



# Collaboration et interchangeabilité entre humains réels et humains virtuels en Environnement Virtuel de Formation

Andrés Saraos Luna, Thomas Lopez, Florian Nouviale, Valérie Gouranton,  
Bruno Arnaldi

## ► To cite this version:

Andrés Saraos Luna, Thomas Lopez, Florian Nouviale, Valérie Gouranton, Bruno Arnaldi. Collaboration et interchangeabilité entre humains réels et humains virtuels en Environnement Virtuel de Formation. AFRV, Association Française de la Réalité Virtuelle, Oct 2012, Strasbourg, France. pp.1-8. hal-00765578

HAL Id: hal-00765578

<https://hal.archives-ouvertes.fr/hal-00765578>

Submitted on 14 Dec 2012

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

# Collaboration et interchangeabilité entre humains réels et humains virtuels en Environnement Virtuel de Formation

Andrés Saraos Luna  
andres.saraos\_luna@irisa.fr

Thomas Lopez  
thomas.lopez@irisa.fr

Florian Nouviale  
florian.nouviale@irisa.fr

Valérie Gouranton  
valerie.gouranton@irisa.fr

Bruno Arnaldi  
bruno.arnaldi@irisa.fr

INSA de Rennes, IRISA, UMR CNRS 6074, Inria, Université Européenne de Bretagne, Rennes, France

## RÉSUMÉ

Nous présentons un nouveau modèle d'interaction collaboratif permettant à plusieurs utilisateurs d'agir simultanément sur un même objet au sein d'environnements virtuels. Basé sur ce modèle, nous proposons une entité appelée *Perceptive Puppet* permettant de s'abstraire de la nature, réelle ou virtuelle, de l'humain dirigeant un mannequin. Cette nouvelle entité rassemble le contrôle du mannequin ainsi que l'acquisition et le stockage des connaissances qui lui sont associées. Cette abstraction permet de mettre en place l'interchangeabilité, autorisant le changement de rôle en cours de procédure tout en assurant la continuité des actions à effectuer. Afin d'illustrer ces contributions, nous montrons leurs applications dans le contexte des environnements virtuels de formation collaboratifs.

**Index Terms:** [Software and its engineering]: Software organization and properties - Virtual worlds training simulations—; [Human-centered computing]: Human computer interaction - Collaborative interaction—;

## 1 INTRODUCTION

Les Environnements Virtuels de Formation Collaboratifs (EVFC) permettent à des groupes d'apprenants de s'entraîner et d'assimiler une procédure nécessitant un travail d'équipe. L'utilisation de la réalité virtuelle pour la formation offre de nombreux avantages [5] :

- Coûts moins importants d'un environnement virtuel réutilisable par rapport à l'ensemble des machines et personnels à mettre en œuvre dans un environnement réel ;
- Pas de risques physiques pour les apprenants comme pour les machines ;
- Degrés variables et adaptables dans le travail collaboratif demandé et dans l'assistance à la formation ;
- Possibilité de répéter autant de fois que nécessaire la procédure ;
- Formation simultanée d'apprenants localisés physiquement à des endroits différents ;
- Enregistrement et analyse du déroulement de la formation afin de déterminer des points critiques dans la formation.

L'utilisation d'humains virtuels est un atout supplémentaire des environnements virtuels de formation. Ces humains virtuels définissent des entités capables de contrôler un mannequin comme le ferait un humain réel afin de remplacer celui-ci dans la réalisation d'un scénario. Ils peuvent remplacer des équipiers manquants lors de la formation et agir en tant que collaborateur contribuant à la procédure mais également en formateur procurant des indications contextuelles ou encore en perturbateur, gênant l'apprenant[23].

Dans cet article, nous introduisons dans un premier temps un modèle d'interaction collaboratif permettant la manipulation d'objets par de multiples apprenants (réels ou virtuels) et de manière simultanée. Dans la suite de cet article, nous utiliserons le terme *acteur* afin de désigner indépendamment un humain réel ou virtuel participant au travail en équipe. Le terme *mannequin* quant à lui désignera la représentation des acteurs dans l'environnement virtuel. A partir du modèle d'interaction introduit, nous définissons une nouvelle entité de la simulation, la *Perceptive Puppet*, qui permet de rassembler le contrôle et les connaissances d'un mannequin dans une structure indépendante de la nature, réelle ou virtuelle, de l'humain qui la contrôle. Cette *Perceptive Puppet* permet la définition d'un nouveau procédé architectural permettant à des acteurs d'échanger leur mannequin au sein de l'environnement. Ce mécanisme, appelé *interchangeabilité*, permet d'ouvrir de nouvelles possibilités pour les concepteurs et usagers des EVFCs comme nous l'illustrons dans des mises en situation pour la formation industrielle, la formation médicale et la formation à des situations d'urgence.

**Formation industrielle :** Les applications industrielles d'EVFC sont très variées, allant de l'apprentissage de procédures militaires [10] à la maintenance d'une machine [12]. Il est par exemple possible de former des apprenants, travaillant en équipes composées d'humains réels et virtuels, sur une machine industrielle. Dans ce contexte, le formateur peut prendre le contrôle d'un mannequin pour remplacer un humain réel et lui montrer une manipulation ou remplacer un humain virtuel pour vérifier le travail d'un apprenant sans que celui-ci ne le sache. L'interchangeabilité permet ainsi au formateur de changer de rôle au cours de la formation afin de pouvoir effectuer ces actions selon les besoins qu'il identifie.

**Formation médicale :** L'utilisation d'EVFC dans le milieu médical permet de former des étudiants à des opérations potentiellement dangereuses sans risques de blesser le patient [11, 15]. Dans ce type de simulations, le formateur effectue généralement une opération pendant que les étudiants observent et posent des questions. L'interchangeabilité permet au formateur d'échanger le mannequin qu'il contrôle et le rôle associé dans la formation avec l'un des étudiants à n'importe quel moment. L'étudiant pourra ainsi pratiquer un point particulier de l'opération sous le regard du formateur et des autres étudiants. Le formateur pourra ensuite reprendre sa place par le même procédé pour corriger l'étudiant ou poursuivre l'opération.

**Formation à des situations d'urgence :** L'entraînement à des situations d'urgence peut aider à préparer des secours à des menaces inhabituelles qu'ils peuvent rencontrer sur le terrain. La simulation d'entraînement de Dugdale [6] et l'application SécuRéVi [21] sont des exemples de formations collaboratives de ce type mettant en jeu des acteurs de natures différentes. L'interchangeabilité permet dans ce contexte d'échanger dynamiquement le rôle des apprenants afin, par exemple, de remplacer un élève en difficulté sur l'un des rôles et de lui montrer la façon de procéder.

