



HAL
open science

Analyse de l'activité des élèves dans l'environnement PixelArt pour l'apprentissage de la séquence et de la répétition au cycle 3

Christophe Declercq, Michael Zeyringer

► **To cite this version:**

Christophe Declercq, Michael Zeyringer. Analyse de l'activité des élèves dans l'environnement PixelArt pour l'apprentissage de la séquence et de la répétition au cycle 3. ETIC3, Jun 2018, Paris, France. hal-01826065

HAL Id: hal-01826065

<https://hal.science/hal-01826065>

Submitted on 29 Jun 2018

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.

Analyse de l'activité des élèves dans l'environnement PixelArt pour l'apprentissage de la séquence et de la répétition au cycle 3

Christophe Declercq^a
Michael Zeyringer^b

a. ESPE de l'académie de Nantes, Centre de Recherches en Education de Nantes,

Christophe.Declercq@univ-nantes.fr

b. Conseiller Pédagogique de Circonscription, Segré, Académie de Nantes, Michael.Zeyringer@ac-nantes.fr

Résumé

L'environnement de dessin PixelArt - qui permet aux élèves de programmer un dessin par déplacement du curseur et colorisation de la case courante - est utilisé pour l'apprentissage de la programmation au cycle 3. Il est destiné à travailler en particulier les notions de séquence et de répétition. On présente une méthode d'analyse de l'activité des élèves pour observer le plus finement possible, la manière dont l'élève construit ses solutions - par essais erreurs ou en anticipant - et à quel moment, il prend conscience de l'intérêt d'utiliser la notion de répétition. On en donne les premiers résultats.

Mots clés : apprentissage de la programmation, analyse de l'activité des élèves

Introduction

Nous avons proposé un micro-monde, PixelArt, dédié à l'apprentissage de la programmation au cycle 3. Cet environnement a été construit en extrapolant une situation du concours Castor, *le robot peintre* identifiée comme permettant l'acquisition du concept de boucle par les élèves (Drot-Delange et Tort, 2018). La mise en oeuvre du prototype a été effectuée en utilisant la bibliothèque AccessiBlock (Declercq, 2018), pour rendre l'interface accessible à tous les élèves.

L'environnement PixelArt a été conçu pour amener l'élève dans une posture de programmeur. Nous avons effectué ce choix après avoir analysé (Declercq et Tort, 2018) les différentes postures d'élèves - utilisateur de télécommande, programmeur pas à pas, programmeur de télécommande, programmeur - lors d'activités d'apprentissage de la programmation avec Scratch, telles que recommandées dans les documents d'accompagnement du cycle 3.

Nous nous intéressons maintenant à l'analyse de l'activité des élèves lors d'un apprentissage de la programmation avec PixelArt. Notre problématique est de comprendre comment, à quel moment et à quelles conditions les élèves construisent les notions de séquence et de répétition et acquièrent les compétences associées, en utilisant cet environnement.

Nous décrivons successivement le scénario proposé dans PixelArt, le cadre théorique de notre étude, la méthodologie - qui repose en partie sur un recueil de données automatisé des traces horodatées de l'activité des élèves - puis les premiers résultats et leur discussion.

Un scénario dans PixelArt

Le scénario, présenté en formation continue dans les circonscriptions de Segré et Chalonnnes et mis en oeuvre à titre expérimental par quatre professeurs des écoles dans leurs classes respectives, comporte sept défis qui ont été abordés lors de trois séances consécutives, pendant lesquelles les élèves ont travaillé en binômes.

Le scénario a été construit pour présenter les différentes instructions élémentaires et la séquence (activités 1 à 3), puis pour faire apparaître la nécessité de la boucle, en introduisant des motifs répétitifs.

FIGURE 1 – Scénario pour le cycle 3 - Motifs à reproduire et activité libre

L'application a un jeu d'instructions réduit : les actions élémentaires permettent les déplacements relatifs du curseur et le coloriage de la case courante. Seules la séquence et la répétition bornée permettent de structurer les programmes.

La figure 2 représente une activité terminée où le programme de droite a permis de dessiner en bas à gauche un motif identique au modèle donné en haut à gauche. Au début le curseur est en bas à gauche dans la case (2,2) et est orienté vers le haut.

