

# Environnement informatique pour la conception, la production et le suivi de Serious Games

Iza Marfisi-Schottman

Université de Lyon,  
LIESP, INSA-Lyon, F-69621, Villeurbanne, France  
iza.marfisi@insa-lyon.fr

## Résumé

On assiste à un essor impressionnant des *Serious Games* (SG) dans beaucoup de domaines. Ces environnements éducatifs sont en effet très prometteurs mais ils posent aussi des problèmes en termes de coût et de temps pour les étapes de conception et de production. Nous proposons un ensemble d'outils informatiques et de démarches principalement organisées autour de composants interopérables et génériques qui visent à surmonter ces barrières.

## Introduction

En 2004, le marché des SG s'élevait déjà à 1 milliard de dollars (Sawyer 2004). Six ans plus tard, les experts parlent de 9 à 11 milliards (Alhadeff 2008) et les années à venir nous réservent une croissance exponentielle. En France, cette tendance sera d'autant plus forte qu'elle est soutenue par le gouvernement qui a souhaité financer des projets de recherche sur les SG à hauteur de 30 millions d'Euros en 2009.

Dans cet article, nous nous intéressons aux SG dont le but principal est de développer des compétences métier. Les SG conçus et développés par le LIESP ces 15 dernières années sont régulièrement utilisés pour la formation des ingénieurs de l'INSA de Lyon ou en formation professionnelle (LIESP 2010). Les retours d'usage vont largement dans le sens d'un plébiscite de ces modes pédagogiques.

De façon générale, nous privilégions ces outils pédagogiques lorsque les matières abordées concernent directement des compétences professionnelles ou lorsque les méthodes dites traditionnelles s'avèrent fastidieuses et peu motivantes pour l'apprenant (systèmes complexes, méthodologies, comportements...). Grâce aux SG, il est possible de mettre en scène des cas concrets qui ne pourraient pas être faits, ou difficilement, en grandeur nature pour que les apprenants appliquent et assimilent le contenu du cours. Nous essayons d'impliquer l'apprenant dans son propre parcours d'apprentissage avec des interactivités, des jeux de rôle ainsi que d'autres ressorts ludiques comme la compétition, le hasard, les récompenses... propices à favoriser la motivation et à activer les capacités des apprenants. Par exemple, dans le SG « Samoulean » dédié à l'apprentissage de la méthode Lean (centré sur le diagnostic et l'utilisation d'indicateurs qualité), les apprenants se retrouvent dans

la peau d'un apprenti Samouraï qui doit s'initier à la démarche Lean dans un temple et la mettre en pratique au fur et à mesure dans une entreprise japonaise. La même idée a été utilisée pour l'apprentissage des méthodes de résolution de problèmes, la gestion des flux, la gestion des stocks, la maintenance...

Nous avons pu constater, en accord avec le retour d'expérience apporté par des enseignants à l'INRP, que ces SG avaient des effets positifs sur les élèves en difficulté avec les techniques d'enseignement traditionnelles sans affecter les résultats des bons élèves.

Dans cet article, nous allons d'abord expliquer les problèmes rencontrés lors de la création de SG et identifier les besoins des auteurs. Ensuite nous exposerons nos objectifs et notre démarche scientifique à travers 3 axes principaux : la collecte de composants génériques, la mise en place d'outils d'aide à la conception et la définition d'un protocole d'interopérabilité pour intégrer les composants de différents partenaires. Les outils développés et ceux en cours de développement seront décrits pour chaque objectif. Nous finirons en rappelant l'apport scientifique de notre travail et les challenges à venir.

## Des investissements coûteux et risqués

La contrepartie des SG est sans conteste leurs coûts de production qui peuvent avoisiner celui des jeux vidéo (15 000 € par heure de jeu apprenant pour nos SG). La dépense est d'autant plus risquée pour les SG ciblés sur des compétences expertes que le public visé est très réduit. De plus, un « bon » SG est un équilibre délicat entre les aspects ludiques et éducatifs pour atteindre un apprentissage humain significatif. Cette symbiose n'est pas facile à obtenir. Dans un contexte scolaire, les SG jugés « trop ludiques » ont du mal à être acceptés comme support de travail par les enseignants. Toutes les phases de jeu doivent être justifiées comme faisant partie d'un mécanisme d'apprentissage mais ce rouage doit être transparent pour l'élève qui doit être absorbé et porté par le jeu. Si le SG n'est pas assez ludique ou si le concept n'est pas assez bien expliqué et que les élèves s'attendent à un jeu vidéo, les SG peuvent s'avérer pire qu'un cours normal (discours de Ben Sawyer à SG expo, 2008).

