



**HAL**  
open science

# Initiation à la programmation informatique à l'école maternelle avec Scratch Junior dès la moyenne section

Nathanaël Crauet

► **To cite this version:**

Nathanaël Crauet. Initiation à la programmation informatique à l'école maternelle avec Scratch Junior dès la moyenne section. Education. 2016. dumas-01367166

**HAL Id: dumas-01367166**

**<https://dumas.ccsd.cnrs.fr/dumas-01367166>**

Submitted on 15 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.

**Année universitaire 2015-2016**

***Master Métiers de l'enseignement, de l'éducation et de la formation***

***Mention Premier degré***

**Initiation à la programmation  
informatique à l'école  
maternelle avec Scratch Junior  
dès la moyenne section.**

**Présenté par Nathanaël Crauet**

Première partie rédigée en collaboration avec Marie Baatsch

**Mémoire de M2 encadré par Marie Busser**

## Attestation de non-plagiat

Je soussigné(e) (Prénom NOM)

Nathanael CRAUET

Auteur du mémoire de master 2 ou de l'écrit scientifique réflexif MEEF-PE / MEEF-SD / MEEF-EE (entourez la mention et indiquez le titre du mémoire)

Initiation à la programmation informatique à l'école maternelle avec  
Scratch Junior dès la moyenne section

déclare sur l'honneur que ce mémoire est le fruit d'un travail personnel, que je n'ai ni contrefait, ni falsifié, ni copié tout ou partie de l'œuvre d'autrui afin de la faire passer pour mienne.

Toutes les sources d'information utilisées et les citations d'auteur ont été mentionnées conformément aux usages en vigueur.

Je suis conscient(e) que le fait de ne pas citer une source ou de ne pas la citer clairement et complètement est constitutif de plagiat, que le plagiat est considéré comme une faute grave au sein de l'Université, pouvant être sévèrement sanctionnée par la loi.

Fait à Bonneville  
le 14/05/2016

Signature de l'étudiant(e)



## Autorisation de diffusion électronique d'un mémoire de Master 2 MEEF dans la base DUMAS<sup>1</sup>

### Autorisation de l'étudiant(e)

Je soussigné(e) Nathanaël CRAVET  
auteur et signataire du mémoire de niveau Master 2, intitulé :  
Initiation à la programmation informatique à l'école maternelle avec Scratch  
Tutoriel des la moyenne section

, agissant en l'absence de toute contrainte,

**autorise**

**n'autorise pas**<sup>2</sup>

le Service Interétablissement de Documentation de l'Université Grenoble Alpes-Grenoble INP à le diffuser, sans limitation de temps, sur la base DUMAS en texte intégral.

- Je certifie la conformité de la version électronique déposée avec l'exemplaire imprimé remis au jury.
- Je m'engage à signaler les documents pour lesquels je ne détiens pas les droits de reproduction et de représentation ou les autorisations afférentes. Ces documents devront être masqués ou retirés de la version diffusée sur la base DUMAS par les auteurs.
- La présente autorisation de diffusion n'a pas de caractère exclusif. L'auteur conserve par conséquent toutes les possibilités de cession de ses droits et de diffusion concomitante de son mémoire.
- Je renonce à toute rémunération pour la diffusion effectuée dans les conditions précisées ci-dessus.
- Conformément à la loi « informatiques et libertés » du 6 janvier 1978, modifiée en 2004, je pourrai à tout moment modifier cette autorisation de diffusion par simple lettre ou courriel à la BUPE : [membupe@univ-grenoble-alpes.fr](mailto:membupe@univ-grenoble-alpes.fr)

Fait à Bonnevillotte le 14/05/2016

Signature de l'étudiant(e),  
Précédée de la mention « bon pour accord »

bon pour accord



<sup>1</sup> La base DUMAS (Dépôt Universitaire des Mémoires Après Soutenance) est en accès libre à l'adresse : <http://dumas.ccsd.cnrs.fr/>  
NB : le mémoire sera diffusé sur Dumas si la note obtenue pour l'écrit est supérieure ou égale à 16/20 et si l'auteur et le responsable de mémoire en donnent l'autorisation

<sup>2</sup> Entourer la mention choisie



Introduction.....	1
-------------------	---

## Première partie: la programmation informatique

1	La place de l'informatique dans notre société contemporaine .....	2
1.1	La révolution numérique est en marche .....	2
1.1.1	Grandir à l'ère du numérique .....	2
1.1.2	Le citoyen : usager et concepteur informatique de demain .....	3
1.2	Pensée informatique et algorithmes : définitions.....	4
1.2.1	Qu'est-ce que la pensée informatique? .....	4
1.2.2	À quoi servent les algorithmes ?.....	5
1.3	Comment est enseignée la pensée informatique ?.....	6
1.4	Les activités « débranchées ».....	7
2	La programmation informatique à l'école primaire.....	8
2.1	Les pays précurseurs .....	8
2.2	Evolution de la place de l'informatique dans les programmes .....	9
2.3	Pourquoi éduquer si tôt les enfants à la programmation ? .....	11
2.4	Apprendre la programmation par le jeu et par la pratique .....	11
3	Présentation du logiciel de programmation Scratch .....	13
3.1	Présentation générale .....	13
3.2	Scratch sur ordinateur.....	13
3.3	Scratch junior sur tablette.....	14