FIGURE 2 – Activité de programmation avec un motif à reproduire

Cadre théorique et hypothèses

On se réfère à la didactique de l'informatique dans le sens que ce terme a pris dans la communauté francophone, qui s'y intéresse depuis 1988, en tant qu'étude de la programmation, de ses méthodes et de son apprentissage (Bruillard et Baron, 2001).

Les cadres théoriques permettant d'analyser au niveau le plus fin les activités de l'élève programmeur débutant viennent de la psychologie de la programmation (Rogalski, 2015) et de la génèse instrumentale (Rabardel, 1995).

C'est dans ces cadres que nous situons notre travail. L'instrumentation par l'élève d'un artefact dédié à l'apprentissage de la programmation est rendu plus complexe par le caractère différé de l'exécution du programme en cours de construction. C'est ce que notait déjà Rogalski (p. 295) : « une propriété difficile à intégrer [...] est le caractère différé d'une exécution du programme » et que confirme Nogry (2018) dans son étude sur l'apprentissage de la programmation en CP avec la robotique pédagogique (p. 240).

Dans l'utilisation de l'environnement d'apprentissage de la programmation, nous distinguons les actions de programmation - ajout d'une instruction au programme - des actions de commande - exécution du programme ou effacement, pour tenter de rendre compte de comment l'élève construit sa solution à une situation donnée : en utilisant uniquement des instructions suivies d'une unique commande, s'il anticipe parfaitement, ou par une alternance instructions / commandes, s'il procède par une succession d'essais-erreurs. Nous formulons l'hypothèse qu'une analyse des traces d'activités peut nous renseigner sur la posture de l'élève.

Concernant l'apprentissage de la boucle, nous émettons l'hypothèse qu'un élève qui a assimilé cette notion et a acquis la compétence de généralisation, lui permettant d'identifier le motif élémentaire à répéter pour résoudre un problème, utilisera cette notion pour lui permettre d'écrire des programmes plus concis. Nous chercherons alors à mesurer la longueur des programmes produits par les élèves en réponse à des situations types, pour en déduire leur acquisition, ou non, de cette notion.

Nous postulons par ailleurs que divers schèmes de construction de programmes répétitifs pourront être observés, du schème *descendant* où la répétition est saisie dès le début, au schème *ascendant* où c'est d'abord le corps de la boucle qui est construit, avant d'être intégré dans une répétition.

Méthodologie

Pour comprendre au mieux les génèses instrumentales et conceptuelles, nous avons fait le choix de recueillir l'ensemble des données qui pourront nous permettre de répondre aux hypothèses posées. Nous avons privilégié les recueils automatisés de traces d'activité avant de les compléter par des enregistrements vidéo de l'activité des élèves qui, en plus de leur activité instrumentale, permettent aussi de capter leurs interactions langagières, ce qui nous a permis de les retranscrire.

Nous avons appareillé le micro-monde PixelArt pour enregistrer l'horodatage de toutes les interactions de l'élève avec le système : ajout d'instructions, exécutions, passage d'une activité à l'autre, suppression d'instruction. La figure 3 montre un exemple de trace recueillie pendant l'activité d'un des binômes d'une des classes participant à l'expérimentation.

Pour traiter automatiquement les traces et en déduire des informations globales et synthétiques en vue de répondre aux questions que l'on se posait, on a programmé un utilitaire calculant pour chaque activité réalisée sur PixelArt, deux indicateurs et une représentation graphique simplifiée de l'activité.

Le premier indicateur, que l'on a nommé *taux d'anticipation* est le rapport du nombre d'instructions sur le nombre total d'instructions et de commandes. Proche de 1, il indique que l'élève a anticipé la totalité du programme avant de l'exécuter une seule fois. Plus l'indicateur se rapproche de 0, plus il témoigne de nombreuses tentatives d'exécution ou d'hésitations ajout/suppression, ou essais/erreurs.

