'Toward a theory of situation awareness in dynamic systems', *Human Factors: The Journal of the Human Factors and Ergonomics Society*, Vol. 37, No. 1, pp.32–64.

Friedman Noy, N., Fergerson, R.W. and Musen, M.A. (2000) 'The knowledge model of Protege-2000: combining interoperability and flexibility', *Knowledge Engineering and Knowledge Management Methods, Models, and Tools*, Springer, pp.17–32.

Gibson, J.J. (1977) 'The theory of affordances', *Perceiving, Acting, and Knowing*, Lawrence Erlbaum, Hillsdale, NJ, pp.67–82.

Gruber, T.R. (1995) 'Toward principles for the design of ontologies used for knowledge sharing', *International Journal of Human-Computer Studies*, Vol. 43, No. 5, pp.907–928.

Hattie, J. and Timperley, H. (2007) 'The power of feedback', *Review of Educational Research*, Vol. 77, No. 1, pp.81–112.

Hoekstra, R. (2009) *Ontology Representation: Design Patterns and Ontologies that Make Sense*, IOS Press.

Houtkamp, J.M. (2012) *Affective appraisal of virtual environments*, PhD thesis, Utrecht University.

Kallmann, M. and Thalmann, D. (1998) 'Modeling objects for interaction tasks', *Eurographics Workshop on Animation and Simulation*, Berlin, Springer, pp.73–86.

Kalyuga, S. and Sweller, J. (2005) 'Rapid dynamic assessment of expertise to improve the efficiency of adaptive e-learning', *Educational Technology Research & Development*, Vol. 53, No. 3, pp.83–93.

Klein, G. (2008) 'Naturalistic decision making', *Human Factors: The Journal of the Human Factors and Ergonomics Society*, Vol. 50, No. 3, pp.456–460.

Mayer, R.E. (2008) 'Applying the science of learning: evidence-based principles for the design of multimedia instruction', *American Psychologist*, Vol. 63, No. 8, pp.760–769.

Mayes, J.T. and Fowler, C.J. (1999) 'Learning technology and usability: a framework for understanding courseware', *Interacting with Computers*, Vol. 11, No. 5, pp.485–497.

Minsky, M. (1975) 'A framework for representing knowledge', *The Psychology of Computer Vision*, McGraw-Hill, New York.

Muller, T.J., Heuvelink, A., Van den Bosch, K. and Swartjes, I. (2012) 'Glengarry Glen Ross: using BDI for sales game dialogues', *Conference on Artificial Intelligence and Interactive Digital Entertainment*, AAAI, pp.167–172.

Noy, N. and McGuinness, D.L. (2001) *Ontology Development 101: a guide to creating your first ontology*, Technical report, Knowledge Systems Laboratory, Stanford University.

Oser, R.L., Cannon-Bowers, J.A., Salas, E. and Dwyer, D.J. (1999) 'Enhancing human performance in technology-rich environments: guidelines for scenario-based training', *Human Technology Interaction in Complex Systems*, Vol. 9, pp.175–202.

Salden, R.J.C.M., Paas, F. and Van Merriënboer, J.J.G. (2006) 'Personalised adaptive task selection in air traffic control: effects on training efficiency and transfer', *Learning and Instruction*, Vol. 16, No. 4, pp.350–362.

Shanks, G., Tansley, E. and Weber, R. (2003) 'Using ontology to validate conceptual models', *Communications of the ACM*, Vol. 46, No. 10, pp.85–89.

Sweller, J., Ayres, P. and Kalyuga, S. (2011) *Cognitive Load Theory*, Vol. 1, Springer.

Tannen, D. (1993) 'What's in a frame? Surface evidence for underlying expectations', in Tannen, D. (Ed.): *Framing in Discourse*, Oxford University Press, New York, pp.14–56.

Van den Bosch, K. and Riemersma, J.B.J. (2004) 'Reflections on scenario-based training in tactical command', in Schiflett, S.G. (Ed.): *Scaled Worlds: Development, Validation, and Applications*, Chapter 1, Ashgate Publishing Ltd, pp.1–21.

Van Welie, M., Van der Veer, G.C. and Eliëns, A. (1998) 'An ontology for task world models', *Eurographics Workshop on Design Specification and Verification of Interactive Systems*, pp.3–5.

Yamnill, S. and McLean, G.N. (2001) 'Theories supporting transfer of training', *Human Resource Development Quarterly*, Vol. 12, No. 2, pp.195–208.

Young, M.F. (1993) 'Instructional design for situated learning', *Educational Technology Research & Development*, Vol. 41, No. 1, pp.43–58.