## Deuxième partie: Scratch Junior en moyenne section

1	Formulation de la problématique .....	16
2	Méthode .....	16
2.1	Participants.....	16
2.2	Matériel .....	17
2.2.1	Les Choix didactiques .....	17
2.2.2	Choix pédagogiques .....	19
3	Procédure .....	21
3.1	Déroulement de l'étude .....	21
3.2	Les outils d'analyses .....	20
4	Résultats.....	24
4.1	Des apprentissages en jeu.....	24
4.1.1	Blocs par blocs .....	24

4.1.2	Planification d'un message.....	25
4.2	Comparaison des supports.....	26
4.3	Participation .....	26
4.4	Plus d'exercices sur tablette.....	27
4.5	La diversité des blocs.....	28
4.6	Conception de l'outil numérique .....	29
5	Discussion .....	30
5.1	Confrontation aux hypothèses.....	30
5.1.1	Apprentissage réalisé .....	30
5.1.2	Nécessité de la phase débranchée .....	30
5.1.3	Utilité d'un scénario et d'un environnement personnalisable. ....	30
5.1.4	Utilisation du logiciel .....	31
5.1.5	Présence de l'enseignant.....	31
5.1.6	Posture face à l'outil numérique .....	31
5.2	Limites et perspectives.....	32
5.3	Autres problèmes rencontrés.....	33
6	Conclusion .....	34
	Bibliographie.....	35

## Introduction

Depuis l'apparition et l'utilisation massive des outils informatiques à partir des années 90, les enjeux des apprentissages ont évolué. Alors que les apprentissages étaient majoritairement centrés sur le savoir « lire, écrire, compter » dans les années 50, l'école de la République doit désormais favoriser la création artistique et inclure les nouvelles technologies comme outils d'aide aux apprentissages. Institutionnalisé dans les nouveaux programmes de 2016 l'apprentissage des mathématiques doit être élargi à l'initiation à la programmation informatique. Dans la perspective de ce nouveau cadre institutionnel, nous avons étudié la place de l'informatique dans notre société et dans système scolaire actuel et analysé comment aborder l'enseignement de la programmation avec des élèves de l'école élémentaire. Il est question dans ce mémoire d'essayer d'initier les élèves de moyenne section au principe algorithmique avec l'utilisation d'un nouvel outil pédagogique, Scratch Junior, une application sur tablette. Cette tentative s'articule autour de différentes choix pédagogiques dont les effets sont observés. Dans la première partie, nous définissons la programmation en tant que "pensée informatique" et nous présentons la place qu'occupe la programmation à la fois dans les sociétés et dans les programmes scolaires. Nous présentons ensuite les différentes approches possibles pour son enseignement et les outils pédagogiques récemment élaborés, en particulier les logiciels Scratch.

Dans la seconde partie, je présente le dispositif imaginé mis en place au sein des activités pédagogiques complémentaires avec les élèves de ma classe. La séquence d'apprentissage est constituée autour d'un scénario de déplacement d'un personnage qui permet d'introduire le principe algorithmique. Cette séquence alterne activité débranchée et utilisation de la tablette .Les élèves sont guidés par un enseignement explicite et un encadrement privilégié.

# Première partie : La programmation informatique

## 1 La place de l'informatique dans notre société contemporaine

### 1.1 La révolution numérique est en marche

#### 1.1.1 Grandir à l'ère du numérique

Wikipédia (2016) définit la société de l'information ou société de la connaissance comme « un état de la société dans lequel les technologies de l'information jouent un rôle fondamental. Elle est en général placée dans la continuité de la société industrielle. » En effet, notre histoire culturelle et celles de nos enseignements est marquée par trois grandes révolutions : à la Renaissance, la diffusion rapide de l'imprimerie, au XIXe siècle, le développement de la presse écrite puis au XXIe siècle, le développement du Web, de l'informatique et des télécoms, appelé aussi révolution numérique. Les TIC, technologies de l'informatique et de la communication, vont peu à peu régir toutes les activités socio-économiques. Nicolas Curien et Pierre-Alain Muet (Curien, N. & Muet, P-A., 2004) ont cherché à analyser l'impact de cette révolution numérique et attestent que cette dernière va entraîner de nouvelles représentations mentales et sociales, un changement de conception du monde et des mutations radicales auxquelles il faut se préparer.

