



HAL
open science

Une ingénierie pour jeux sérieux

Mathieu Muratet, Fabienne Viallet, Patrice Torguet, Jean Pierre Jessel

► **To cite this version:**

Mathieu Muratet, Fabienne Viallet, Patrice Torguet, Jean Pierre Jessel. Une ingénierie pour jeux sérieux. EIAH 2009, Jun 2009, Le mans, France. pp.53-63. hal-01359776

HAL Id: hal-01359776

<https://hal.science/hal-01359776>

Submitted on 4 Sep 2016

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.

Une ingénierie pour jeux sérieux

Conférence EIAH'2009 (Environnements Informatiques pour l'Apprentissage Humain)

Mathieu Muratet*, Fabienne Viallet, Patrice Torquet*, Jean-Pierre Jessel***

* *IRIT*

** *DIDIST CREFI-T*

Université de Toulouse Paul Sabatier

118 Route de Narbonne

F-31062 Toulouse Cedex 9

Mathieu.Muratet@irit.fr

Fabienne.Viallet@iut-tlse3.fr

Patrice.Torquet@irit.fr

Jean-Pierre.Jessel@irit.fr

RÉSUMÉ. Les jeux vidéo font partie de la culture au même titre que les films, la musique ou les livres. Les jeux vidéo conçus avec un objectif autre que le simple divertissement sont appelés : jeux sérieux. Ils sont présents dans de nombreux secteurs d'activité comme l'éducation, la santé, la défense, l'industrie, la sécurité civile et les sciences. Cet article présente le jeu sérieux Prog&Play destiné à l'apprentissage de la programmation informatique. À partir de cette expérience et d'une ingénierie didactique, nous tentons de généraliser notre approche pour la construction de jeux sérieux à l'université en répondant aux questions suivantes : quelle est la pertinence de développer un jeu sérieux pour un apprentissage donné, quel jeu choisir pour quel savoir à enseigner et comment évaluer les apprentissages.

MOTS-CLÉS : jeux sérieux, ingénierie didactique, Prog&Play, programmation informatique.

1. Introduction

Depuis quelques années, les jeux sérieux prennent une place de plus en plus importante dans la sphère éducationnelle. Une illustration de ce phénomène se traduit par exemple à l'Université Paul Sabatier Toulouse III (UPS) par le nombre de projets TICE soumis au Service Universitaire de Pédagogie (SUP) qui se réclament de cette catégorie (3 projets sur 20 cette année) ou le nombre de présentations aux journées TICE Vivaldi (trois présentations de jeux sérieux

différents à destination de formations universitaires). Tous les projets ou outils sont présentés par des enseignants qui tentent d'enseigner un savoir particulier à travers l'utilisation de jeux : qu'entendent-ils donc par « jeux sérieux » ? Comment construit-on un jeu sérieux ? Quels apprentissages peut-on attendre d'un jeu sérieux ?

Dans cet article nous essayons d'aborder ces questions à travers une expérience menée à l'UPS dans le cadre d'une thèse portant sur l'élaboration d'un jeu sérieux nommé Prog&Play, destiné à faire apprécier la programmation à des étudiants novices. Après avoir défini les caractéristiques d'un jeu sérieux et les problèmes posés a priori par son élaboration, nous présentons la problématique qui nous a conduit à élaborer un jeu sérieux pour l'apprentissage de la programmation chez les novices, puis les réponses que nous avons apportées aux questionnements liés à l'élaboration de Prog&Play. Enfin, nous essayons de déterminer sous quelles conditions notre expérience pourrait servir de base à la construction de futurs jeux sérieux à l'université.

2. Jeux sérieux et contraintes de développement

Selon Michael Zyda, un jeu sérieux est « un défi intellectuel lancé sur un ordinateur selon des règles spécifiques, qui utilise le divertissement pour promouvoir la formation et l'entraînement dans les milieux institutionnels ou privés, dans les domaines de l'éducation, de la santé, de la sécurité civile, ainsi qu'à des fins de stratégie de communication »¹ [ZYDA 05]. Ainsi, tout jeu vidéo conçu avec un objectif autre que le simple divertissement peut être considéré comme un jeu sérieux.