Pour éviter la prise de risque et réduire les coûts, les auteurs essaient de reprendre certains éléments de leurs précédents SG qui se sont montrés performants. Seulement, avec les techniques dont ils disposent actuellement, ils n'arrivent pas à extraire ces éléments de façon efficace et finissent souvent, soit par reprendre la production du SG à zéro, soit par reprendre toute l'ossature scénaristique du SG en changeant uniquement le contenu. Ceci pose problème quand il s'agit de construire des compétences différentes impliquant d'autres mécanismes d'apprentissage.

De plus, du fait de leurs coûts de production élevés, les SG sont des produits souvent dédiés à des domaines de formation touchant un large public ou/et réservés aux grosses entreprises. En France, ce sont majoritairement des multinationales et des banques qui les utilisent et quelques écoles dans le secteur de la santé, du commerce ou de l'ingénierie. En améliorant les processus de conception afin de diminuer les coûts, ces outils trouveraient sûrement leur place dans les PME et les établissements d'enseignement public.

## Objectif et démarche

Notre objectif est de proposer des environnements et des outils appropriés pour pallier les problèmes de conception et de production de SG.

Nous avons d'abord cherché à approfondir les problèmes rencontrés et les besoins des auteurs de SG. Cette étude a été faite en grande partie grâce à l'expérience et à la production interne de SG au LIESP. L'équipe de recherche participe en effet régulièrement à la création de SG avec des entreprises partenaires (SEB, Thalès, CNES, HP, CALOR, COGIX ...), complétés par des SG « maison » qui sont utilisés pour les enseignements de plusieurs départements de l'INSA. Pour ces productions, nous mettons à contribution les élèves de la filière « Génie industriel », qui prennent en charge une partie de la conception ou de la production du SG. Des groupes de 8 à 10 élèves-ingénieurs doivent répondre, comme une entreprise, aux demandes d'un vrai client. Il faut en général 3 années pour couvrir le cycle conception / scénarisation / production / retour d'usage d'un SG, du fait que ces étudiants ne sont à l'origine ni experts des compétences métiers à apprendre, ni informaticiens professionnels et qu'il faut du temps supplémentaire pour que chaque groupe s'approprie le sujet. Cela représente en tout environ 10 000 heures (8 étudiants x 380 heures x 3 ans + 450 heures de tutorat et d'accompagnement + 80 heures graphiste + 400 heures d'ingénieur). Par le biais du projet FEDER (Fond Européen de Développement Régional) « Learning Game Factory » ([www.learning-games-factory.com](http://www.learning-games-factory.com)) auquel nous participons, nous avons pu recueillir les besoins de quelques entreprises spécialisés en e-learning et confronter nos idées à d'autres laboratoires de recherche ayant des préoccupations proches.

Enfin, grâce aux des groupes de travail à l'INRP de Lyon, nous avons pu nous familiariser avec les besoins

des enseignants qui souhaitent mettre en place des SG dans leurs cours.

Les besoins recensés peuvent se regrouper en 3 catégories :

### **B1. Capitaliser et partager l'expérience acquise.**

Notre équipe de production interne et les industriels ressentent le besoin de tirer partie des SG existants et notamment de réutiliser des modules ou des interactivités qui peuvent s'appliquer à plusieurs SG pour en accélérer la production.

### **B2. Aider à la formalisation des connaissances.**

Comme pour les formations de e-learning, les auteurs des SG ressentent le besoin d'organiser les connaissances à apprendre dans un SG.