Déjà, on constate que l'arrivée massive d'Internet a complètement modifié notre rapport à l'apprentissage et nos rapports sociaux. Beaucoup de personnes de la génération précédente restent sceptiques face à un apprentissage de l'informatique dès l'école primaire. Mais tandis que la génération passée a vu peu à peu le numérique s'immiscer dans la vie quotidienne, la nouvelle génération est née avec, et être un « natif numérique » ne signifie pas qu'on est « alphabétisé numérique » (Pasquinelli, E., Zimmerman, G., Bernard-Delorme, A., & Descamps-Latscha, B., 2013). Comme le rappelle Emmanuel Davidenkoff, ne pas s'intégrer à l'innovation c'est mourir. Ainsi, de la même façon que les géants de la marine marchande à voile, qui n'ont pas su se convertir à la vapeur, ont disparu en une dizaine d'années, les secteurs qui ne sauront pas intégrer le numérique échoueront ou disparaîtront (Davidenkoff, E. 2014). C'est là tout l'intérêt de former les plus jeunes pour qu'ils grandissent non seulement avec un état d'esprit ouvert et éclairé sur ce nouveau mode de travail et de communication mais qu'ils sachent également comment utiliser activement ces outils.

Le Ministère de l'éducation nationale (MEN) confirme régulièrement la volonté d'équiper les écoles d'outils numériques et les nouveaux programmes de 2016 vont en ce sens. Pour



Davidenkoff, l'innovation était jusqu'à présent portée principalement par les parents qui investissaient dans des tablettes pour apprendre à lire, à écrire ou à compter mais « c'est par le primaire que la révolution numérique pourrait entreprendre la conquête de l'enseignement scolaire » (Davidenkoff, E. 2014). C'est ainsi que le MEN a l'ambition de déployer 1 milliard d'euros sur 3 ans à partir de la rentrée 2016 afin d'être le leader de l'e-éducation. Outre l'achat des ordinateurs et des tablettes, ce budget devra servir à former les enseignants et à développer les ressources pédagogiques. Ainsi, 100% des établissements devront être connectés à l'horizon 2018, et d'ici 2020, 70% des élèves de primaires et de collèges équipés en terminaux individuels et collectifs (MEN, 2016).

La révolution numérique est en marche, y compris dans le milieu scolaire !

### **1.1.2 Le citoyen : usager et concepteur informatique de demain**

Le citoyen du XXI<sup>e</sup> siècle doit être en mesure d'utiliser les outils numériques mis à sa disposition non seulement en tant qu'utilisateur mais aussi en tant que concepteur.

Les spécificités des nouvelles technologies de l'information et de la communication sont leur rapidité de transmission et le double flux qu'elles permettent : le récepteur devient émetteur et vice versa. Si avec les anciens médias, tels que les livres, les affiches, la télévision, les journaux, l'information circulait unilatéralement dans le sens de l'émetteur vers le récepteur, les médias numériques ont cette particularité d'offrir une réponse du récepteur vers l'émetteur. En d'autres termes, les médias numériques sont interactifs. Grâce aux médias numériques, l'utilisateur prend alors une nouvelle envergure. Comme le pense Patrice Flichy (Flichy, P. 2010), grâce à l'accès aux nouvelles technologies, en particulier Internet, celui qui était un usager plus ou moins passif, subissant les usages imposés par les grands groupes industriels, peut devenir expert et créateur de ses usages. Ainsi, selon ses termes, « l'amateur » peut s'appuyer sur ces compétences acquises par l'expérience pour rivaliser avec les experts, « c'est dans son quotidien que l'amateur puise une source considérable de connaissances et acquiert de nouveaux savoirs et savoir-faire ». Désormais un renversement des pouvoirs est possible, puisque désormais les usages ne peuvent plus uniquement être imposés par des industries.

L'émergence de sociétés façonnées par les usagers confirme cette évolution. Les nouveaux géants économiques de ce monde étaient initialement des usagers qui ont diffusé un de leur usage réel. Des modèles participatifs et collaboratifs, comme Wikipédia, semblent être également la conséquence des nouveaux usages créés par l'utilisateur. Le citoyen actuel doit donc être formé pour faire face à ces nouvelles caractéristiques du monde. D'une part c'est de

plus en plus à lui d'inventer son quotidien, son travail, et d'autre part cette nouvelle invention passera par la voie numérique. Cependant, une différence s'installe entre les citoyens capables de créer leur propre activité numérique grâce à la maîtrise des langages de programmation et ceux qui participent en ne faisant que partager leur travail sur des supports numériques déjà existants. Ces derniers dépendent d'un savoir qu'ils ne maîtrisent pas. Donner les moyens de comprendre ce langage informatique aux élèves est l'un des nouveaux et principaux enjeux de l'école pour préparer l'élève autant à être citoyen qu'à trouver sa place dans le monde professionnel où algorithme et pensée informatique deviennent des mots du vocabulaire courant.