Pour Michael Zyda, bien que le contenu pédagogique constitue un point crucial du développement d'un jeu sérieux, il « doit être subordonné à l'histoire – le divertissement doit primer »² [ZYDA 05]. L'hypothèse sous-jacente est qu'un jeu attractif et amusant encourage le joueur à progresser dans la trame du jeu. Ce faisant, le joueur intègre un grand nombre d'informations liées au contenu pédagogique du jeu sérieux. Afin d'améliorer l'immersion du joueur dans l'univers du jeu et lui faire acquérir des compétences, les jeux sérieux ont ainsi un positionnement endogène³ du contenu pédagogique.

¹«A mental contest, played with a computer in accordance with specific rules, that uses entertainment to further government or corporate training, education, health, public policy, and strategic communication objectives».

²«Pedagogy must be subordinate to story – the entertainment component comes first».

³ Malone et Lepper [MALONE & LEPPER 87] définissent une approche endogène comme un jeu où le contenu pédagogique a un lien direct avec le déroulement du scénario, en opposition à une approche exogène où les exercices n'ont aucun lien avec le déroulement du jeu.

Ainsi le jeu doit assurer deux propriétés : être un jeu apprécié par la majorité des joueurs apprenants ciblés et être compatible avec le savoir à transmettre. Un jeu mal ciblé peut conduire les apprenants à délaisser le jeu et donc à ne pas apprendre. Un savoir à enseigner incompatible avec le jeu entraîne des effets de bords tels que le joueur n'apprend pas ce qui lui est demandé. Enfin, lorsque la compatibilité entre le jeu et le savoir est assurée, il reste à évaluer l'apprentissage au travers d'un jeu sérieux.

3. Elaboration de Prog&Play

Elaborer un jeu sérieux nécessite d'abord une motivation d'ordre pédagogique. Pour nous, elle a résidé dans la constatation de difficultés récurrentes des étudiants dans l'apprentissage de la programmation qui les conduisent à abandonner la formation [ACM/IEEE 05, p. 39] [CRENSHAW et al. 08] [KELLEHER 06]. Aucun jeu sérieux adaptable à différents types d'enseignements n'adressant encore ce problème, nous avons décidé de construire Prog&Play. Nous avons abordé la question par le choix du jeu support en cherchant quels types de jeux plébiscités par nos étudiants seraient compatibles avec l'activité de programmation. Ensuite pour évaluer le jeu, nous avons déterminé un cadre théorique adapté à l'analyse de ce type de situation d'enseignement / apprentissage.

3.1. Problématique : apprentissage de la programmation

L'apprentissage de la programmation est une activité centrale dans les études en informatique et non négligeable dans la plupart des études scientifiques. Ainsi, par exemple, à l'IUT de Toulouse, la programmation est enseignée au travers de différents langages aux départements informatique (algorithmique, C, C++, Java, PHP), génie civil (VBA) et mesure physique (langage C). Au niveau international, le rapport ACM/IEEE [ACM/IEEE 05] donne une vue générale des différentes formations liées à l'informatique. Il identifie deux points clés indispensables à toutes formations en informatique : acquérir les fondamentaux de la programmation et réaliser de petits programmes.

Or cet apprentissage s'avère être difficile, en particulier pour les novices. Les étudiants sont en effet confrontés à un ensemble d'obstacles épistémologiques, comme l'apprentissage des structures de contrôle (boucle, sélection) ou la compilation des programmes. De plus, l'environnement informatique qu'ils utilisent quotidiennement n'a que peu de rapport avec celui proposé pour l'apprentissage de la programmation et ils ne perçoivent pas immédiatement les liens existant entre les deux univers. Enfin cet apprentissage se heurte à un paradoxe [GREITZER et al. 07] difficile à résoudre par les enseignants : pour programmer l'étudiant doit acquérir des compétences et des concepts, mais pour les acquérir l'étudiant doit pratiquer la programmation.