### **B3. Aider à l'utilisation des ressorts du jeu.**

De façon générale, il y a une forte demande, de la part des auteurs experts, comme des novices, pour savoir comment utiliser les ressorts du jeu comme catalyseur dans leurs SG. Comment provoquer auprès des élèves une envie d'apprendre en développant en « juste utile » le plaisir du jeu ?

Pour couvrir ces besoins nous avons organisé notre travail autour de 3 grands axes:

- 1- Nous avons décidé de capitaliser et de rendre génériques certains composants qui pourraient être repris dans d'autres SG. Il peut s'agir de mini jeux comme d'un puzzle, d'un jeu de l'oie..., d'activités plus formelles comme un QCM ou de modules fonctionnels comme une barre d'outils, des modules d'aides...
- 2- Pour aider les auteurs, nous voulons mettre en place une plateforme qui assiste les différentes phases de la conception et production. Nous avons donc planifié les phases de création d'un SG sous la forme d'une chaîne de production en nous inspirant des mécanismes des jeux vidéo, des modèles du e-learning et des méthodes industrielles de production.
- 3- Afin de créer des SG avec des composants provenant d'origines différentes, nous avons défini des normes d'interopérabilité entre la plateforme centrale du SG et les différents serveurs qui fournissent les composants.

Les méthodes et les outils développés pour chacun de ces axes seront décrits en détail dans la suite de cet article. Pour valider notre travail, nous fournirons les outils développés et une base de composants aux auteurs de SG et notamment des groupes d'élèves de la filière Génie Industriel. En comparant leur productivité à celle des groupes des années précédentes, nous aurons des indications quant aux outils les plus utiles et ceux qui peuvent être améliorés.

## **1. Capitalisation de composants, de ressources et de scénarios types**

Pour couvrir le besoin de capitalisation des acquis, nous avons créé une bibliothèque de composants de

SG. Nous avons trouvé 3 types de composants qu'il serait intéressant de rassembler :

- Les *composants pédagogiques* pour lesquels il y a un but éducatif ou ludique. Il peut s'agir de jeux de rôle, de QCM, de puzzle, d'interactivités types, de vidéo...
- Les *composants fonctionnels* qui composent une partie de l'interface ou des fonctionnalités du SG. Il peut s'agir de barre d'outils, d'outils de communication, de fonctionnalités pour le tuteur...
- Les *composants outils* qui peuvent être mis à la disposition des joueurs indépendamment du sujet du SG. Il s'agit là d'outils de prise de notes, de calcul, de dessin, de prise de décision...

Ces composants peuvent également être de différentes granularités. Nous avons repris les trois niveaux de SCORM : asset, SCO (*sharable content object*) et agrégation (Dodds et Thropp 2004) largement utilisés dans le milieu du e-learning. Les composants peuvent donc être de simples ressources ou des SG complets. Le modèle que nous avons mis au point est très ouvert, sans trop de contraintes, mais nous avons pris le parti de garder les modules assez petits pour laisser la possibilité de faire des assemblages riches et originaux.

Les composants ainsi proposés peuvent être paramétrables grâce à des outils simples pour aider les auteurs à choisir ces paramètres et les adapter à des besoins spécifiques sans avoir à modifier le code source. Ces composants sont dit génériques. Nous en avons déjà développé quelques-uns, notamment pour créer des composants paramétrables de QCM, de puzzles et de quelques minis jeux que nous utilisons souvent dans nos SG. Pour chaque éditeur, nous essayons de fournir un maximum d'indicateurs de qualité pour aider les auteurs à faire un bon paramétrage et faire ressortir le meilleur du composant. Nous mettons aussi à disposition 2 ou 3 exemples de composants pédagogiques instanciés à partir de cet éditeur pour inspirer les auteurs (Cf. figure 2).