## **1.2 Pensée informatique et algorithmes : définitions**

### **1.2.1 Qu'est-ce que la pensée informatique?**

Enseigner l'informatique revient souvent pour les chercheurs à enseigner « la pensée informatique ». Nous nous efforcerons ici de clarifier ce que signifie ce terme.

Tout d'abord, il existe plusieurs définitions du terme « pensée informatique ». Généralement, ce qui est mis en avant dans la pensée informatique c'est le concept d'algorithme. « Le concept d'algorithme est au cœur de la pensée informatique. [...] L'algorithme est un enchaînement mécanique d'actions, dans un certain ordre, qui chacune a un effet, et dont l'exécution complète permet de résoudre un problème ou de faire quelque chose » (Tchounikine, P. 2016). Les algorithmes sont la base de tous les programmes informatiques mais ils ne sont pas exclusifs à l'informatique. Nous utilisons des algorithmes pour effectuer des calculs, des recettes de cuisines... Ce qu'on appelle la pensée informatique est en fait plus largement une méthode de résolution de problèmes qui s'applique dans tous les domaines de la vie dès lors que l'on rencontre un problème ou que l'on souhaite mener un projet. La pensée informatique est « un ensemble d'attitudes et de connaissances universellement applicables [...] concrètement, face à un problème donné, se posent les questions suivantes : quel est son degré de difficulté ? Et quel est le meilleur moyen de le résoudre ? L'informatique, parce qu'elle s'appuie sur des fondations théoriques solides, nous offre des outils pour répondre à ces questions. » La pensée informatique est donc « une capacité de réflexion à différents niveaux d'abstraction » qui va bien au-delà de la capacité de programmation (Wing, J. 2006).

Selon Jérémy Scott, la pensée informatique peut être scindée en cinq grandes catégories :

- appréhender un problème et sa solution à différents niveaux (abstraction)

- réfléchir aux tâches à accomplir sous forme d'une série d'étapes (algorithmes)
- comprendre que pour résoudre un problème complexe il faut le décomposer en plusieurs problèmes simples (décomposition)
- comprendre qu'il est probable qu'un nouveau problème soit lié à d'autres problèmes déjà résolus par l'élève (reconnaissance de formes), et
- réaliser que la solution à un problème peut servir à résoudre tout un éventail de problèmes semblables (généralisation) ». (Scott, J. 2012)

Cette pensée informatique fait particulièrement écho à la résolution de problèmes en mathématiques. Domaine qui fait l'objet de nombreuses études. Dans un article paru dans Grand N, Jean Julo (Julo, J. 2002), maître de conférence en sciences de l'éducation, fait la synthèse de ces différentes approches en expliquant que « l'une des variables qui caractérise le plus un expert dans un domaine donné est sa manière de catégoriser un ensemble de problèmes qu'on lui propose ». D'autres approches citées dans cet article se concentrent sur la méthodologie et donc la compétence à décomposer un problème complexe en plusieurs problèmes simples.

Il n'en demeure pas moins qu'au delà de cette approche universelle de résolution de problèmes, la pensée informatique a ses spécificités. En effet, pour résoudre un problème, la pensée informatique utilise un langage artificiel où les algorithmes sont prédominants (condition, boucle, variables) et c'est en ce sens qu'un apprentissage spécifique de ce langage doit être envisagé.

### 1.2.2 À quoi servent les algorithmes ?

L'informatique regroupe beaucoup de concepts souvent confus. Notre usage personnel et quotidien participe à entretenir ses confusions. Lorsque nous utilisons des logiciels ou applications sur des ordinateurs, tablettes et téléphones nous sommes en position d'usagers de supports techniques. Le véritable enjeu de l'enseignement de l'informatique est de dépasser ce cadre de simple usager pour faire prendre conscience du véritable fonctionnement de l'informatique. L'écueil à éviter est de penser que le fait d'utiliser des ressources et supports informatiques pour enseigner revient à enseigner l'informatique. La simple utilisation d'un TBI n'a aucun intérêt du point de vue de l'apprentissage de l'informatique. Les nouveaux programmes parlent de former les élèves à la programmation informatique, mais de quoi s'agit-il exactement ? Pour le dictionnaire Larousse, programmer c'est « écrire les programmes informatiques correspondant à l'algorithme de résolution d'un problème. »

Les algorithmes font partie du vocabulaire opaque de l'informatique, il convient d'en donner ici une définition plus claire. Olivier Ertzscheid, enseignant chercheur en sciences de l'information et de la communication à l'université de Nantes, en donne la définition suivante : « Il s'agit de formules mathématiques qui permettent de faire appliquer des règles, de trier des informations pour en extraire un résultat. » Pour nombre de chercheurs, un algorithme c'est comme une recette de cuisine, c'est à dire une suite d'instructions menant à un résultat, mais avec la notion de condition ou de répétition.