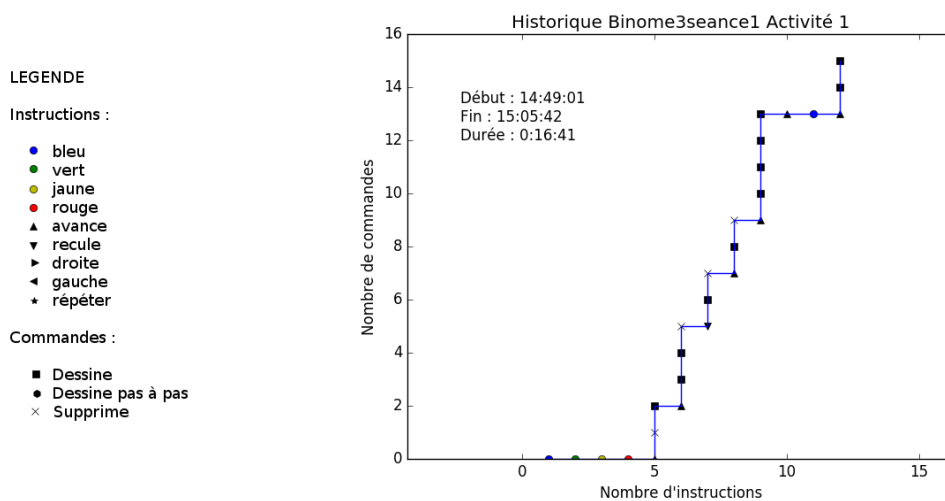
Le second indicateur, que l'on a nommé *taux d'efficacité* est le rapport entre le nombre minimal d'instructions pour résoudre le problème, et le nombre d'instructions utilisées par l'élève. Si ce taux est proche de 1, la solution

FIGURE 3 – Enregistrement horodaté de l'activité

14:49:01 : Début activité 1	14:58:27 : Création d'un bloc dessin_avance
14:54:05 : Création d'un bloc dessin_bleu	14:58:31 : Dessine
14:54:22 : Création d'un bloc dessin_vert	14:58:49 : Destruction d'un bloc
14:54:35 : Création d'un bloc dessin_jaune	14:59:03 : Création d'un bloc dessin_avance
14:54:48 : Création d'un bloc dessin_rouge	14:59:09 : Dessine
14:55:17 : Création d'un bloc dessin_avance	14:59:44 : Dessine
14:55:25 : Destruction d'un bloc	15:00:11 : Dessine
14:56:01 : Dessine	15:00:21 : Dessine
14:57:10 : Création d'un bloc dessin_avance	15:00:47 : Création d'un bloc dessin_avance
14:57:36 : Dessine	15:01:27 : Création d'un bloc dessin_bleu
14:57:40 : Dessine	15:01:33 : Création d'un bloc dessin_avance
14:57:53 : Destruction d'un bloc	15:01:40 : Dessine
14:57:56 : Création d'un bloc dessin_recule	15:05:23 : Dessine
14:58:04 : Dessine	15:05:42 : Début activité 2
14:58:19 : Destruction d'un bloc	

de l'élève, à condition qu'elle soit correcte, est proche de la solution optimale. Un taux faible peut témoigner, en particulier, d'une longue séquence d'instructions identiques, qui aurait pu être remplacée par une répétition.

FIGURE 4 – Représentation graphique de l'activité d'un binôme



La représentation graphique (voir figure 4) montre en abscisse la suite des instructions et en ordonnée la suite des commandes. La forme et la pente de la trajectoire renseignent globalement sur le profil de l'activité et sur les proportions et positions relatives des instructions et des commandes au sein de l'activité. Chaque instruction ou commande est notée par un symbole particulier (voir légende de la figure).

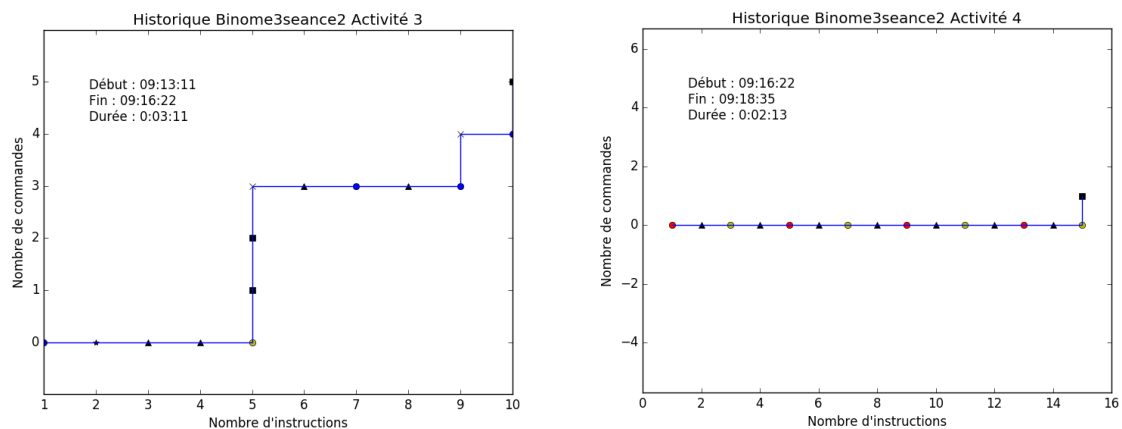
Dans cet exemple d'activité, le binôme a construit un premier programme avec les quatre instructions *bleu*, *vert*, *jaune*, *rouge* en séquence, l'a testé, puis a ajouté successivement des instructions *avance*, avant de tester à nouveau son programme à chaque modification. Au final, ce binôme a réussi l'activité, mais avec une efficacité moyenne qui s'explique par une mauvaise compréhension de l'effet des instructions : les instructions de coloriage n'effectuent aucun déplacement, contrairement à la représentation initiale de ces élèves. Cette analyse a pu être confirmée par la lecture de la retranscription.