## Contributions

Nous définissons tout d'abord dans cet article un modèle d'interaction collaboratif. Ce modèle permet à différents utilisateurs d'agir de manière simultanée sur un même objet. En se basant sur ce modèle nous introduisons ensuite une nouvelle entité, appelée *Perceptive Puppet*, qui permet de rassembler le contrôle d'un mannequin ainsi que l'acquisition et la conservation de l'ensemble des connaissances qui lui sont rattachées. L'utilisation de cette *Perceptive Puppet* permet de s'abstraire de la nature réelle ou virtuelle de l'humain la contrôlant. Utilisant ce modèle, nous proposons également un mécanisme d'interchangeabilité permettant à deux acteurs d'échanger le mannequin qu'ils manipulent au cours d'une procédure. L'acteur effectuant un échange est alors capable de continuer les actions de son nouveau rôle en accédant à la base de connaissance dans la *Perceptive Puppet*, ce qui lui permet d'enchaîner sur la procédure sans discontinuité. L'ensemble des contributions proposées dans cet article a été développé dans le cadre du projet ANR Corvette ayant pour but de proposer de nouvelles solutions dans le domaine de la formation industrielle.

## Organisation

Nous présentons dans un premier temps un état de l'art des EVFC dans la section 2 en nous concentrant principalement sur la représentation des utilisateurs, la formalisation des interactions et des modèles de connaissances. Nous détaillons ensuite le modèle unifié d'interaction Collaborative-STORM dans la section 3 avant de proposer la définition de la *Perceptive Puppet* et le protocole d'interchangeabilité en section 4. Dans la section 5 nous montrons l'implémentation qui a été faite de ces concepts avant de conclure.

## 2 ÉTAT DE L'ART

Les Environnements Virtuels de Formation (EVF) ont fait l'objet de nombreuses recherches. Toutefois, il n'existe pas à notre connaissance d'EVF ou d'EVF-Collaboratif (EVFC) proposant un mécanisme permettant de résoudre les exemples de scénarios exposés en introduction. Notre analyse des travaux existants se concentre sur différents aspects : la définition de l'utilisateur au sein d'un EVFC, les interactions entre les utilisateurs et les objets de l'environnement, puis les modèles de connaissances.

### 2.1 Les utilisateurs dans les EVFC

Les EVFC se concentrent sur la formation d'équipes d'apprenants au sein d'une même simulation. Ces simulations possèdent des degrés variés de collaboration entre les intervenants et des humains virtuels peuvent également y participer à des niveaux d'implications divers. Trois niveaux de collaboration ont ainsi été identifiés [16] :

1. La prise de conscience de la présence de coéquipiers dans le monde virtuel et la communication entre eux ;
2. Les interactions individuelles de chaque équipier avec les objets de l'environnement ;
3. Les interactions simultanées collaboratives et codépendantes de multiples utilisateurs sur un même élément.

Les EVFC s'intéressent généralement à aborder les problèmes de troisième niveau afin de proposer des solutions collaboratives. Toutefois, certains EVFC permettent à un apprenant d'interagir avec son environnement tandis que les autres apprennent de façon collaborative en observant et en communiquant entre eux. Ces rôles sont généralement définis a priori et ne peuvent pas être modifiés au cours de la formation. De telles méthodes sont utilisées aussi bien pour des formations industrielles, comme c'est le cas dans CO-VET [19], que médicales [3]. De plus, aucune interaction entre des humains réels et des humains virtuels n'est proposée dans ces environnements.

La plateforme GVT (Generic Virtual Training) permet la collaboration entre humains réels et virtuels [12]. Toutefois, les utilisateurs ne dirigent pas directement un mannequin dans la scène. Afin de dérouler la formation, les apprenants doivent sélectionner les actions à effectuer puis le gestionnaire de scénario se charge ensuite de contrôler les mannequins afin d'effectuer ces actions. Ce procédé permet ainsi de détacher l'apprenant du mannequin jouant son rôle dans la formation et présente ainsi un premier pas vers l'interchangeabilité. Toutefois, ce système est limitant puisqu'il ne permet pas à l'apprenant d'interagir directement avec son environnement mais seulement de dérouler des actions déjà programmées.

Le modèle MASCARET [4], implémenté dans l'application SécuRéVi, permet à des apprenants de collaborer avec des humains virtuels en accédant à leurs connaissances et en leur donnant des ordres d'action. L'apprenant supervise alors la formation sans prendre place concrètement au sein du monde virtuel. Toutefois, le formateur a la possibilité de jouer le rôle d'un humain virtuel et d'agir à sa place. Cette possibilité, bien que permise seulement au formateur, est un autre pas vers le mécanisme d'interchangeabilité, même si ni les apprenants, ni les humains virtuels n'en bénéficient ici.

À notre connaissance, la proposition d'une interchangeabilité totale et de manière homogène entre humains réels et virtuels n'a pour le moment pas été proposée. Pour cela, un point d'entrée commun au contrôle des mannequins doit d'abord être défini, ce qui n'a pas été abordé actuellement dans la littérature.

### 2.2 Les interactions

Différentes solutions existent afin de proposer des mécanismes d'interactions entre des objets de l'environnement. Nous pouvons notamment citer le modèle STORM (Simulation and Training Object-Relation Model [18]) qui propose une méthode hybride issue de la famille des machines à état et qui se compose d'une part d'un modèle comportemental permettant de décrire facilement des objets réutilisables et d'autre part d'un moteur d'interaction définissant une procédure standard d'interaction de ces objets avec d'autres objets de l'environnement. Des capacités et paramètres configurables sont rattachés à chaque objet STORM, permettant ainsi de décrire un comportement générique pour ceux-ci lorsqu'ils participent à des interactions. La principale limitation de cette approche est que les manipulations collaboratives ne sont pas explicitement prises en compte, rendant ainsi difficile les actions collaboratives (par plusieurs utilisateurs).

Les métaphores d'interactions pour un utilisateur servent généralement de base pour les techniques collaboratives. Des méthodes telles que la main virtuelle et le pointeur virtuel ont déjà été adaptées pour un usage collaboratif [2]. Une première méthode pour l'adaptation à un usage collaboratif est de restreindre les utilisateurs à manipuler certains degrés de liberté qui sont alors indisponibles pour les autres utilisateurs lors de l'interaction [20]. Toutefois, de telles méthodes heurtent la crédibilité et le réalisme de l'interaction.

D'autres techniques se sont intéressées à la manipulation collaborative d'objets en autorisant un accès simultané aux mêmes degrés de liberté de cet objet. On retrouve notamment de la manipulation à l'aide de plusieurs mains virtuelles, de curseurs 3D virtuels ou la métaphore du rayon virtuel [7, 8]. L'utilisation de ces techniques pose la question du retour d'information à l'utilisateur pendant son interaction afin qu'il ressente et comprenne la manipulation collaborative. Une autre méthode propose de mettre en place une communication entre deux objets lors d'une interaction [1, 9]. L'objet *outil* envoie ainsi des commandes et de nouveaux paramètres à un *objet interactif* qui traite ensuite ces demandes. Plusieurs *outils* peuvent manipuler le même *objet interactif* dans le

cadre d'une collaboration, l'objet interactif fusionne alors les demandes et informe les outils du résultat de l'interaction.