Plusieurs approches sont proposées dans la littérature pour aider à la résolution de ces difficultés. La première consiste à concevoir des exercices répondant à certains critères de pertinences comme être liés à des problèmes du monde réel, être intéressants, permettre la créativité et l'innovation... [STEVENSON & WAGNER 06]. Une deuxième approche consiste à concevoir des environnements de programmation attractifs et ludiques, comme la programmation à base de bloc [KELLEHER et al. 02] [KLOPFER & YOON 05] [MALONEY et al. 04]. Une troisième approche utilise la compétition pour stimuler les compétences des participants. Les « coding parties » sont de bons exemples de ce type d'approche. Enfin la dernière approche utilise les jeux vidéo pour motiver les étudiants. Deux formes sont possibles : la première consiste à faire réaliser aux étudiants leur propre jeu vidéo [CHEN & CHENG 07] [GESTWICKI & SUN 08] [LEUTENEGGER & EDGINGTON 07] ; la seconde consiste à apprendre en jouant à un jeu vidéo [COLOBOT 07] [ROBOCODE 07]. Prog&Play s'inscrit dans cette dernière approche. Et par opposition aux autres jeux existants, il peut être utilisé avec différents langages de programmation, dans différents contextes d'enseignement définis en fonction des choix pédagogiques des enseignants.

3.2. Choix du jeu support

Le choix du jeu vidéo support de notre jeu sérieux a débuté par une étude visant à déterminer quels étaient les jeux favoris pratiqués par les étudiants cibles de notre outil. Ainsi nous avons réalisé une enquête auprès de 181 étudiants de notre université (154 garçons et 27 filles) issus de trois formations informatiques différentes. Elle nous a montré que 86% des étudiants sont joueurs (52% chez les filles) ce qui corrobore les données de l'ESA [ESA 08] à propos du rapport des étudiants aux jeux vidéo. Pour déterminer le type de jeu pratiqué par nos étudiants, nous avons proposé neuf familles de jeu qui nous semblaient être représentatives de la sphère vidéoludique : les jeux de combat, les jeux de tir, les jeux de sport, les jeux de course, les jeux de stratégie, les jeux de rôle, les jeux de plateforme, les jeux de réflexion et les jeux d'aventure. Nous nous sommes alors rendu compte que cette classification issue de la presse spécialisée est incomplète et sujette à discussion. Pour remédier à ce problème, nous envisageons dans nos futures enquêtes de demander aux étudiants de lister leurs jeux préférés plutôt que de leur demander de les classer selon des critères discutables. Nous nous chargerons ensuite d'analyser les jeux cités selon une ou plusieurs classifications comme par exemple celle de « Game Classification » [GC 09]. Toutefois, cette enquête nous a montré que les jeux de stratégie étaient les plus joués parmi ceux proposés à nos étudiants.

Les jeux de stratégie sont-ils compatibles avec l'activité de programmation ? Dans les jeux de stratégie, le joueur donne des ordres à des entités virtuelles pour réaliser des actions comme se déplacer, construire, ou attaquer. Ces ordres sont donnés en cliquant sur une carte à l'aide de la souris. Dans la mesure où ces ordres peuvent être remplacés par des instructions de programmation, puis enchaînés et ordonnés, la transformation du jeu peut être réalisée : l'activité de jeu consiste alors à écrire des programmes plus ou moins sophistiqués destinés à commander des

entités, ouvrant ainsi la voie à différents niveaux de complexité. L'utilisation de la programmation peut devenir un ressort qui renforce l'apprentissage car elle apporte un intérêt nouveau au jeu en permettant au joueur de dépasser les contraintes d'interaction liées aux périphériques d'entrée/sortie et à ses compétences physiques. La programmation peut lui permettre de déléguer certaines tâches à ses programmes afin de se concentrer sur les aspects du jeu les plus intéressants.

Est-il possible de construire un scénario d'apprentissage à partir d'un jeu de stratégie ? Les jeux de stratégie possèdent deux modes : le mode « campagne » destiné à familiariser le joueur avec le jeu et le mode « escarmouche » qui permet au joueur de se confronter à l'ordinateur et à d'autres joueurs. Le scénario de Prog&Play exploite ces deux modes : le mode « campagne » propose plusieurs missions de complexité croissante aux étudiants, chaque mission correspondant à l'écriture d'un programme ; le mode « escarmouche » permet l'élaboration de projets dans lesquels les étudiants doivent s'affronter au travers des programmes qu'ils ont écrits.

D'un point de vue didactique, cette approche permet un enseignement au travers de la pratique. Dans le mode « campagne », l'enseignant répond à ses objectifs pédagogiques en construisant les missions. Elles proposent des exercices de complexité croissante qui permettent aux étudiants de se confronter aux difficultés de l'apprentissage de la programmation. Le mode « escarmouche » au travers d'une pédagogie par projet permet aux étudiants d'éprouver leurs compétences en programmation et en stratégie de jeu. Dans les deux modes, le ressort de l'apprentissage réside dans le désir d'atteindre les objectifs de la partie. Les jeux de stratégie semblent donc être bien adaptés à un enseignement pratique de la programmation.

