

# Utilisation de jeux sérieux pour enseigner les fondamentaux de la programmation

Mathieu Muratet<sup>1</sup>, Fabienne Viallet<sup>2</sup>, Patrice Torguet<sup>1</sup>, Jean-Pierre Jessel<sup>1</sup>  
Mathieu.Muratet@irit.fr, fabienne.viallet@univ-tlse3.fr, Patrice.Torguet@irit.fr, Jean-  
Pierre.Jessel@irit.fr

<sup>1</sup> IRIT (Toulouse, France)

<sup>2</sup> UMR EFTS (Toulouse, France)

## Introduction

Le jeu sérieux est une nouvelle technologie utilisée pour atteindre des objectifs variés tels que la communication, l'enseignement, le recrutement ou la formation. Dans un contexte d'enseignement, une des principales difficultés consiste à intégrer le savoir à enseigner dans la situation de jeu sans en dénaturer la composante ludique. Pour aborder ce problème, la recherche dans le domaine des jeux sérieux est transdisciplinaire et mobilise les connaissances de différentes approches scientifiques (informatique, didactique, pédagogie, psychologie, etc.). Dans cette optique, nous présentons un travail mené de concert entre informaticiens et didacticiens. Cet article présente donc quelques résultats liés à la conception de deux jeux sérieux pour l'apprentissage de la programmation (aspect informatique) et à leur évaluation en situation écologique (aspect didactique).

## Conception des jeux sérieux : la bibliothèque Prog&Play

Les deux jeux sérieux présentés dans cet article ont pour objectif de motiver les étudiants à programmer. Ces jeux sont basés sur une famille de jeu adaptée à la mise en œuvre de problèmes algorithmiques : les jeux de Stratégie Temps Réel (ou STR). Ce choix est conforté par la popularité des jeux de STR que nous avons pu vérifier au cours d'une série d'enquêtes réalisée auprès d'un millier d'étudiants. L'intégration de la programmation, comme un moyen d'interaction avec ce type de jeu, a été rendu possible grâce à la création d'une bibliothèque nommée Prog&Play. Cette dernière, interface entre le moteur du jeu et les programmes du joueur, a été conçue pour atteindre deux objectifs : **abstraire la complexité de la programmation des jeux de stratégie temps réel** pour permettre à des étudiants novices en programmation de créer, en fonction de leurs compétences, des programmes capables de piloter les entités virtuelles de l'environnement ludique et **être interopérable avec différents langages et paradigmes de programmation** pour ne pas imposer aux enseignants une approche particulière de l'enseignement de l'informatique. Ainsi, Prog&Play est une bibliothèque de fonctions

compréhensibles par les étudiants, qui donnent accès aux principales fonctions des jeux de STR, qui permet de concevoir des situations de jeu diversifiées en bénéficiant et exploitant les caractéristiques propres à chaque jeu de STR et qui est adaptable aux choix pédagogiques de l'enseignant. Actuellement la bibliothèque Prog&Play est disponible à travers six langages de programmation (Ada, C/C++, Compalgo, Java, Ocaml et Scratch) et permet donc d'aborder les jeux basés sur Prog&Play avec une approche impérative, orientée objet et fonctionnelle.

Cette architecture, **basée sur une communication inter-processus utilisant une mémoire partagée**, a donné naissance à plusieurs jeux sérieux. Le premier, intitulé *Kernel Panic Campaign*, est centré sur les fondamentaux de la programmation (affectation, fonctions, structures de contrôle). Il mise sur un scénario de jeu original, décomposé en missions, pour inciter le joueur à pratiquer la programmation. Pour atteindre l'objectif de chaque mission, le joueur doit mobiliser ses connaissances acquises lors des enseignements préliminaires pour concevoir des programmes qui résoudront les problèmes posés.

Le second jeu, intitulé *Byte Battle*, est centré sur la décomposition fonctionnelle. Il mise sur la compétition des jeux de STR multi joueurs pour motiver les étudiants. Compte tenu d'une situation de jeu initiale bien définie, chaque joueur doit imaginer sa stratégie de jeu, la décomposer en sous-problèmes, l'exprimer sous la forme d'algorithmes et l'implémenter à l'aide d'un langage de programmation en vue de la tester contre d'autres joueurs.