### 2.3 Modèle de connaissance

Les modèles de connaissance au sein d'équipes mixtes composées d'humains réels et virtuels ont été très peu étudiés. Toutefois, Sycara et al. décrivent plusieurs théories existantes sur ces modèles de connaissance [23]. Dans les domaines de la psychologie et des sciences cognitives [14], chaque personne possède une représentation interne du monde qui l'entoure lui permettant de le comprendre et d'y agir. Cette représentation, existant dans l'esprit d'une personne, est appelée *modèle mental* et sert de mécanisme de raisonnement permettant de prédire, de raisonner et de formuler des explications sur l'environnement.

Un modèle mental peut être vu comme une base de connaissance qui sert de support lors d'un processus de décision pour les membres d'une équipe. Il offre ainsi une base de raisonnement et de prédiction des effets de certaines actions sur le monde. Les modèles mentaux partagés sont souvent utilisés pour améliorer les performances de travaux en équipes [13, 17]. Cette théorie est notamment utilisée dans des contextes de simulations multi-agents [24] puisque la décomposition de modèles mentaux en modèles spécialisés est un concept facilement transposable en informatique. Ces propriétés rendent les modèles mentaux particulièrement adaptés pour la construction d'un modèle de connaissance destiné à une utilisation homogène entre humains réels et humains virtuels. Toutefois, aucun EVFC n'utilise actuellement de tels modèles mentaux pour des équipes mixtes de ce type.

La gestion de la connaissance est également centrale dans notre contexte afin de conserver la continuité des actions d'un mannequin lors d'un échange. En effet, lorsqu'un acteur prend le contrôle d'un mannequin au cours d'une simulation il a besoin de pouvoir récupérer les connaissances précédemment acquises par celui-ci afin d'avoir les informations nécessaires au déroulement de sa tâche. Une manière récurrente de représenter les connaissances est de considérer les humains virtuels comme omniscients sur le monde et la procédure tandis que les humains réels sont supposés avoir une connaissance a priori de la procédure [6, 12, 21]. Le caractère omniscient des humains virtuels permet d'obtenir la continuité des actions lors d'un échange pour un surcoût quasiment nul. Toutefois la question est moins triviale pour les humains réels qui ne sont pas omniscients et n'ont pas accès à cette base de connaissance.

### 2.4 Synthèse

À la vue de ces éléments, nous pouvons mettre en avant les points suivants :

- Il n'existe actuellement pas de solution unifiée pour des humains réels et virtuels permettant d'avoir des interactions collaboratives avec des objets de l'environnement ;
- Bien que brièvement abordé dans les EVFC, aucune solution ne propose actuellement de protocole d'échange permettant de résoudre les scénarios présentés en introduction ;
- Humains réels et virtuels ont toujours été considérés comme des entités agissant sur l'environnement virtuels via des moyens différents, rendant ainsi complexe tout mécanisme générique d'interchangeabilité ;
- L'omniscience des connaissances permet d'avoir une bonne continuité des actions lors d'un échange de mannequin, toutefois cette solution reste limitée dans le cas d'humains réels ne possédant pas cette omniscience.

Nous avons présenté dans cet état de l'art que le modèle STORM permettait l'interaction d'un utilisateur avec les objets présents au sein d'un environnement virtuel. Nous nous basons ainsi sur ce modèle afin de définir dans la section 3 le modèle

*Collaborative-STORM* permettant la gestion d'interactions collaboratives par différents utilisateurs. Nous montrons ensuite dans la section 4 comment ce modèle d'interaction collaborative est utilisé afin de définir l'entité appelée *Perceptive Puppet* permettant de s'abstraire de la nature réelle ou virtuelle de l'acteur dirigeant un mannequin et de mettre en place par ce biais l'interchangeabilité de mannequins entre différents acteurs. L'agencement de ces éléments architecturaux est rapidement présenté dans la Fig.1. Enfin le concept de *Perceptive Puppet* permet également de gérer la continuité des actions grâce à la gestion des connaissances associées à un mannequin. Enfin nous présentons une implémentation fonctionnelle de l'ensemble de ces principes dans la section 5.

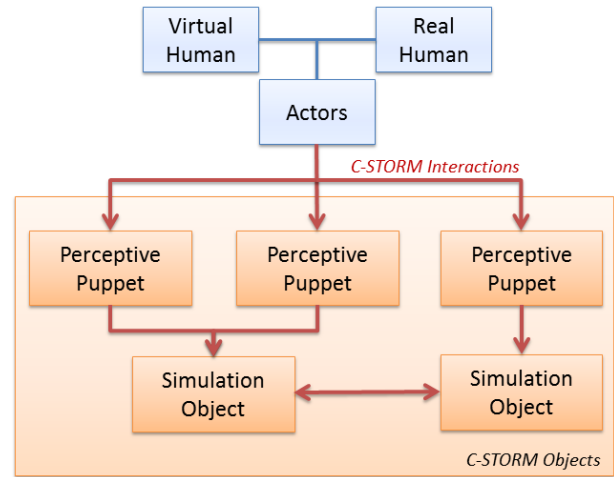


FIGURE 1: Les interactions C-STORM permettent la mise en place d'interactions collaboratives. De plus l'introduction de l'entité *Perceptive Puppet* permet de s'abstraire de la nature virtuelle ou réelle de l'acteur qui la dirige.

### 3 COLLABORATIVE-STORM : MODÈLE UNIFIÉ D'INTERACTION

Au sein des EVF, la capacité d'interaction des acteurs avec leur environnement est primordiale. Le modèle STORM (Simulation and Training Object-Relation Model [18]) propose un moteur d'interaction permettant de définir le comportement d'objets et les interactions possibles qu'ils offrent avec l'environnement. Un exemple de relation et d'interaction utilisant le modèle STORM est monté Fig.2. Afin d'interagir dans le contexte d'environnements collaboratifs, nous proposons un nouveau modèle appelé *Collaborative-STORM* (C-STORM) qui est une extension de STORM supportant la manipulation d'objets par des utilisateurs multiples et permettant ainsi d'atteindre une interaction collaborative [22].

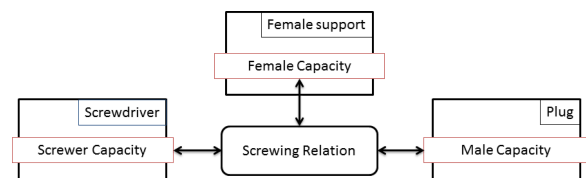


FIGURE 2: Une relation STORM : les objets comportementaux sont liés à travers cette relation.

La construction d'un modèle de manipulation collaboratif a pour but de proposer une façon générique de décrire des interactions collaboratives. De plus, dans le cadre d'EVFC, ce modèle doit être

unifié pour que son utilisation soit possible de manière transparente par des humains réels ou virtuels. La définition de ce modèle permet de créer de façon similaire des interactions entre des acteurs et des objets de l'environnement mais également entre deux objets. Les interactions sont ainsi simplifiées puisqu'elles ne reposent ni sur la nature des acteurs, ni sur celle des objets impliqués qui peuvent être des mannequins comme des objets.