Le choix du jeu de stratégie ayant été réalisé pour supporter Prog&Play, la question du développement du jeu et de son moteur s'est posée. L'objectif n'étant pas de réinventer un jeu connu, voire pratiqué par les étudiants, il nous a semblé opportun d'utiliser un moteur de jeu existant, gratuit et open source, adaptable à notre problématique. Ainsi, nous utilisons le jeu de stratégie temps réel multijoueur « Kernel Panic » [KP 08] qui repose sur le moteur de jeu Spring. [MURATET et al. 09] présentent en détail sa mise en œuvre.

3.3. Le cadre théorique d'analyse du jeu

L'analyse de ce type d'enseignement requiert le recours à une ingénierie didactique. Cette ingénierie, présentée dans [MURATET et al. 09], repose sur le cadre théorique défini en didactique des mathématiques par Cobb et al. [COBB et al. 03].

3.3.1. Le cadre théorique de Cobb

L'ingénierie didactique proposée par Cobb et al. [COBB et al. 03] se propose d'étudier d'un point de vue théorique et de façon systématique, des formes

particulières d'apprentissage mises en œuvre au travers d'un protocole expérimental supporté par des moyens d'enseignement adéquats. Le processus mis en œuvre est itératif. Ainsi, le protocole expérimental est décliné au sein de plusieurs expérimentations se déroulant dans différents contextes afin de faire varier des paramètres et d'améliorer au fur et à mesure le processus d'apprentissage. Cependant, la finalité de l'étude ne réside pas dans la construction d'un protocole expérimental qui fonctionne empiriquement. Il s'agit de pouvoir expliquer pourquoi un protocole fonctionne et de suggérer comment il peut s'adapter à de nouveaux contextes. Ainsi, les protocoles expérimentaux conçus ne sont pas bricolés en vue de perfectionner un enseignement donné, mais constituent un creuset pour générer des études théoriques.

L'ingénierie didactique s'inscrit dans le cadre théorique de l'écologie de l'apprentissage définie comme étant « un système complexe d'interactions qui engage de multiples éléments de différents types et niveaux ». Elle n'étudie pas l'apprentissage du point de vue strict de l'apprenant mais dans le cadre complexe du système éducatif et l'envisage sous la forme d'un système interactif afin d'adresser la complexité de l'activité d'enseignement/apprentissage. La méthodologie associée à l'ingénierie didactique est pragmatique. Elle repose sur une équipe de chercheurs qui enseignent ou collaborent avec des enseignants dans différents contextes. Au cours de chaque expérimentation, les chercheurs analysent l'ensemble des données obtenues et procèdent à des révisions du protocole. Chaque expérimentation est considérée comme un cas paradigmatique d'une classe plus large de phénomènes. Ainsi l'ingénierie didactique fournit un cadre d'analyse pour des formes d'apprentissage spécifiques, supporté par des moyens d'enseignement adéquats.

3.3.2. Application du cadre théorique de Cobb à Prog&Play

Dans le cadre de Prog&Play, l'ingénierie didactique nous permet de vérifier la théorie selon laquelle l'apprentissage de la programmation est favorisé par la pratique du jeu vidéo. Ce cadre théorique nous conduit à mener une analyse multifactorielle d'expérimentations de Prog&Play menées dans différents contextes auprès de différents enseignants. La figure 1 illustre les différentes étapes de cette évaluation basée sur les travaux de [CHEN & CHENG 07], [GESTWICKI & SUN 08], [GREITZER et al. 07], [LEUTENEGGER & EDGINGTON 07] et [VIAU 97].