Que ce soit pour le jeu *Kernel Panic Campaign* ou *Byte Battle* la programmation des solutions passe par l'utilisation d'une des interfaces de la bibliothèque Prog&Play. L'ensemble de ces ressources sont disponible sur le site : <http://www.irit.fr/~Mathieu.Muratet/progAndPlay.php>.

## Évaluation des jeux sérieux

Pour réaliser l'évaluation de ces jeux sérieux, nous nous sommes appuyés sur l'ingénierie didactique de Cobb (Cobb et al., 2003). Cette méthodologie itérative consiste à construire, en collaboration avec les enseignants, une ingénierie didactique, à la mettre en œuvre au cours d'une ou plusieurs expérimentations en procédant à de multiples observations puis à analyser les résultats afin de proposer une éventuelle nouvelle ingénierie didactique destinée à tester de nouveaux éléments.

Quatre itérations ont été réalisées. Les trois premières avec le jeu *Kernel Panic Campaign* et la quatrième avec le jeu *Byte Battle*. La première itération avait pour

objectif de vérifier l'utilisabilité du jeu en contexte écologique. La deuxième consistait à vérifier la portabilité du jeu dans de nouveaux contextes d'enseignements. La troisième itération visait la diffusion du jeu pour permettre à des enseignants externes au projet d'intégrer par eux même le jeu dans leur propre situation pédagogique. Enfin la quatrième itération visait à évaluer le jeu *Byte Battle* avec son approche basée sur la compétition. Au cours de ces itérations, 11 expérimentations ont engagés 25 enseignants et plus de 400 étudiants. Le jeu *Kernel Panic Campaign* a notamment été utilisé avec trois paradigmes de programmation différents : impératif, orienté objet et fonctionnel.

Deux résultats notables ont pu être extraits de ces expérimentations. Le premier porte sur la validation de la motivation intrinsèque des jeux qui ont su inciter les étudiants à investir du temps dans la pratique de la programmation. D'une manière indirecte, ceci valide également la cohérence des jeux sérieux et leur juste équilibre entre les concepts véhiculés et le divertissement. Toutefois, nous avons pu mettre en évidence une difficulté liée à l'intégration des jeux sérieux dans la situation pédagogique. En effet, les jeux ne se suffisent pas à eux mêmes et doivent être intégrés dans une formation encadrée par un enseignant. De même qu'il est complexe d'équilibrer le scénario utilitaire avec la composante ludique d'un jeu, une attention toute particulière doit porter sur le positionnement d'un jeu au sein de la situation pédagogique. Deux extrêmes doivent notamment être évitées : laisser les étudiants seuls face au jeu risque plus de les décourager que de les motiver, la bibliothèque Prog&Play devant notamment être introduite par des exemples pour permettre aux étudiants de comprendre la mécanique de jeu ; utiliser les jeux comme de simples plates-formes pour mettre en application des exercices décontextualisés du jeu, dans ce cas les étudiants ne perçoivent pas la composante ludique ce qui a tendance à fortement diminuer la motivation induite par le jeu.

Le second résultat porte sur la souplesse du système Prog&Play et de sa capacité à être intégrable dans différents contextes d'enseignement. Outre le fait que les jeux ont été utilisés avec différents paradigmes de programmation, ils ont pu être intégrés aux formations de différentes manières (en formation initiale, en soutien, intégré dans tout ou partie des travaux pratiques, avec une approche par projet, etc.). Mais quel que soit le mode de mise en œuvre, l'enseignant doit être intégré dans le processus de conception de la situation pédagogique.

## **Conclusion**

Les travaux menés à travers le projet Prog&Play ont été le moteur d'une riche collaboration entre informaticiens et didacticiens. La conception des jeux a fourni

des objets d'étude originaux pour les sciences de l'éducation et les résultats issus de l'analyse des expérimentations – ingénierie didactique – ont permis de nourrir la réingénierie informatique des jeux sérieux et a donc été articulé avec la dimension de la conception.

### **Bibliographie**

Cobb, P., Confrey, J., Disessa, A., Lehrer, R., & Schauble, L. (2003). Design Experiments in Educational Research. *Educational Researcher*, 32(1), 9-13.