Quand un composant est intégré à un SG il doit être possible de le paramétrer avant son lancement en fonction du profil de l'apprenant et de l'avancement du SG. De la même façon, le SG doit pouvoir récupérer les traces d'utilisation du composant pour mettre à jour le profil de l'apprenant. Pour faciliter ces échanges de données, chaque composant doit satisfaire le modèle décrit dans la figure 1 avec 4 connecteurs d'entrées et de sorties (LGF 2009) :

- **Moissonnabilité** des métadonnées du composant, c'est-à-dire la possibilité de faire des recherches précises sur les caractéristiques décrites par l'auteur.
- **Paramétrage** des données du composant pour le configurer avant son lancement. Par exemple, pour les composants génériques, l'auteur pourra utiliser cette fonctionnalité et modifier les éléments de l'interface, le contenu ou le contrôle en fonction des possibilités données par l'éditeur. Dans certain cas, et si l'auteur lui en a laissé la possibilité, le tuteur pourra lui aussi changer certains paramètres pour

l'adapter à ses élèves. Enfin, c'est grâce à cette fonctionnalité que le composant s'adapte aux actions effectuées par l'apprenant dans les composants précédents.

- **Adaptabilité** pédagogique des paramètres du composant en cours d'exécution. Cette fonction est accessible au tuteur pour qu'il puisse adapter le déroulement du SG à la volée. Dans les SG du LIESP, nous avons trouvé utile de court-circuiter certaines parties du scénario ou de donner une partie des réponses aux élèves en difficulté. De la même façon, il s'est avéré utile d'ajouter des difficultés ou des activités d'attente aux élèves qui seraient trop en avance.
- **Observabilité** des traces d'exécution et récupération des variables internes du composant. Cette fonctionnalité sert à la fois aux tuteurs qui peuvent observer l'exécution en temps réel et aux analystes qui se serviront des données après coup. Mais elle est surtout indispensable pour gérer la continuité du scénario puisque les données ainsi recueillies sont utilisées pour paramétrer les composants suivants.

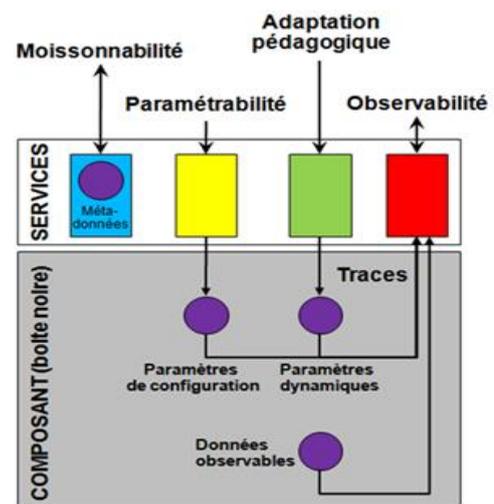


Figure 1: Format standard d'un composant

La figure 2 montre la base de composants qui est remplie, petit à petit, par des *fournisseurs de composants*. La plupart du temps, il s'agit des auteurs eux-mêmes qui, lors de la conception, sauvegardent certaines interactivités ou modules sous formes de composants. Avant de les stocker, ils doivent évidemment les mettre au format standard et notamment les décrire à l'aide de métadonnées (figure 1). Par ailleurs, nous avons prévu de créer une base de ressources graphiques et sonores accompagnées d'un outil de recherche adapté.

Nous commençons également à collecter un ensemble de modèles de scénarios pédagogiques réutilisables. Nous disposerons ainsi d'une base semblable à PALOMA qui répertorie des scénarios pour l'apprentissage à distance (Paquette 2005).