Étant donnée la problématique sous jacente à Prog&Play, l'évaluation des expérimentations est centrée sur la motivation des étudiants, le savoir enseigné et appris, et l'activité conjointe de l'enseignant et des étudiants [SENSEVY 07]. L'évaluation de la motivation des étudiants est basée sur le modèle de [VIAU 97]. Elle est effectuée au moyen de questionnaires diffusés avant et après l'expérience. Pour mesurer l'évolution des savoirs chez les étudiants, nous procédons en deux temps : avant et après l'expérience. Ces mesures comportent des évaluations ad hoc basées sur l'analyse des savoirs à enseigner, ainsi que la prise en compte des résultats scolaires. Pour l'analyse de l'activité conjointe des étudiants et des enseignants, nous procédons à l'observation de séances d'enseignement centrée sur l'enseignant et un étudiant. Du point de vue enseignant, nous étudions quelle est la

part de l'enseignement consacrée au jeu et celle consacrée à l'enseignement du savoir proprement dit afin de pouvoir effectuer une comparaison avec l'enseignement ordinaire. Du point de vue de l'étudiant, nous observons son activité tant du point de vue de sa participation au cours que du nombre et de la qualité des programmes qu'il crée en séance et en dehors. Nous avons établi une grille d'évaluation des programmes basée sur des critères d'ordre didactique et de pratique professionnelle [SMITH & CORDOVA 05].

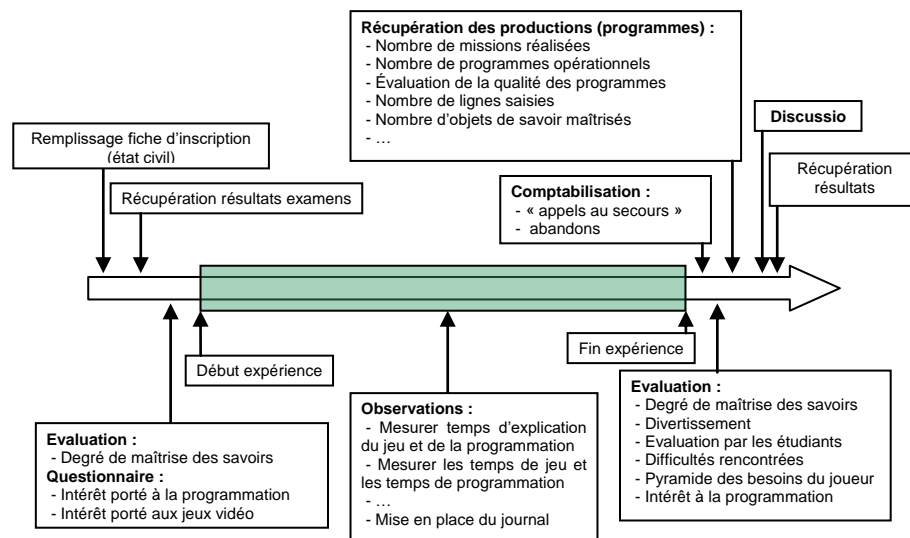


Figure 1. Organisation temporelle de l'évaluation.

Actuellement, nous avons procédé à deux expériences où nous sommes les enseignants. La première a eu lieu au département informatique de l'IUT de Toulouse et a simplement permis de valider les fonctionnalités de Prog&Play et de l'ingénierie pédagogique proposée. Cette expérience a été conduite auprès de 15 étudiants volontaires ayant des niveaux scolaires et des pratiques de jeux très différents et a montré une réelle motivation pour la plupart d'entre eux. La seconde expérience est actuellement en cours à l'UPS auprès d'étudiants volontaires inscrits en L1 sciences. L'objectif de cette expérience est de valider la portabilité de Prog&Play à d'autres enseignements et d'expérimenter la faisabilité de nos critères d'évaluation. Ainsi, étant donné l'avancement de nos travaux, nous n'avons à ce jour pas encore pu procéder à une évaluation complète de Prog&Play telle que présentée ci-dessus. Notre objectif est de pouvoir diffuser Prog&Play à la rentrée scolaire prochaine. Nous sommes actuellement en pourparlers avec des enseignants de l'IUT Informatique de Blagnac, de l'IUT SeReCom de Castres et de l'UPS pour intégrer Prog&Play à leur formation officielle. La mise en œuvre de ces expériences nous permettra d'évaluer la prise en main du jeu par des enseignants externes au projet et pas toujours volontaires. L'hétérogénéité de ces formations et la diversité de leurs enseignants nous permettraient d'évaluer notre approche dans des contextes distincts.

4. Perspectives

Les trois questions auxquelles nous avons été confrontées pour élaborer Prog&Play se posent lors de la construction de jeu sérieux : quelle est la pertinence de développer un jeu sérieux pour un apprentissage donné, quel jeu choisir pour quel savoir à enseigner et comment évaluer les apprentissages ?