Les algorithmes qui étaient jusqu'à présent uniquement utilisés par la recherche scientifique ont désormais envahi notre vie quotidienne. En effet, les algorithmes sont utilisés autant par les géants de l'informatique que par les pouvoirs publics pour classer des informations, ajuster les prix, identifier et analyser les choix et comportements des consommateurs. En d'autres termes, les algorithmes scrutent et analysent le moindre clic de souris de l'utilisateur pour adapter l'offre à la demande. De nos jours, tous les secteurs utilisent des algorithmes, de l'optimisation des réseaux d'énergie, à la construction des voitures sans chauffeurs, de la surveillance faite par la police aux inscriptions à l'université (Devillard, A. 2015).

Pensée informatique, programmation et algorithmes font désormais partie de notre quotidien que nous en soyons conscients ou non.

### **1.3 Comment est enseignée la pensée informatique ?**

Il existe deux principales façons d'appréhender l'enseignement de la pensée informatique. D'une part, cet enseignement peut se faire selon une méthode « traditionnelle » qui se concentre sur l'enseignement des algorithmes au sein d'une discipline appelée sciences de l'informatique. On apprend à utiliser les algorithmes sur des supports informatiques pour pouvoir les réutiliser sur ces mêmes supports. Dans cette approche, l'enseignant cible une notion algorithmique (notion de condition, de boucle,...) que l'élève explorera de façon plus ou moins guidée selon que l'enseignant abordera l'enseignement de manière socio-constructiviste ou explicite.

Il existe d'autre part l'approche créative. L'idée principale est de mettre un outil informatique au service de la créativité de l'élève. Ce dernier va dans un premier temps se familiariser avec l'outil avant de chercher naturellement à saisir le fonctionnement des algorithmes. La mise en place de projet personnel est ici particulièrement adaptée. Ici, l'enseignant ne met pas en avant une notion mais répond au fur et à mesure aux interrogations de chaque élève lorsqu'il rencontre une difficulté dans l'élaboration de leur projet. La pédagogie par projet, souvent

plébiscitée en didactique de l'enseignement, est ici complètement mise en oeuvre.

Ces deux approches ne sont cependant pas exclusives l'une de l'autre. Il est tout à fait possible d'envisager que des projets créatifs soient dirigés à certains moments vers des apprentissages plus ciblés et vice versa. Toutefois, ces deux types d'apprentissages se distinguent par le fait que la première méthode délivre des connaissances tandis que la deuxième tend à développer la créativité.

#### 1.4 Les activités « débranchées »

A ce stade, il semble important de mentionner que l'apprentissage de l'informatique ne se limite pas uniquement à l'utilisation de supports informatiques. De nombreuses activités permettent d'aborder la notion d'algorithmes sans avoir recours au support informatique : elles sont appelées les activités débranchées. Leur intérêt est multiple et il est parfois indispensable d'y recourir. En effet, aborder l'algorithme de déplacement avec un enfant de 5 ans nécessite inéluctablement une première approche sur un support manipulable. L'activité débranchée peut donc s'avérer être une nécessité pour construire un apprentissage en phase avec le développement cognitif de l'enfant. Par exemple, le programme néozélandais *Computer sciences unplugged* (informatique sans ordinateur) a conçu tout un système pour apprendre les fondements de l'informatique (nombres binaires, algorithmes, compression de données, etc.) à travers l'utilisation de jeux de cartes, de couleurs et de mouvements qui rendent cet enseignement ludique et ce sans avoir recours à un ordinateur (2016). La start-up américaine Primo Toys a également inventé le Cubetto, un robot jouet pour introduire au fondamentaux de la programmation sans ordinateur, à la planification d'une séquence de déplacement (Le Monde, mai 2016).



Dans d'autres situations d'enseignement, l'activité débranchée peut être utile sans être indispensable. Pour Tchounikine (Tchounikine, P. 2016) c'est un moyen d'étude d'un algorithme particulier avant un réinvestissement sur machine. « Les activités pédagogiques

sans ordinateurs sont alors des moyens pour enseigner des notions et principes qui seront ensuite réinvestis ; les exercices proposés aux élèves sont typiquement conçus pour les amener à mobiliser telle ou telle compétence algorithmique cible ; et les langages de programmation sont des moyens d'écrire des algorithmes sous une forme exécutable par un ordinateur ». Encore, cette activité débranchée peut être envisagée comme un moyen de prise de conscience du degré d'intelligence d'un ordinateur. Gérard Berry, professeur au Collège de France, où il occupe la chaire « Algorithmes, machines et langages » explique dans une des émissions « Rue des Ecoles » sur France-Culture : « l'ordinateur fait exactement ce qu'on lui dit mais [...] il n'a aucune intelligence [...], programmer c'est expliquer quelque chose non pas à un animal intelligent mais à une machine extrêmement rapide mais tout à fait stupide. »

Les activités débranchées participent donc à la structuration du développement cognitif de l'enfant, à une première approche de concept postérieurement réinvesti sur tablette et à la prise de conscience de la véritable nature de la machine.