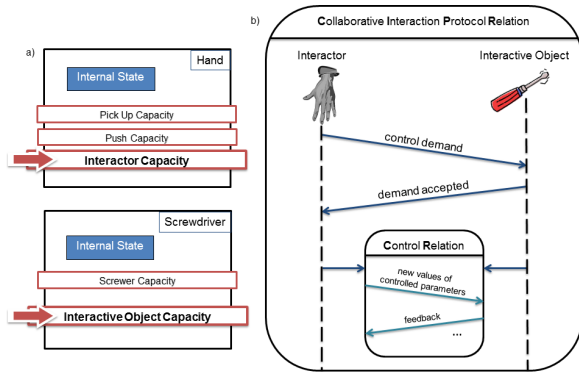


FIGURE 3: a) Deux objets C-STORM avec les capacités d'Interacteur et d'Objet Interactif. b) La CIPR lance le protocole de communication entre les deux objets C-STORM et la relation de contrôle gère ensuite les communications entre les deux éléments.

Le modèle *Collaborative-STORM* hérite des avantages du modèle STORM et s'en appuie pour décrire la manipulation collaborative. C-STORM va ainsi permettre d'interagir sur un même objet de manière simultanée pour des utilisateurs distincts, rendant possible une interaction collaborative sur cet objet. Pour cela, C-STORM agrmente les objets de nouvelles possibilités ce qui leur donne la capacité de contrôler d'autres objets mais surtout d'en partager le contrôle avec plusieurs sources. C-STORM introduit ainsi deux nouvelles capacités : la capacité *Interacteur* et la capacité *Objet Interactif*. L'*Interacteur* ajoute aux objets C-STORM la capacité de prendre le contrôle d'objets possédant la capacité d'*Objet Interactif*. Ce dernier offre alors une liste de paramètres contrôlables et traite les demandes reçues de la part de l'*Interacteur*. Afin de rendre cette interaction collaborative, chaque *Objet Interactif* peut-être manipulé par plusieurs *Interacteurs* en même temps, chaque *Interacteur* pouvant être contrôlé par un acteur différent de la simulation.

Le mécanisme de contrôle de l'objet est encapsulé dans une relation STORM appelée **Collaborative Interaction Protocol Relation** (CIPR) qui connecte un *Interacteur* à un *Objet Interactif* (cf. Fig.3). La relation une fois lancée ouvre un canal entre les deux objets qui permet de déterminer le mécanisme de contrôle sur l'*Objet Interactif*. Une relation de contrôle interne au CIPR est lancée et s'occupe de délivrer les paramètres, commandes et retours entre les deux objets tant qu'aucun signal de fin n'est reçu au niveau de la CIPR.

Dans le cadre collaboratif, chaque *Objet Interactif* peut accepter la connexion simultanée de multiples *Interacteurs*. Si les *Interacteurs* agissent sur des paramètres différents et indépendants il n'y a pas de problèmes d'accès concurrents. Dans le cas d'accès simultanés concurrents à un même paramètre par des *Interacteurs*, l'*Objet Interactif* est agrémenté d'une méthode de fusion des différentes données qu'il reçoit pour modifier le paramètre. Dans le cas le plus simple, cette politique peut être une simple moyenne entre les valeurs reçues en entrée. De la même manière, un *Interacteur* peut également être associé à plusieurs *Objets Interactifs* au même moment et ainsi contrôler l'ensemble de ces objets de façon simultanée.

#### 4 LA PERCEPTIVE PUPPET : INTERCHANGEABILITÉ POUR L'HUMAIN RÉEL ET VIRTUEL

Les objectifs de la *Perceptive Puppet* sont de :

1. Définir une abstraction commune aux humains réels comme virtuels dans un environnement virtuel ;
2. Créer un modèle permettant d'assurer la continuité des actions après un échange de rôle, peu importe la nature de l'acteur (humain réel ou virtuel) ;
3. Formaliser ce processus d'échange.

Afin de répondre à ces critères, nous introduisons la *Perceptive Puppet*, permettant d'abstraire les humains réels comme virtuels, en combinant un modèle d'interaction et un modèle de connaissance. Nous exposons les concepts servant de base à la création de cette entité dans la section 4.1, avant de nous concentrer sur les modèles constituant la *Perceptive Puppet* dans les sections 4.2 et 4.3. Enfin, nous définissons le protocole d'échange de rôle dans la section 4.4.

##### 4.1 Concept

Nous avons choisi de regrouper sous une abstraction commune les acteurs dans le cadre de la formation au travail en équipe. En effet, la nature de l'acteur (réelle ou virtuelle) est indépendante du rôle que ce dernier va jouer en tant que membre de l'équipe, c'est pourquoi nous utilisons le terme d'*acteur* pour regrouper les caractéristiques communes aux humains réels comme virtuels. Les acteurs sont dissociés des mannequins qui les représentent dans le monde virtuel et sont donc considérés comme extérieurs à ceux-ci. Ils ne peuvent donc pas interagir directement avec eux.

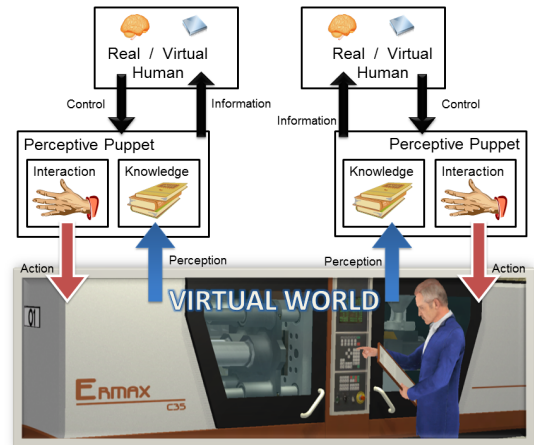


FIGURE 4: Flux de commandes et d'informations entre membres d'équipe, *Perceptive Puppet* et monde virtuel

L'entité que nous appelons *Perceptive Puppet* (Figure 4) sert de médiateur entre le monde virtuel et les acteurs formant l'équipe. La *Perceptive Puppet* est capable de percevoir le monde virtuel dans lequel elle évolue, récupérer des connaissances à partir des éléments perçus, et interagir avec d'autres objets de ce monde. Elle est en revanche dénuée de tout processus de décision et ne peut interagir d'elle-même avec le monde virtuel.

Lorsqu'un acteur contrôle une *Perceptive Puppet*, il a accès aux connaissances contenues dans celle-ci (objets du monde virtuel, actions réalisable, ...) et peut la contrôler afin de réaliser des actions sur le monde virtuel.

Une *Perceptive Puppet* est composée de deux principaux éléments :

1. Un modèle d'interaction qui décrit la manière dont l'acteur peut agir sur le monde virtuel à travers la *Perceptive Puppet* ;
2. Un modèle de connaissance, alimenté par le module de perception de la *Perceptive Puppet*. Les connaissances sont ainsi contenues dans cette dernière et sont donc indépendante de l'acteur qui la contrôle.