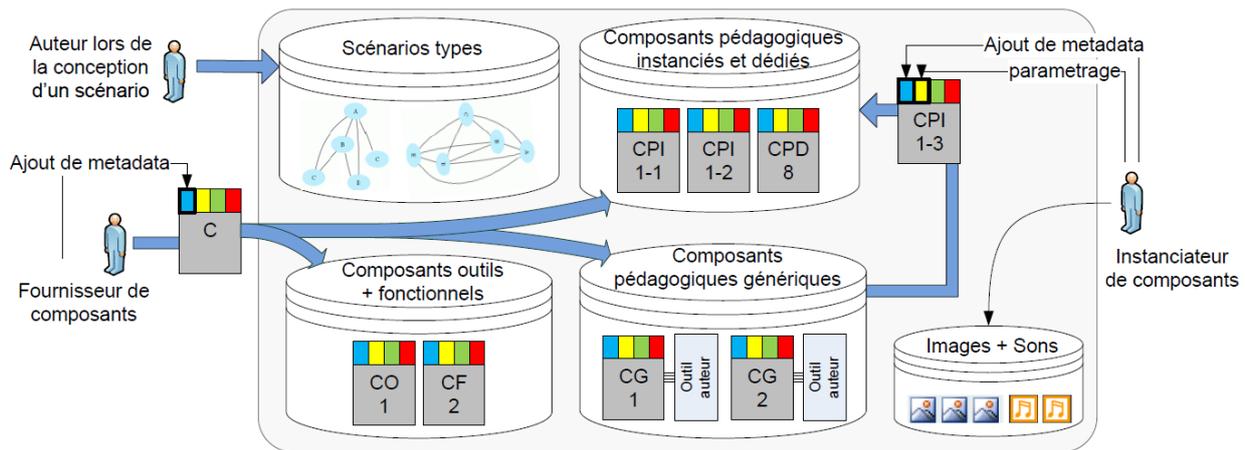


Figure 2 : Bases de composants, de ressources et de scénarios types

Nous avons, par exemple, dans la plupart de nos jeux, des modules de formation conceptuelle qui suivent toujours le même scénario : la notion est présentée avec un exemple type, l'apprenant doit ensuite résoudre quelques exercices du même type et le module se finit avec un QCM pour vérifier que les notions sont bien comprises avant de les mettre en application dans le jeu. Ce scénario type serait sûrement très utile à modéliser, permettant ainsi aux auteurs de le réutiliser dans leurs SG.

## 2. Assister les auteurs de SG

Nous avons d'abord modélisé les étapes du processus de création d'un SG en identifiant, pour chaque étape, les acteurs internes, les acteurs externes et les tâches à accomplir (Marfisi *et al.* 2009). Les grandes « phases » sont similaires à la méthode MISA proposé par Paquette *et al.* (1997) pour la conception de systèmes d'apprentissage mais nous avons choisi de viser un protocole plus léger. Seul le scénario doit être décrit sous forme de document normé, l'autre partie du SG, n'étant pas obligatoire mais conseillé. Nous avons aussi parié sur la mise en place d'outils informatiques et d'aide contextuels pour assister les concepteurs. Voici les éléments sur lesquels nous travaillons et qui seront intégrés à la plateforme d'assistance globale :

### Base de ressources et de documents types

Nous avons mis au point des formats types pour certains documents importants tels que le cahier des charges, la charte de description des scènes pour les graphistes ou les fichiers d'aides pour les apprenants et les tuteurs. Nous espérons ainsi que les auteurs seront beaucoup plus productifs.

### Outil de validation de la conception du scénario

Nous avons prévu une étape de validation à la fin de chaque étape importante de la création du SG pour rectifier les erreurs le plus vite possible. Nous avons notamment créé un outil pour valider la conception du scénario d'apprentissage avant même la phase de programmation. En effet, une fois que le scénario du SG est modélisé sous forme de graphe, il peut être testé

de différentes façons avec des outils informatiques. Dans un premier temps, l'étude des propriétés des graphes du scénario peut aider à détecter des chemins sans issue ou des parcours qui ne permettent pas aux apprenants d'atteindre les compétences ciblées. Pour des tests plus approfondis, nous avons mis au point des joueurs virtuels qui vont agir en fonction de leurs niveaux de connaissances et de profils comportementaux prédéfinis (prudent, fonceur, curieux, ...) (George *et al.* 2005, Manin *et al.* 2006). Pour le moment, cette méthode n'existe que pour les SG de type « jeu plateau » qui possède des structures très formalisées mais elle devrait pouvoir être étendue à d'autres types de jeu. L'objectif de ces simulations est d'expérimenter le déroulement d'un jeu afin de juger statistiquement de l'atteinte des objectifs pédagogiques. Sans la remplacer, cette étape permet de gagner énormément de temps par rapport à la méthode classique qui consiste à tester le système sur un échantillon d'apprenants à la fin de la réalisation et qui implique souvent de revoir la conception.