## **2 La programmation informatique à l'école primaire**

### **2.1 Les pays précurseurs**

Selon Federica Minichiello (Minichiello, F., 2014), le Royaume-Uni s'affiche comme le premier pays du G20 à avoir introduit la programmation informatique à l'école primaire. En effet, en juillet 2014, un rapport de la commission européenne présente les compétences en programmation comme « une solution au chômage des jeunes et au déficit de qualifications dans les TIC », estimé à 900 000 professionnels d'ici 2020, et invitent les ministres à intégrer cet apprentissage dès le primaire (Vassiliou, A. & Kroes, N., 2014). C'est ainsi que le « code » est devenu une matière à part entière dans les écoles publiques anglaises dès septembre 2014.

L'Estonie a été précurseur en l'intégrant dans ces programmes dès 2012 puis d'autres pays industrialisés suivent comme Israël, la Finlande, la Corée du Sud et plus récemment la France ou l'Italie.

Les Etats-Unis ont rapidement rejoint le banc des pays précurseurs en lançant l'initiative « L'heure du code ». Cette dernière a été créée par l'association à but non lucratif *code.org* qui souhaite démocratiser l'accès à l'informatique et étendre l'apprentissage du code aux femmes mais aussi aux groupes ethniques sous-représentés. Le projet est en partenariat avec les



géants de l'informatique comme Microsoft, Apple ou encore Amazon, et est soutenu par Barack Obama lui-même. Il a pour but d'initier les gens à la programmation notamment à l'aide du logiciel Scratch. En une heure, les participants peuvent avoir un aperçu de ce que veut dire « coder ». Ils ne sauront pas programmer en une heure, mais ils verront que c'est une activité amusante et créative qui peut être menée par tous, quelles que soient son origine ou ses compétences préalables. Cette initiative a rapidement pris une ampleur internationale et en avril 2016 plus de 230 millions d'élèves ont déjà participé dans plus de 180 pays.

Tandis que ce projet s'étend toujours plus, montrant l'engouement pour cette discipline, le Québec reste en retard comme l'explique Isabelle Porter. Lors du dernier événement mondial « L'heure du code » le Québec a été l'une des provinces du Canada où le moins de gens a participé. On constate dans cette région que de plus en plus de voix s'élèvent pour que les rudiments du codage et de la programmation informatique soient enseignés aux élèves. Certains enseignants tentent l'expérience dans leurs classes et confirment que ces projets sont bien accueillis par les élèves et les parents mais les institutions n'ont pas encore franchi le pas en mettant le code dans les programmes de façon officielle (Porter, I., 2015).

Envisager d'intégrer la programmation informatique dès le primaire n'est pas sans poser de problèmes, notamment dans la formation des enseignants. C'est ainsi qu'au Royaume-Uni, de nombreux partenariats ont été créés entre les écoles et les grands groupes privés comme Google et Microsoft. Comme le rappelle Minichiello, la société Microsoft a lancé par exemple le « First class computing programm », pour former 160 000 enseignants pour la rentrée 2014 et a mis en place la plate-forme d'échange d'expériences entre écoles « Switched on computing » (Minichiello, F., 2014).

Enseigner la programmation informatique soulève ainsi la nécessité d'allier la volonté du gouvernement, les compétences des personnels enseignants, et les entreprises à la pointe de la technologie informatique.

## **2.2 Evolution de la place de l'informatique dans les programmes**

L'école a donc un rôle important à jouer pour former les futurs citoyens de demain à la culture scientifique et technologique et à l'éducation aux médias.

L'informatique s'inscrit dans les programmes officiels dès 1995 mais il faudra attendre plus de vingt ans pour que la programmation informatique y trouve également sa place.

En 2000, le gouvernement (MEN) crée le Brevet Informatique et Internet (B2I) rendant obligatoire l'apprentissage des nouveaux outils informatiques. L'élève doit être en mesure de

s'approprier un environnement informatique de travail dans le respect des règles et des autres. Entre autre, il doit savoir créer, produire, traiter et exploiter des données. Toutefois, à ce moment-là, apprendre l'informatique c'est principalement apprendre à manipuler le clavier, la souris, l'ordinateur et à savoir utiliser les logiciels de base de traitement de texte ou de bases de données.

En 2002, les documents étudiés par le biais d'Internet sont désormais d'une égale qualité aux documents papiers.

En 2008, la culture numérique doit permettre d'utiliser l'informatique, le multimédia et l'Internet « de façon réfléchie et efficace ». En fin de CE1, l'élève est capable de commencer à s'approprier un environnement numérique. A partir du CE2, le recours aux TICE devient habituel dans le cadre du B2I.