Dans ce contexte, l'échange de rôle entre humains réels et virtuels se définit par un échange des contrôles sur les *Perceptives Puppets* de ces derniers. Puisque la *Perceptive Puppet* contient ses propres connaissances, l'acteur la contrôlant a accès aux données concernant les actions effectuées ou pouvant être effectuées par la *Perceptive Puppet* ainsi qu'aux données concernant les objets du monde virtuel qui ont été perçus auparavant.

## 4.2 Modèle unifié d'interaction

L'objectif du système d'interchangeabilité entre deux acteurs, réels ou virtuels, est de permettre à un membre de l'équipe d'échanger sa *Perceptive Puppet* avec un autre et de poursuivre le travail d'équipe, sous ce nouveau rôle, avec l'environnement virtuel. On peut citer, comme exemple d'utilisation de cette méthode, le cas du formateur et de l'apprenant. Un formateur peut à tout moment prendre le contrôle de la *Perceptive Puppet* de l'apprenant, réaliser certaines actions puis redonner le contrôle à cet apprenant qui continuera la réalisation de la procédure.

Puisque les *Perceptives Puppets* doivent médiatiser les actions entre les acteurs et le monde virtuel, leur modèle d'interaction sous-jacent doit prendre en compte les contraintes suivantes :

- Un interacteur interagit avec une *Perceptive Puppet* ;
- Une *Perceptive Puppet* peut interagir avec des objets l'environnement ;
- Les objets de l'environnement peuvent interagir entre eux.

Nous avons défini le modèle d'interaction Collaborative-STORM dans la section 3. Dans ce modèle, nous avons introduit un moyen pour un *Interacteur* de prendre le contrôle de paramètres sur un *Objet Interactif*. Notre modèle de *Perceptive Puppet* s'appuie sur ce principe et définit :

- un acteur comme étant un *Interacteur* ;
- une *Perceptive Puppet* comme étant à la fois un *Objet Interactif* et un *Interacteur*.

Grâce à ce système, il est possible pour un acteur membre de l'équipe de prendre le contrôle de paramètres appartenant à la *Perceptive Puppet* (position global, position de la main, . . .). De plus, la *Perceptive Puppet* peut elle-même contrôler des paramètres contenus dans d'autres objets du monde virtuel, permettant ainsi la mise en place d'un lien indirect entre les acteurs et les objets du monde virtuel.

## 4.3 Modèle de connaissance

Après un échange de rôle, l'acteur concerné doit être en mesure de continuer les actions de la procédure en fonction de ce qui a été réalisé par son prédécesseur. Les connaissances contenues dans la *Perceptive Puppet* ont pour but d'informer l'acteur des connaissances acquises par ce rôle, qu'il s'agisse, par exemple, de connaître la position d'un objet dans le monde ou de la façon de réaliser une action. Les connaissances sont regroupées dans un modèle mental propre à la *Perceptive Puppet* et sont facilement accessibles et interprétables par les humains réels comme virtuels. En plus d'un formalisme de stockage connu à l'intention des humains virtuels, nous permettons la mise en place de différentes méthodes d'accès et événements relatifs aux connaissances par l'humain réel. Ces accès peuvent aller d'une simple interface graphique dans l'application à l'utilisation d'un périphérique d'affichage supplémentaire, comme par exemple l'utilisation d'un Tablet-PC.

## 4.4 Protocole d'échange

Ce protocole formalise l'interchangeabilité entre deux acteurs. Le protocole d'échange spécifie qui est à l'origine de la demande et à quelle *Perceptive Puppet* il souhaite accéder et est paramétrable. Il est par exemple possible de gérer des niveaux d'accès, de bloquer ou autoriser temporairement les échanges (durant la réalisation d'une action par exemple), d'autoriser l'utilisateur de la *Perceptive Puppet* cible à refuser l'échange. Le protocole peut également permettre aux humains virtuels d'être à la source de la demande d'échange. Enfin, il est possible que l'échange fasse partie de la procédure d'apprentissage et soit donc déclenché par un état de cette procédure et/ou du monde. Lorsqu'une demande d'échange est effectuée par un membre de l'équipe, le protocole s'assure donc de la possibilité d'effectuer la transaction avant de réaliser l'échange demandé.

## 5 IMPLÉMENTATION

Nous présentons dans cette section une implémentation du modèle unifié d'interaction collaborative, du concept de *Perceptive Puppet* et du protocole d'échange qui en découle. Cette implémentation se base sur une version modifiée de GVT<sup>1</sup>, plateforme logicielle pour environnements virtuels de formation. Ce choix est motivé par l'utilisation à échelle industrielle de GVT, ce dernier s'appliquant à des domaines variés allant des procédures militaires à l'utilisation dans le domaine ferroviaire. Notre version, GVT-Corvette, présente de plus la gestion de procédures collaboratives en équipe hétérogènes (humains réels et virtuels). GVT-Corvette, étant ainsi adapté aux formations collaboratives, présente un système client(humain réel)/serveur.

GVT [12] contient deux principaux moteurs :

1. Le moteur de scénario, qui décrit et déroule les étapes de la procédure et permet de répartir les tâches à effectuer en fonction des rôles de chacun ;
2. Le gestionnaire de relations qui contient les éléments de l'environnement virtuel ainsi que les comportements leur correspondant. Son implémentation est basée sur le modèle STORM, sur lequel le modèle Collaborative-STORM que nous avons présenté dans la section 3 s'appuie.

Le modèle d'interaction, la *Perceptive Puppet* et les humains réels et virtuels sont ajoutés à la plateforme GVT-Corvette comme nouveaux objets STORM dans l'environnement. Nous commençons donc par décrire notre implémentation du modèle d'interaction dans la section 5.1. Dans la section 5.2, nous explicitons l'implémentation des éléments constituant la *Perceptive Puppet*. Le protocole d'échange est, quant à lui, décrit dans la section 5.3 alors que notre scénario de test validant nos concepts est présenté en section 5.4.