### Éditeur multi-vues de scénarios

Durant ces dernières décennies, les entreprises de jeu vidéo sont devenues expertes en organisation et réutilisation notamment en regroupant les fonctions réutilisables dans de véritables moteurs de jeux. En ce qui concerne les SG, il serait dommage de ne pas tirer partie des moteurs de jeux existants sur le marché. La plupart des moteurs de jeu requiert une connaissance de la programmation informatique, mais il en existe des plus accessibles, utilisant des graphes visuels et des mécanismes de glisser-déposer pour éditer le scénario. Ceux-ci ne permettent pas toujours des résultats visuels aussi perfectionnés mais conviennent tout à fait pour concevoir des SG. Dans la suite de notre travail nous prévoyons de reprendre un de ces éditeurs et d'y ajouter des fonctionnalités pour identifier les compétences et l'apprentissage visé dans chaque scène de SG. Nous avons déjà commencé à faire des maquettes de cet outil multi-vues en nous inspirant de principes et modèles du e-learning et notamment d'IMS-LD pour modéliser les scénarios d'apprentissage (<http://www.imsglobal.org>). Il sera

sans doute nécessaire d'ajouter un niveau plus fin, comme un modèle de tâche CTT (*Concurrent Task Tree*), pour détailler les types d'interaction utilisés.

L'éditeur de scénario proposera donc au moins 3 vues différents : **l'interface du jeu**, les **interactions liées aux mécanismes internes** et le **scénario pédagogique**. Ces perspectives peuvent être modélisées à différent niveau de granularité, du plus global (graphes conceptuels) au plus détaillé (code). De plus, cet éditeur devra fournir des outils différents en fonction du rôle des auteurs (cogniticien, expert pédagogique, codeur...) Il serait probablement intéressant de visualiser ces éléments en même temps en multi écrans et de synchroniser leurs affichages en fonction des actions et des mises à jour.

### Outil de recherche de composant

Les auteurs peuvent avoir besoin de consulter les composants dans la base à différents moments. Ils peuvent bien sûr reprendre certains composants au moment de la production mais ils peuvent aussi les consulter avant, lors de la conception, pour leur donner des idées de scénarios ou d'interactivités ludiques. Dans les deux cas, nous allons concevoir un outil de recherche spécifique à leurs besoins. Nous avons choisi de partir des métadonnées LOM (LOM 2002), largement utilisées comme base pour décrire les modules de e-learning, et d'y ajouter les extensions suivantes relatives aux SG :

- **Nature des composants** : pédagogique, fonctionnel ou outil (voir figure 2)
- **Degrés de paramétrage** possibles : aucun, générique, adaptation de la présentation...
- **Éléments de suivi** fournis par le composant : taux de complétion, exactitude des réponses ou le temps mis à répondre... Il s'agit d'informations pouvant être remontées par le composant pour mettre à jour le profil de l'apprenant en fonction de ses actions.
- **Indicateurs d'analyse** disponibles : individuel comme les éléments cliqués, les temps de pause... ou collectif comme le pourcentage, la moyenne, l'écart-type, le nombre d'abandons et d'autres statistiques sur le score d'un groupe. Ces indicateurs pourront être utilisés par les équipes d'analystes pour apprécier l'utilisation du SG.
- **Objectifs éducatifs** : acquisition de connaissances, de compétences, capacités logiques personnelles ou uniquement amusement et loisir, mise en confiance ou renforcement de lien sociaux...
- **Couverture éducative** des composants. S'agit-il d'une formation complète ou d'une introduction sur un sujet, d'une évaluation des acquis ou d'une mise en application de concepts déjà appris ?
- les **ressorts du jeu** utilisés : hasard, compétition, mimicry, vertige (Caillois 1992)
- **Mécanisme scénaristique** du composant pour que l'auteur sache s'il est possible de l'intégrer dans son SG. Nous avons identifié cinq mécanismes mais un jeu peut évidemment contenir un mélange de ces types (Babari 2000):
  - Le jeu plateau au cours duquel le joueur est confronté à une succession d'éléments aléatoires qu'il doit gérer pour arriver au point d'arrivée.