Mais qu'en est-il de la programmation et du code ? La loi de refondation de l'école du 8 juillet 2013 réaffirme cette volonté du gouvernement de faire entrer l'école dans l'ère du numérique comme un « levier de modernisation, d'innovation pédagogique et de démocratisation du système scolaire », mais ce n'est qu'à la rentrée 2016 que la programmation intégrera officiellement les programmes de l'Education Nationale.

Les TICE s'inscrivent désormais au domaine numéro 2 intitulé « Les méthodes et outils pour apprendre » (MEN BO spécial du 26/11/2015). L'informatique doit permettre d'accompagner les élèves dans tous les apprentissages et dans toutes les disciplines. La programmation devient également une compétence à part entière. Les attentes sont les suivantes :

- Au cycle 2 : « Dès le CE 1, les élèves peuvent coder des déplacements à l'aide d'un logiciel de programmation adapté, ce qui les amènera au CE 2 à la compréhension, et la production d'algorithmes simples. »
- Au cycle 3 : « En mathématiques, ils apprennent à utiliser des logiciels de calculs et d'initiation à la programmation. Des activités géométriques peuvent être l'occasion d'amener les élèves à utiliser différents supports de travail : papier et crayon, mais aussi logiciels d'initiation à la programmation. Une initiation à la programmation est faite à l'occasion notamment d'activités de repérage ou de déplacement (programmer les déplacements d'un robot ou ceux d'un personnage sur un écran), ou d'activités géométriques.
- Cycle 4 : « Les élèves conçoivent tout ou partie d'un programme, le compilent et l'exécutent pour répondre au besoin du système et des fonctions à réaliser. »

### 2.3 Pourquoi éduquer si tôt les enfants à la programmation ?

Éduquer les élèves à utiliser l'informatique (manipulation de la souris, du clavier, des logiciels) n'est pas un but en soi. L'objectif est de faire comprendre à ces jeunes apprenants que c'est à eux de maîtriser l'outil et non de se faire maîtriser par les machines ou ceux qui les maîtrisent et qui ont donc par ce biais un certain pouvoir sur les masses.

Comme l'explique John Naughton (Naughton, J. 2012), les cours d'informatique actuels poussent les enfants à appréhender passivement les machines. Selon l'auteur, « Au lieu d'éduquer les enfants au sujet d'une des technologies les plus révolutionnaires de leur jeune vie, nous nous sommes bornés à les entraîner à utiliser des programmes obsolètes. [...] On a pensé qu'il fallait apprendre aux enfants à utiliser un ordinateur comme on apprenait à conduire une voiture, c'est à dire sans avoir besoin de comprendre comment fonctionne la machine pour l'utiliser. Toutefois, on n'oublie que les voitures ne dictent pas notre vie quotidienne tandis que l'informatique fait fonctionner le monde entier, contrôle nos communications, alimente nos téléphones portables, dirige nos comptes bancaires, conserve nos agendas, arbitre et espionne nos relations sociales, et compte même nos votes. C'est pourquoi les enfants devraient être initiés au plus tôt au fonctionnement même de l'appareil et être en mesure de comprendre comment fonctionne les algorithmes, le cryptage, la résolution de problèmes informatiques. Ne pas former les jeunes maintenant, c'est en faire de futurs estropiés. Ils grandiront comme des consommateurs passifs de [...] technologies créés par une élite travaillant pour les plus grosses entreprises comme Google ou Facebook. Nous élèverons une génération de hamsters dans des cages brillantes construites par des Mark Zuckerberg et autres. »

C'est pourquoi, il semble important de former les plus jeunes à comprendre comment fonctionne la machine pour leur permettre de devenir concepteur et non simplement usager.

### 2.4 Apprendre la programmation par le jeu et par la pratique

Apprendre quelque chose se fait de différentes manières mais force est de constater que l'apprentissage par l'expérience et par la manipulation ont un impact plus grand sur l'apprenant qu'une simple mémorisation théorique. Comme dit le proverbe c'est en forgeant qu'on devient forgeron ! Déjà cinq siècles avant Jésus Christ, le philosophe et éducateur chinois Confucius disait la chose suivante : « J'entends et j'oublie, je vois et je me souviens, je fais et je comprends. » Plusieurs siècles plus tard, de nombreux auteurs comme Rousseau, Kant, Dewey, Piaget, pour ne citer qu'eux, insistent sur la nécessité absolue d'apprendre par

l'expérience pour ancrer solidement les connaissances.

**Dewey** a été le premier à étudier le rôle important de l'apprentissage par l'expérience, son leitmotiv était « apprendre en faisant » (learning by doing) et il considérait que l'école propose des apprentissages « trop souvent déconnectés de l'expérience ». Pour **Piaget**, on ne naît pas intelligent mais on le devient à force d'interaction entre la personne qui apprend et son environnement.