Premiers résultats

Pour les classes participant à l'étude, nous avons enregistré ou fait enregistrer les traces d'activités pour tous les binômes et avons filmé, quand cela était possible, un binôme par classe. Certains enregistrements de traces ont été perdus, à cause des modalités de mise en oeuvre : l'enregistrement étant fait dans le navigateur, si un élève quitte l'application avant d'avoir laissé l'enseignant enregistrer sa session, celle-ci n'est pas sauvegardée ; de plus un des navigateurs, pour des raisons de sécurité, ne permet pas cet enregistrement. Nous disposons cependant d'un ensemble de fichiers de traces conséquent ayant permis de générer plus de 50 graphiques d'activité.

L'analyse détaillée des graphiques en lien avec les vidéos correspondantes est encore en cours, mais nous avons déjà pu identifier un certains nombres de faits saillants. Pendant les trois premières activités, le taux d'anticipation, pour chaque binôme est systématiquement orienté à la hausse. Une évolution typique est donnée par la figure 5, où l'activité du binôme 3 est de plus en plus anticipée pour atteindre à l'activité 4 une anticipation parfaite avec une seule exécution finale. L'instrumentation a bien permis aux élèves de développer des schèmes efficaces.

FIGURE 5 – Evolution de la capacité d'anticipation des élèves au fil des activités



L'activité 4 nous semblait, dans une analyse a priori, être celle où pouvait apparaître la boucle. Sur les huit traces disponibles pour cette activité, seules deux montrent une tentative d'usage de la répétition. Les six autres sont similaires au second graphique de la figure 5 et montrent des élèves qui construisent une séquence de quatre fois *rouge, avance, jaune, avance* sans utiliser la construction de boucle.

Le binôme dont l'activité est illustrée ici avait d'ailleurs fait, pour l'activité 3, une première tentative - infructueuse - d'utilisation de la boucle, rapidement remplacée par une recopie de trois fois les mêmes instructions.

C'est lors de l'activité 5 que la nécessité de la répétition apparaît réellement aux élèves. La quasi-totalité des binômes l'utilise sauf un qui écrit la séquence des 48 instructions nécessaires à la reproduction du motif. Ce qui est notable est la variété des usages constatés de la répétition, introduite dès le début ou en fin d'activité, pour répéter cinq fois un pixel vert, ou pour répéter 4 fois la séquence dessinant un côté du carré. Un seul binôme a construit une solution avec deux répétitions imbriquées : l'analyse de la vidéo montre que cela a donné lieu à une discussion vive au sein du binôme qui a effectué plusieurs tentatives avant de réussir à dessiner le motif proposé. Extrait :

G6 Ah oui... Donc. Donc on fait rouge, après avance, vert, avance, vert, avance (repris en chœur par F).

F7 Pourquoi est-ce que tu fais pas un répéter ?

G7 On va répéter pour ici (G montre le carré sur l'écran avec la flèche de la souris). 1,2,3,4,5 (sur le dessin) et là 1,2,3,4,5 (dans le programme). Après on va faire tourne à droite et avance (G prend le bloc répéter et y intègre l'ensemble des blocs préalablement créés). Comme ça on le fait répéter, euh ça on le répète...

F8 Quatre fois.