- Le jeu d'enquête pour lequel un problème initial est présenté et où le joueur est poussé à découvrir une solution par lui-même.
- Le puzzle où l'apprenant découvre chaque brique au fur et à mesure et doit ensuite reconstruire le tout en un ensemble de connaissances cohérent.
- Le jeu d'aventure dans lequel le joueur, livré à lui-même, part à la découverte d'un micro monde et le découvre en fonction de ses envies.
- Le jeu « poupée russe » dans lequel, pour différents sujets, le joueur peut choisir d'approfondir ses connaissances petit à petit.

Pour rendre la tâche des fournisseurs de composants plus facile, nous avons développé un éditeur de métadonnée LOM-SG en reprenant l'outil open source LomPad (<http://helios.liceef.ca:8080/LomPad>) développé par le LICEF. Cet éditeur est facilement maniable, donne la possibilité de cacher les éléments optionnels et génère des fichiers HTML et XML automatiquement.

### Retour d'usage et démarche qualité globale

Pour chaque composant de SG, nous prévoyons de demander à chaque utilisateur (auteurs, commanditaire, tuteur et apprenant) de répondre à des études de satisfaction. Couplé à des études statistiques sur les traces d'utilisation, notre objectif est de pouvoir faire de la réingénierie et ainsi améliorer nos composants et nos outils d'aide à la conception et production. Notre modèle se construit sur l'expérience. Plus il y aura de SG, de composants et d'apprenants et plus les auteurs auront des bibliothèques riches en exemple et en retour d'usage.

### 3. Favoriser l'interopérabilité

Les SG se démocratisent et de nombreuses entreprises, venant du jeu vidéo ou du e-learning commencent à développer des SG en collaboration étroites. Nous avons pu constater cette tendance au sein de notre projet LGF où les partenaires ressentaient le besoin de partager leurs composants.

Pour rendre cette collaboration efficace, nous avons dû définir un protocole de communication entre les différents serveurs puisque les composants doivent rester chez leurs fournisseurs pour des raisons de droits. De plus, certains de ces composants ne peuvent s'exécuter qu'avec un moteur de jeu interne. Au début de l'exécution du SG, le serveur du SG doit donc s'abonner, par le biais d'une architecture REST (Fielding, 2000) aux événements proposés par les serveurs fournisseurs des composants intégrés dans le SG. Au moment de lancer le composant, le serveur du SG fait une requête http au serveur fournisseur pour qu'il exécute le bon composant et lui envoie les paramètres à modifier en fonction du profil de l'apprenant (figure 3). Une fois l'exécution du composant terminée, ce dernier renvoie ces traces d'utilisations à son serveur fournisseur, qui les envoie, à son tour au serveur central du SG afin qu'il mette à jour le profil apprenant et les données stockées dans son propre serveur de données.

Étant donné que les composants sont stockés chez leurs fournisseurs, il nous faudra utiliser un protocole de communication comme OAI-PMH pour regrouper les métadonnées de ces différents éléments et construire un outil de recherche pour les auteurs.

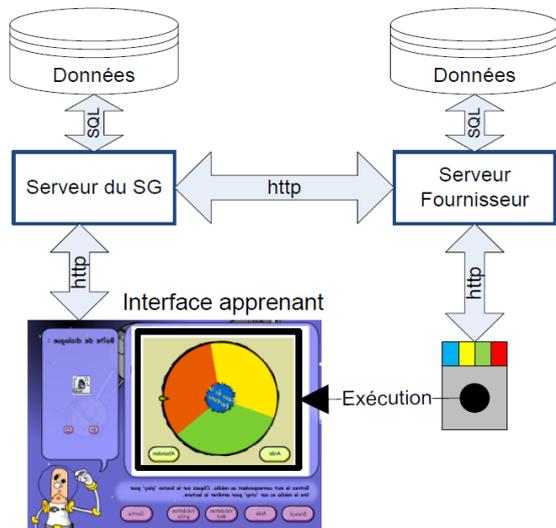


Figure 3 : protocole de communication entre serveur

## Conclusion et perspectives de travail

Cette thèse a pour but de créer des outils qui facilitent la création de SG en allégeant le travail fastidieux des auteurs et en leur donnant les moyens d'explorer des innovations pédagogiques intéressantes. Nous essayons de tirer le meilleur des jeux vidéo, du e-learning pour faire un nouveau modèle dédié aux SG. L'expérience du laboratoire et nos connaissances en informatique et en modélisation nous ont permis d'établir les fondements d'une « machine » à produire des SG, dans le souci d'assister l'auteur et non de le contraindre dans des modèles trop stricts. Le champ des SG est encore loin d'avoir atteint sa pleine maturité et nous tentons, grâce à nos travaux, de fournir un tremplin qui lui ouvrira de nouvelles perspectives et élargira les domaines d'application.