### 5.1 Modèle unifié d'interaction

Dans notre implémentation du modèle d'interaction, nous avons créé et testé plusieurs types de paramètres permettant la manipulation de un à trois degrés de libertés de manière simultanée sur un objet. Nous avons également ajouté des modèles de fusion de données pour ces types qui réalisent pour le moment une moyenne des données d'entrées. Etant donné le caractère client/serveur de GVT-Corvette, nous avons réalisé une implémentation plus spécifique des *Interacteurs*, *Objets Interactifs* et paramètres contrôlables. Ainsi, lorsqu'un *Interacteur* envoie une valeur à un paramètre dont il possède le contrôle, cette valeur est également envoyée au serveur et aux clients. Chaque paramètre contrôlable effectuera donc

1. Generic Virtual Training – <http://www.gvt-nexter.fr/>



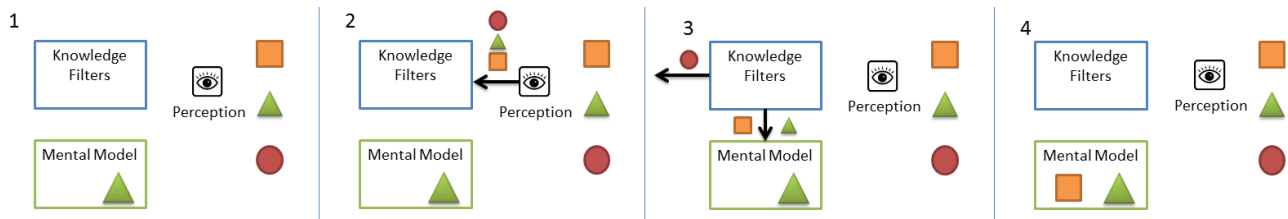


FIGURE 6: Acquisition de connaissances par la *Perceptive Puppet*. La *Perceptive Puppet* possède des connaissances antérieures sur le monde (1). Elle perçoit de nouvelles connaissances (2) et envoie ces informations à travers ses filtres qui conservent les informations pertinentes (3) et les intègrent ou les mettent à jour dans la base de connaissances (4).

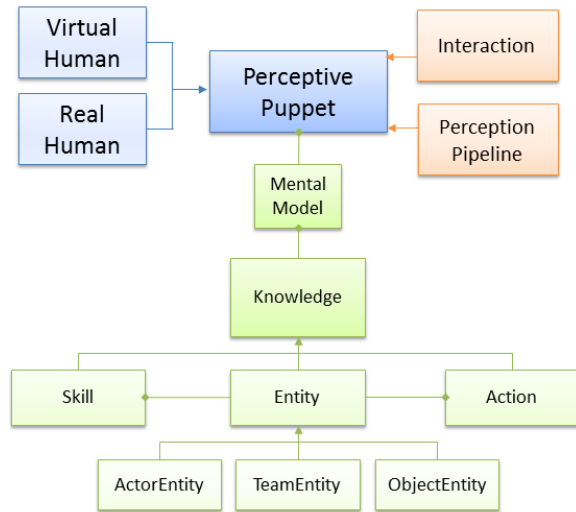


FIGURE 5: Architecture de l'implémentation

localement la moyenne des valeurs qu'il aura reçues des *Interactors*. Notre implémentation permet par exemple d'agir sur la position d'un objet ou sur l'angle de rotation de ce dernier autour d'un axe.

## 5.2 Perceptive Puppet

L'architecture proposée pour l'implémentation des membres de l'équipe et de la *Perceptive Puppet* (Fig.5) est divisée en 3 parties :

- Le modèle mental (*Mental Model*) regroupe les connaissances de la *Perceptive Puppet*, triées selon leur type ;
- Les membres de l'équipe sont liés à la *Perceptive Puppet*. Ils peuvent lui envoyer des commandes, accéder à ses données et en recevoir des évènements ;
- L'interface avec l'environnement (*Environment Interface*) est le lien entre la *Perceptive Puppet* et l'environnement virtuel. Cette interface est composée d'un module de perception et d'un module d'interaction.

Le module de perception et le modèle mental assurent l'acquisition et le stockage des connaissances par la *Perceptive Puppet*, permettant une continuité des actions plus intuitive après un échange de rôle. Le module d'interaction permet une médiation unifiée des actions effectuées par les membres de l'équipe sur le monde virtuel.

### 5.2.1 Perception

La *Perceptive Puppet* reçoit régulièrement des connaissances sur l'environnement via le module de perception. Dans notre implé-

mentation, nous utilisons un lancer de rayon paramétré par l'angle et la distance de visée comme module de perception. Les connaissances reçues sont transmises aux filtres de la *Perceptive Puppet* avant d'être rejetées ou d'alimenter le modèle mental. Dans notre implémentation, nous avons implémenté les filtres suivants :

- AllPass : Accepte toutes les connaissances ;
- NoPass : Refuse toutes les connaissances ;
- ObjectName : Accepte les connaissances sur les objets dont le nom est contenu dans le filtre ;
- ObjectType : Accepte les connaissances sur les objets dont le type est contenu dans le filtre.

Les filtres de la *Perceptive Puppet* sont ordonnés. Une connaissance fournie par la perception passera dans le premier filtre qui pourra la refuser ou l'accepter. La connaissance sera alors testée par les filtres suivants selon le même schéma. Si à l'issue du passage à travers tous les filtres l'un d'entre eux a accepté la donnée, elle est transmise au modèle mental qui la trie selon son type. Une connaissance passera systématiquement à travers tous les filtres car elle peut en satisfaire plusieurs ayant un traitement spécifique à effectuer. On peut, par exemple, chercher à connaître la position de deux objets du même type. Il faut alors pouvoir décrémenter le compteur de recherche à chaque nouvel objet du bon type perçu. Un exemple d'acquisition et de filtrage de nouvelles connaissances est montré dans la Fig. 6.

### 5.2.2 Connaissances

Nous avons réalisé une implémentation basique du modèle mental. Les connaissances sont séparées en fonction de leur type (objets, réalisation des actions). Nous utilisons le moteur de scénario de GVT-Corvette comme source de connaissance concernant les actions à effectuer.

Les connaissances sur les objets contiennent le nom de l'objet, sa position, la date de perception et potentiellement toute information contenue par l'objet dans le gestionnaire de relations de GVT-Corvette (propriétés, type, comportements possibles, ...). Parmi les informations sur les objets issues de GVT-Corvette, notre implémentation supporte uniquement leurs propriétés. Nous avons également mis en place une première version d'interface utilisateur (Fig.7), permettant à l'utilisateur réel de parcourir les données de la *Perceptive Puppet* qu'il contrôle et d'avoir des retours visuels sur les objets contenus (mise en évidence d'un objet lors de l'acquisition d'une nouvelle connaissance sur cet objet, mise en évidence d'un objet lors de l'accès à la connaissance de cet objet).

Les connaissances sur la réalisation des actions concernent les interactions et contiennent les paramètres mis en jeu dans l'interaction (objet interactif, paramètre à contrôler et son type, état final visé). Elles sont pour le moment exploitées par les humains virtuels afin qu'ils soient capables de réaliser une interaction de façon automatique. Ils peuvent ainsi choisir comment réaliser une interaction s'ils possèdent plusieurs connaissances à ce sujet.

### 5.2.3 Interaction

La *Perceptive Puppet* s'appuie sur le modèle d'interaction (voir section 5.1) afin de devenir médiateur entre les membres de l'équipe et l'environnement virtuel. Elle est donc à la fois *Interacteur* et *Objet Interactif*. Les membres de l'équipe contrôlent la *Perceptive Puppet* via ses paramètres contrôlables et ces derniers nourrissent les *Interacteurs* correspondants pouvant eux-mêmes contrôler d'autres *Objets Interactifs* du monde virtuel. Ceci s'exprime par exemple par le déplacement de la main de la *Perceptive Puppet* par l'humain réel ou virtuel, main qui est liée à un objet de la scène et déplace donc ce dernier en conséquence.