G8 Quatre fois. On essaie ?

Discussion

Les mesures et graphiques proposés nous ont permis de faire rapidement une première analyse des traces enregistrées. L'hypothèse que ces graphiques contribuent à qualifier la posture de l'élève nous semble pouvoir être confirmée : en effet le choix d'avoir distingué instructions et commandes rend visible l'acquisition par l'élève d'une posture de programmeur capable d'anticiper l'écriture du programme entier avant de faire la première exécution, alors que le *programmeur pas à pas*, commence l'écriture, exécute une première fois, complète le programme et recommence.

Concernant l'apprentissage de la boucle, nous avons pu être surpris au premier abord, de la voir très peu utilisée jusqu'à la quatrième activité incluse. Cela peut nous interroger sur le scénario proposé et nous inciter à proposer plus d'activités aux élèves le temps de consolider l'acquisition de la séquence. L'analyse des vidéos doit être complétée pour identifier le plus précisément possible les moments où la nécessité de la répétition a été questionnée par les élèves. L'analyse détaillée de l'activité 5 nous a montré une variété de schèmes construits par les élèves pour résoudre cette situation difficile. L'hypothèse de l'existence de plusieurs schèmes associés à la construction de la notion de boucle est manifestement confirmée.

Nous n'avons, par contre, pas pu conclure, concernant la pertinence d'une mesure de la longueur du programme, comme indice de l'acquisition du concept de boucle. En effet, dans la version actuelle de notre prototype, nous n'enregistrons pas tous les états successifs du programme au cours de son édition : en particulier les déplacements de blocs ne sont pas enregistrés, ce qui nous empêche de reconstituer a posteriori tous les états successifs des programmes. Nous n'avons donc pu étudier l'efficacité des solutions proposées par les élèves que dans les cas, où ils ont, soit été filmés, soit ont eux-mêmes sauvegardé leur programme à la fin de leur activité.

Les perspectives de ce travail sont, à court terme, de terminer l'analyse des données recueillies : c'est l'objet du mémoire de M. Zeyringer en cours de rédaction. Nous envisageons ensuite, après cette première étude exploratoire, de mener une expérimentation impliquant plus d'élèves, après avoir amélioré à la fois le scénario d'apprentissage et le protocole automatisé de recueil de données.

Remerciements

Aux collègues des écoles des circonscriptions de Segré et Chalonnnes pour leur participation à cette recherche, et aux relecteurs anonymes pour leur remarques constructives et bienveillantes.

Références

- Bruillard, E. et Baron, G.-L. (2001). Une didactique de l'informatique ? *Revue française de pédagogie*, 135(1), 163–172.
- Declercq, C. (2018, février). AccessiBlock : propositions pour un environnement accessible et observable d'apprentissage de la programmation pour l'école élémentaire et le collège. Dans *Didapro 7 – DidaSTIC*. Lausanne, Switzerland. Consulté sur <https://hal.archives-ouvertes.fr/hal-01706657>
- Declercq, C. et Tort, F. (2018, avril). Organiser l'apprentissage de la programmation au cycle 3 avec des activités guidées et/ou créatives. Dans *RJC EIAH 2018*. Besançon, France. Consulté sur <https://hal.archives-ouvertes.fr/hal-01765408>
- Drot-Delange, B. et Tort, F. (2018, février). Concours Castor, ressource pédagogique pour l'enseignement de l'informatique ? Etude exploratoire auprès d'enseignants. Dans *Didapro 7 – DidaSTIC. De 0 à 1 ou l'heure de l'informatique à l'école*. Lausanne, Switzerland.
- Nogry, S. (2018, février). Comment apprennent les élèves au cours d'une séquence de robotique éducative en classe de CP ? Dans *Didapro 7 – DidaSTIC. De 0 à 1 ou l'heure de l'informatique à l'école*. Lausanne, Switzerland. Consulté sur <https://hal.archives-ouvertes.fr/hal-01753106>
- Rabardel, P. (1995). *Les hommes et les technologies ; approche cognitive des instruments contemporains*. Armand Colin. Consulté sur <https://hal.archives-ouvertes.fr/hal-01017462>
- Rogalski, J. (2015). Psychologie de la programmation, didactique de l'informatique. Déjà une histoire... Dans *Informatique en éducation : perspectives curriculaires et didactiques*. Presses universitaires Blaise-Pascal.