Notre première expérience au SUP de l'UPS nous montre que la motivation des enseignants pour développer un jeu sérieux concerne les difficultés d'apprentissages liées au savoir à enseigner. L'enseignement classique délivré ne permet pas aux étudiants de surmonter certaines difficultés ou d'acquérir des automatismes en termes d'apprentissage. Dans ces cas, les enseignants pensent que les jeux sérieux offrent une perspective d'apprentissage innovante et alternative dont la finalité est de motiver les étudiants et de les éveiller à une problématique spécifique, tout en leur permettant d'acquérir des mécanismes de résolution de problèmes, de façon parfois systématique.

Ainsi, la question de la pertinence de développement d'un jeu sérieux est envisagée en termes de difficulté d'apprentissage et non en termes de transposabilité d'un savoir. Pourtant, la question mérite d'être posée : est-ce que tout savoir peut se prêter à un apprentissage par le jeu ? Actuellement il n'existe aucun critère permettant d'effectuer cette discrimination.

Le choix du jeu support est également un point critique. Deux solutions sont envisageables : prendre un jeu existant ou construire un nouveau jeu. Chercher à adapter un jeu existant permet d'orienter son choix vers des jeux pratiqués par le public cible et de rester cohérent avec le point de vue de Zyda selon lequel le divertissement doit primer afin que l'apprentissage puisse se réaliser. Cependant, le problème de la compatibilité entre le jeu et le savoir à enseigner se pose : rien ne garantit l'adaptabilité du jeu à la pédagogie souhaitée. De plus, à notre connaissance, il n'existe aucun modèle permettant de détecter a priori les éventuels effets de bord d'un jeu sur l'apprentissage.

La seconde solution consiste à construire un moteur de jeu. Dans ce cas la priorité peut être donnée au savoir à enseigner. Le jeu peut être construit à partir d'un scénario pédagogique bien défini. Cependant, rien ne garantit dans ce cas que le jeu procure un divertissement suffisamment intense pour soutenir l'apprentissage ou qu'il soit en adéquation avec les pratiques du public ciblé. De plus les jeux existants bénéficient d'une longue expérience de développement avec laquelle il semble difficile de rivaliser. La question de la primauté du jeu sur le savoir à enseigner reste donc entière.

Pour évaluer Prog&Play nous avons choisi d'utiliser le cadre théorique de Cobb et al. [COBB et al. 03]. Ce cadre, adapté à tous les jeux sérieux permettant le développement d'une ingénierie didactique, propose une analyse multifactorielle dont la portée semble intéressante pour tout type de jeu. Nous retiendrons notamment l'évaluation de la motivation des étudiants et de l'évolution des savoirs. À ce titre, un enregistrement de la trace des actions des étudiants au sein du jeu peut

permettre une analyse fine des processus d'apprentissage mis en œuvre au cours du jeu.

5. Conclusion

L'évolution de la société conduit à la pratique régulière, voire intensive, de jeux vidéo, particulièrement chez les jeunes générations. Ainsi, le concept de jeux sérieux, destiné à supporter des apprentissages au travers de la pratique du jeu, est logiquement apparu, offrant ainsi une nouvelle forme d'apprentissage. Actuellement, l'enseignement universitaire, lorsqu'il se trouve confronté à des difficultés considérées comme irréductibles par les médias classiques d'apprentissage, investit ce domaine des possibles. La question se pose alors de l'opportunité et de la faisabilité de développer des jeux nouveaux.

Au travers d'une expérience menée dans le cadre d'une thèse sur un jeu sérieux destiné à motiver des étudiants novices en programmation, nous avons tenté de définir un cadre possible d'élaboration de jeux sérieux en répondant à trois questions clés : quelle pertinence pour le jeu, quel jeu vidéo choisir comme support et quelle performance peut-on attendre. Cette approche pose les problèmes de la compatibilité entre le jeu et le savoir à enseigner ainsi que la faisabilité de l'évaluation des processus d'enseignement apprentissages mis en œuvre.