### 5.3 Protocole d'échange

Le protocole d'échange a été défini comme une relation GVT-Corvette. De façon analogue à l'interaction, une relation d'échange est déclenchée si une *Perceptive Puppet* compatible est sélectionnée et si la relation d'échange en valide la faisabilité (validation par l'humain réel contrôlant la *Perceptive Puppet* cible par exemple).

Dans notre implémentation actuelle, nous avons certains critères d'utilisation :

- Les humains virtuels acceptent toujours un échange ;
- Les humains virtuels ne demandent jamais un échange mais prennent le contrôle d'une *Perceptive Puppet* si celle-ci n'est pas contrôlée ;
- Les humains réels peuvent accepter ou refuser un échange via l'apparition d'une boîte de dialogue sur la fenêtre de l'humain réel ;
- Les échanges sont possibles à tout moment.

### 5.4 Scénario de test

Nous avons testé l'ensemble de nos contributions au sein d'un scénario type qui nous a été fourni par le partenaire *Nexter Training*<sup>2</sup> dans le cadre du projet ANR Corvette. Ce scénario test se déroule dans un contexte militaire et traduit des besoins réels identifiés dans le cadre de la formation virtuelle. Il met en scène une équipe de cinq soldats pouvant être des apprenants et devant réaliser une procédure spécifique visant à effectuer un tir avec un véhicule militaire. Ils doivent donc suivre une série d'actions coordonnées sur différentes parties du véhicule, chaque soldat ayant un rôle défini et des actions que lui seul peut effectuer.

Au lancement de l'application, les humains réels doivent choisir un rôle de départ. Ils obtiennent le contrôle de la *Perceptive Puppet* correspondante lorsque la procédure démarre. Les rôles qui n'ont pas été choisis par des humains réels sont pris par des humains virtuels, qui réaliseront automatiquement les actions décrites dans le scénario. Chaque membre de l'équipe, réel ou virtuel, peut alors réaliser les actions de la procédure qui correspondent à son rôle actuel en ayant recours au modèle d'interaction lorsque des étapes de ce type sont présentes. Les humains réels peuvent également profiter de l'interface d'accès aux données perçues (Fig.7) par la *Perceptive Puppet* qu'ils contrôlent et demander un changement de rôle en cliquant sur le mannequin qu'ils souhaitent contrôler (Fig.8). Les membres de l'équipe peuvent échanger leur rôle à tout moment de la simulation et poursuivre la procédure. Ce scénario de test valide le fonctionnement de l'interchangeabilité, basée sur le concept de *Perceptive Puppet*.

## 6 CONCLUSION

Nous avons tout d'abord présenté dans cet article le modèle d'interaction C-STORM qui permet à plusieurs utilisateurs d'agir simultanément et collaborativement sur un même objet. Nous avons utilisé ce modèle afin de définir une entité, la *Perceptive Puppet*, qui

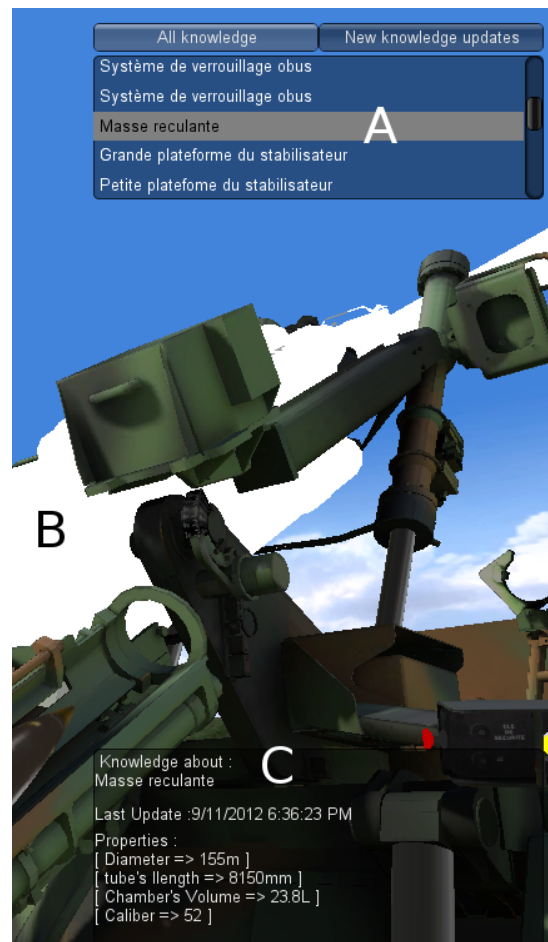
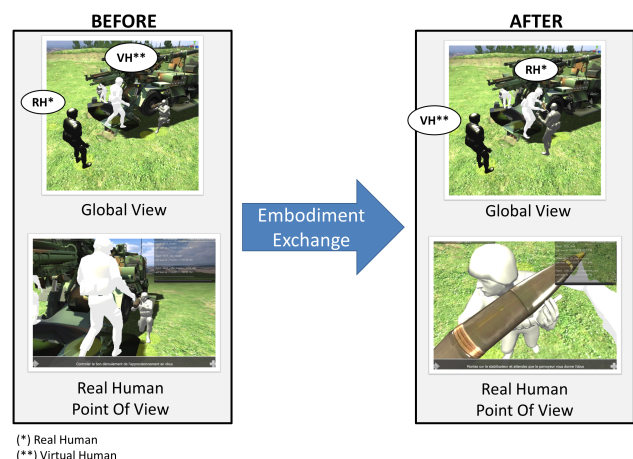


FIGURE 7: Capture d'écran de l'interface d'un utilisateur réel accédant à une connaissance de la *Perceptive Puppet* qu'il contrôle. La connaissance sur le canon (masse reculante) est sélectionnée (A). Le canon est mis en évidence en conséquence (B), et les connaissances le concernant sont affichées (C).



(\*) Real Human  
(\*\*) Virtual Human

FIGURE 8: Un humain réel effectue un échange avec un humain virtuel et réalise une action avec son nouveau rôle (prise de l'obus).

2. <http://www.nexter-group.fr/fr/filiales/nexter-training/>

permet de mettre en relation un acteur, réel ou virtuel, et son man-



nequin associé dans la simulation. Cette entité regroupe le contrôle du mannequin ainsi qu'une perception et une base de connaissances associée. Grâce à l'utilisation de notre modèle d'interaction, la gestion de la *Perceptive Puppet* et l'accès à ses connaissances sont découplés de la nature de l'acteur qui la dirige. Ce dernier point permet d'obtenir un modèle homogène d'interaction qui permet de s'abstraire notamment de la nature des acteurs présents dans la simulation.