Les bases de la plateforme globale d'assistance à la conception et la réalisation de SG sont maintenant bien définies. Il nous faut continuer à développer les outils d'aide pour compléter la chaîne de production. Nous devons aussi finir de convertir nos composants pour qu'ils fournissent les 4 services requis par le format standard proposé. Le challenge suivant est de mettre au point un éditeur multi-vues de scénario et de valider le protocole de communication notamment pour le cas où le tuteur veut adapter le composant pendant son exécution. A moyen terme, nous voulons mettre au point la base fonctionnelle de tous les outils nécessaires pour suivre la création d'un SG du début à la fin en intégrant des composants externes pour vérifier l'enchaînement des étapes et la fluidité de l'exécution. Il sera ensuite possible d' étoffer les outils avec des fonctionnalités supplémentaires en fonction des besoins.

## Remerciements

Ce travail de recherche et les perspectives offertes n'auraient pas pu se réaliser sans la coopération riche et productive avec les partenaires du projet Learning Game Factory financé par les fonds européen de développement régional. (www.idf.pref.gouv.fr/dossiers/europe\_FEDER\_07\_13.htm)

## Références

- Alhadeff, E. 2008. Reconciling Serious Games Market Size Different Estimates. *Futurlab Business & Games Magazine* 9-04-2008.
- Babari M.; Mahi A.; Prévôt P. 2000. Approche générique pour la conception et l'ingénierie de jeux d'entreprise multimédias coopératifs - Cas du jeu de la maintenance multimédia, In Proceedings of the International Symposium TICE Troyes, France, 377-384.
- Caillois, R. eds. 1992. *Les Jeux et les hommes*. France : Gallimard.
- Dodds, P.; and Thropp, S. E. 2004. SCORM content aggregation model, version 1.3.1. Report of the Advanced Distributed Learning Initiative, July.
- Fielding, Roy T. 2000. Architectural Styles and the Design of Network-based Software Architectures. Ph.D. diss., Dept. of Computer Science, Stanford Univ.
- George S.; Titon D.; Prévôt P. 2005. Simulateur de comportements d'apprenants dans le cadre de jeux d'entreprise. In Proceedings of the EIAH05 Environnements Informatiques pour l'Apprentissage Humain, Montpellier, France, 389-394.
- LGF. 2009 Format standard d'un composant. Référence interne projet LGF publié le 21-07-2009.
- LIESP. 2010. Jeux d'entreprise multimédia. Site : <http://criport1.insa-lyon.fr/gi/index.php?Rub=176>
- LOM. 2002. Standard for Learning Object Metadata, IEEE 1484.12.1.
- Manin N.; George S.; Prévôt P. 2006. Virtual Learners Behaviours in Educational Business Games, Lecture Notes in Computer Science, Innovative Approaches for Learning and Knowledge Sharing, Vol. 4227, Heidelberg, 287-301.
- Marfisi-Schottman I.; Sghaier A.; George S.; Prévôt P.; Tarpin-Bernard F. 2009. Vers une industrialisation de la conception et de la production de Serious Games. In Proceedings of EIAH09 Environnements Informatiques pour l'Apprentissage Humain, Workshop Serious Games. Le Mans, France, 75-84.
- Paquette G., et al, 1997, Méthode d'ingénierie d'un système d'apprentissage (MISA). *Cognito, numéro 8, 1997*.
- Paquette, G. eds. 2005. *Innovations et tendances en technologies de formation et d'apprentissage*. Presse de l'école polytechnique de Montréal, eds S. Pierre. pp. 1-30
- Sawyer, B. 2004. Serious Games Market Size. *Serious Games initiative Forum* 01-04-2004.