Bibliographie

- [ACM/IEEE 05] ACM/IEEE-Curriculum 2005 Task Force, Computing Curricula 2005, The Overview Report, IEEE Computer Society Press et ACM Press, Septembre 2005.
- [CHEN & CHENG 07] Chen, W.-K., Cheng, Y. C., « Teaching Object-Oriented Programming Laboratory With Computer Game Programming », *Education, IEEE Transaction on*, vol. 50, n° 3, 2007, p. 197-203.
- [COBB et al. 03] Cobb, P., Confrey, J., DiSessa, A., Lether, R., Schauble, L., « Design Experiments in Educational Research », *Educational Researcher*, vol. 32, n° 1, 2003, p. 9-13.
- [CRENSHAW et al. 08] Crenshaw, T. L., Chambers, E. W., Metcalf, H., Thakkar, U., « A case study of retention practices at the University of Illinois at Urbana-Champaign », *39^e ACM Technical Symposium on Computer Science Education*, vol. 40, n° 1, 2008, p. 412-416.
- [GESTWICKI & SUN 08] Gestwicki, P., Sun, F.-S., « Teaching Design Patterns Through Computer Game Development », *ACM Journal on Educational Resources in Computing*, vol. 8, n° 1, 2008, p.1-22.
- [GREITZER et al. 07] Greitzer, F. L., Kuchar, O. A., Huston, K., « Cognitive science implications for enhancing training effectiveness in a serious gaming context », *ACM Journal on Educational Resources in Computing*, vol. 7, n° 3, art. 2, 2007.
- [KELLEHER 06] Kelleher, C., « Alice and The Sims: the story from the Alice side of the fence », *The Annual Serious Games Summit DC Washington*, 2006.

- [KELLEHER et al. 02] Kelleher, C., Cosgrove, D., Culyba, D., Forlines, C., Pratt, J., Pausch, R., « Alice2: Programming without Syntax Errors », *15^e annual symposium in the User Interface Software & Technology*, 2002.
- [KLOPFER & YOON 05] Klopfer, E., Yoon, S., « Developing Games and Simulations for Today and Tomorrow's Tech Savvy Youth », *TechTrends: Linking Research & Practice to Improve Learning*, vol. 49, n° 3, 2005, p. 33-41.
- [LEUTENEGGER & EDGINGTON 07] Leutenegger, S., Edgington, J., « A games First Approach to Teaching Introductory Programming », *SIGCSE Bull.*, vol. 9, n° 1, 2007, p. 115-118.
- [MALONE & LEPPER 87] Malone, T. W., Lepper, M. R., « Making learning fun: A taxonomy of intrinsic motivations for learning », *Aptitude, learning and instruction: III. Conative and affective process analyses*, 1987, p. 255-286.
- [MALONEY et al. 04] Maloney, J., Burd, L., Kafai, Y., Rusk, N., Silverman, B., Resnick, M., « Scratch: A Sneak Preview », *2^e International Conference on Creating Connecting, and Collaborating through Computing*, 2004, p. 104-109.
- [MURATET et al. 09] Muratet, M., Torguet, P., Jessel, J. P., Viallet, F., « Towards a Serious Game to Help Students Learn Computer Programming », *International Journal of Computer Games Technology*, vol. 2009, , Article ID 470590, 12 pages, 2009.
- [SENSEVY 07] Sensevy, G., *Agir ensemble – L'action Didactique Conjointe du Professeur et des Elèves*, Rennes : Presses Universitaires, p. 13-49, 2007.
- [SMITH & CORDOVA 05] Smith, L., Cordova, J., « Weighted primary trait analysis for computer program evaluation », *J. Comput. Small Coll.*, vol. 20, n° 6, 2005, p. 14-19.
- [STEVENSON & WAGNER 06] Stevenson, D. A., Wagner, P. J., « Developing real-world programming assignments for CS1 », *11^e annual SIGCES conference in Innovation and technology in computer science education*, 2006, p. 158-162.
- [VIAU 97] Viau, R., *La motivation en contexte scolaire*, Bruxelles : De Boeck, 1997.
- [ZYDA 05] Zyda, M., « From Visual Simulation to Virtual Reality to Games », *IEEE Computer*, vol. 38, n° 9, 2005, p. 25-32.

Références sur le web

- [COLOBOT 07] <http://www.ccebot.com/colobot/index-e.php> accédé le 21 Septembre 2007.
- [GC 09] <http://www.gameclassification.com/> accédé le 28 Mai 2009.
- [ESA 08] http://www.theesa.com/facts/pdfs/ESA_EF_2008.pdf accédé le 26 Août 2008.
- [KP 08] http://spring.clan-sy.com/wiki/Kernel_Panic accédé le 26 Août 2008.
- [ROBOCODE 07] <http://robocode.sourceforge.net/> accédé le 17 Avril 2007.