L'introduction de ce concept permet d'adresser différentes problématiques qui n'ont pour le moment pas été abordées dans la littérature. Nous avons ainsi présenté l'interchangeabilité. En utilisant le modèle de *Perspective Puppet* nous permettons ainsi à deux acteurs de l'EVFC d'échanger le contrôle qu'ils ont sur leurs mannequins respectifs. La connaissance de la *Perspective Puppet* contribue à conserver la continuité des actions des acteurs intervenant dans le scénario. Ce mécanisme permet ainsi de résoudre les exemples de scénarios présentés en introduction.

Enfin, ces modèles ont fait l'objet d'une implémentation fonctionnelle au sein de la plateforme GVT. Cette implémentation a permis de valider les concepts portés par la *Perceptive Puppet* au cœur d'un EVFC existant et d'en tester les possibilités. Nos travaux vont à présent s'intéresser à évaluer l'apport de ces concepts auprès d'utilisateurs réels. La mise en place de ces études va permettre de tester les interactions avec les connaissances intégrées à la *Perceptive Puppet* et de voir quels moyens privilégier pour les rendre facilement assimilables.

#### ACKNOWLEDGEMENTS

Ces recherches ont été soutenues par l'Agence Nationale de la Recherche dans le cadre du projet CORVETTE (ANR-10-CONTINT-CORD-012).

#### RÉFÉRENCES

- [1] L. Aguerreche, T. Duval, and B. Arnaldi. A description of a dialog to enable interaction between interaction tools and 3D objects in collaborative virtual environments. In *VRIC 2009*, pages 63–73, Laval, France, 2009.
- [2] L. Aguerreche, T. Duval, and A. Lécuyer. Comparison of three interactive techniques for collaborative manipulation of objects in virtual reality. In *CGI 2010 (Computer Graphics International)*, Singapour, Singapour, 2010.
- [3] P. Boulanger, G. Wu, W. Bischof, and X. Yang. Hapto-audio-visual environments for collaborative training of ophthalmic surgery over optical network. In *Haptic Audio Visual Environments and their Applications, 2006. HAVE 2006. IEEE International Workshop on*, pages 21–26. IEEE, 2006.
- [4] C. Buche, R. Querrec, P. D. Loor, and P. Chevaillier. MASCARET : pedagogical Multi-Agents system for virtual environment for training. *Cyberworlds, International Conference on*, page 423, 2003.
- [5] S. Cobb, M. D'Cruz, A. Day, P. David, F. Gardeux, E. van den Broek, M. van der Voort, F. Meijer, J. Izgara, and D. Mavrikios. How is vr used to support training in industry? : The intuition network of excellence working group on education and training. In *VRIC'08 : Proceedings of the 10th Virtual Reality International Conference*, pages 75–83, 2008.
- [6] J. Dugdale, B. Pavard, N. Pallamin, M. el Jed, and C. L. Maugan. Emergency fire incident training in a virtual world. In *Proceedings ISCRAM2004*, volume 167, 2004.
- [7] T. Duval and A. Fenals. Faciliter la perception de l'interaction lors de manipulations coopératives simultanées en environnements virtuels 3D. *communication informelle à IHM 2002*, 2002.
- [8] T. Duval, A. Lécuyer, and S. Thomas. Skewer : a 3d interaction technique for 2-user collaborative manipulation of objects in virtual environments. 2006.
- [9] T. Duval and C. Le Tenier. Interactions 3D coopératives sur des objets techniques avec OpenMASK. *Mécanique & Industries*, 5(2) :767–797, 2004.
- [10] B. Fonseca, H. Paredes, L. Rafael, L. Morgado, and P. Martins. A software architecture for collaborative training in virtual worlds : F-16 airplane engine maintenance. In *Collaboration and Technology*, volume 6969 of *Lecture Notes in Computer Science*, pages 102–109. Springer Berlin / Heidelberg, 2011.
- [11] A. Gallagher and C. Cates. Virtual reality training for the operating room and cardiac catheterisation laboratory. *The Lancet*, 364(9444) :1538–1540, 2004.
- [12] S. Gerbaud, N. Mollet, F. Ganier, B. Arnaldi, and J. Tisseau. GVT : a platform to create virtual environments for procedural training. In *IEEE Virtual Reality*, pages 225–232, Reno États-Unis, 2008.
- [13] C. M. Jonker, M. B. van Riemsdijk, and B. Vermeulen. Shared mental models : A conceptual analysis. *Coordination, Organization, Institutions and Norms in Multi-Agent Systems@ AAMAS2010*, page 41, 2010.
- [14] R. Klimoski and S. Mohammed. Team mental model : Construct or metaphor? *Journal of management*, 20(2) :403–437, 1994.
- [15] K. Kunkler. The role of medical simulation : an overview. *The International Journal of Medical Robotics and Computer Assisted Surgery*, 2(3) :203–210, 2006.
- [16] D. Margery, B. Arnaldi, and N. Plouzeau. A general framework for cooperative manipulation in virtual environments. In *Virtual Environments*, volume 99, pages 169–178, 1999.
- [17] J. E. Mathieu, T. S. Heffner, G. F. Goodwin, E. Salas, and J. A. Cannon-Bowers. The influence of shared mental models on team process and performance. *Journal of Applied Psychology*, 85(2) :273–283, 2000.
- [18] N. Mollet, S. Gerbaud, and B. Arnaldi. STORM : a generic interaction and behavioral model for 3D objects and humanoids in a virtual environment. In *IPT-EGVE the 13th Eurographics Symposium on Virtual Environments*, volume Short Papers and Posters, pages 95–100, Weimar Allemagne, 2007.
- [19] J. Oliveira, M. Hosseini, S. Shirmohammadi, M. Cordea, E. Petriu, D. Petriu, and N. Georganas. Virtual theater for industrial training : A collaborative virtual environment. *Proc. 4th WORLD MULTICONFERENCE on Circuits, Systems, Communications & Computers (CSCC 2000)*, Greece, 2000.
- [20] M. S. Pinho, D. A. Bowman, and C. M. Freitas. Cooperative object manipulation in immersive virtual environments. In *Virtual Reality Software and Technology Proceedings of the ACM symposium on Virtual reality software and technology Hong Kong, China*, page 171–178, 2002.
- [21] R. Querrec, C. Buche, E. Maffre, and P. Chevaillier. SécuRévi : virtual environments for fire-fighting training. In *Proceedings of the 5th Virtual Reality International Conference, VRIC 2003*, 2003.
- [22] A. Saraos Luna, V. Gouranton, and B. Arnaldi. Collaborative virtual environments for training : A unified interaction model for real humans and virtual humans. *Edutainment 2012 - International Conference on E-Learning and Games*, 2012.
- [23] K. Sycara and G. Sukthankar. Literature review of teamwork models. Technical Report CMU-RI-TR-06-50, Robotics Institute, Pittsburgh, PA, 2006.
- [24] J. Yen, X. Fan, S. Sun, T. Hanratty, and J. Dumer. Agents with shared mental models for enhancing team decision makings. *Decision Support Systems*, 41(3) :634–653, 2006.