Au vu de ces différentes théories, on comprend la nécessité de faire expérimenter les élèves pour leur faire apprendre quelque chose.

Toutefois, il n'est pas toujours aisé de trouver des activités qui permettent à l'élève d'agir en toute quiétude tout en restant amusantes et en mesure de capter longtemps son attention. Les jeux ont cette faculté. Le jeu permet d'apprendre différemment mais n'est pas un frein ou un obstacle aux apprentissages. Dans les années 90, l'engouement pour les jeux vidéo a été très fort. Mais la peur de « l'addiction » à des jeux où l'intellect de l'enfant est peu sollicité a également été forte. Peu à peu, des jeux « sérieux » ont vu le jour permettant à la fois de combiner apprentissage et jeu. Une nouvelle image du jeu vidéo est née. Ces « jeux sérieux » présentent de nombreux avantages. Tout d'abord, « la motivation des élèves est importante car c'est un environnement qu'ils aiment » (Bihouée, P. & Coliaux, A., 2011). Comme le rappelle la Direction du numérique pour l'éducation du Ministère de l'Education Nationale, « de nombreuses expérimentations ont montré que l'usage bien pensé d'outils et de ressources pédagogiques numériques rend les enseignements plus stimulants, plus attractifs et augmente la motivation, l'activité et l'engagement des élèves dans leurs apprentissages. » (Direction numérique du MEN, 2015)

On peut alors activer ou réactiver des connaissances mais également cibler sur de nouveaux apprentissages. Les élèves doivent alors faire des choix stratégiques et ainsi développer leur autonomie. Le logiciel Scratch est l'un de ces « jeux sérieux » où la concentration, la prise d'initiative et les capacités créatives sont sollicitées. Le jeu modifie le rapport aux apprentissages et permet à des élèves peu scolaires de trouver une autre source de motivation. Nous avons donc souhaité tester en classe l'utilisation d'un jeu sérieux, le logiciel Scratch, et observer quelle était la meilleure manière d'aborder cet enseignement.

## 3 Présentation du logiciel de programmation Scratch

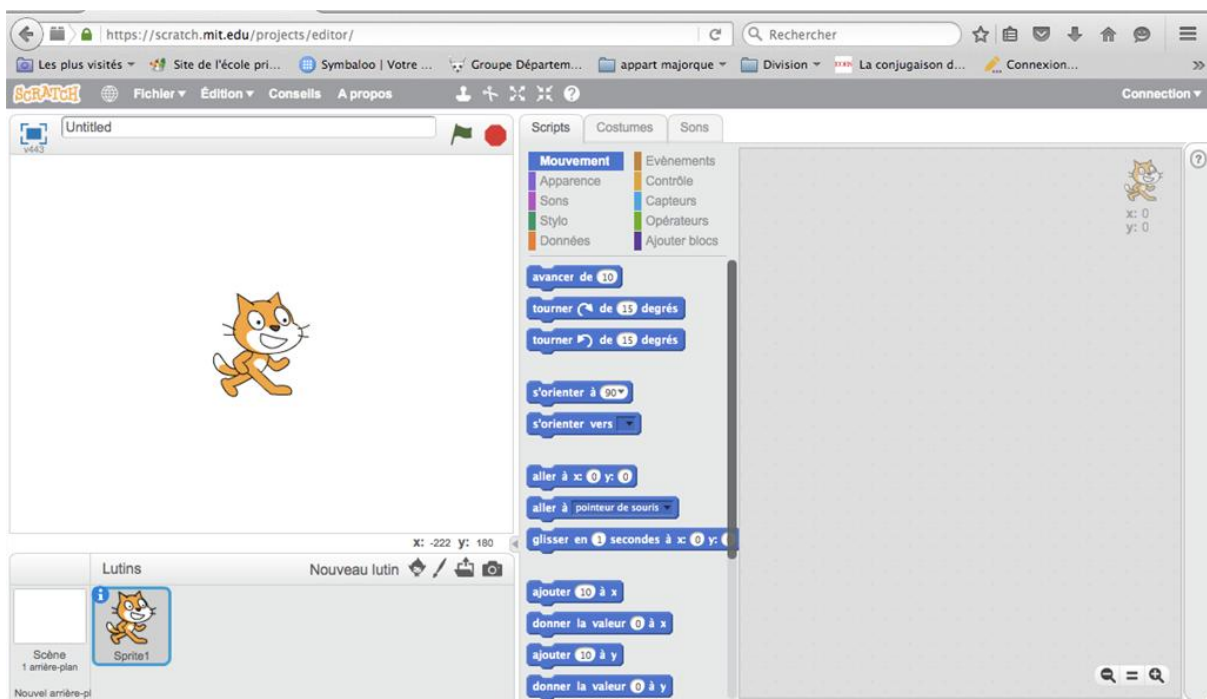
### 3.1 Présentation générale

C'est dans ce nouveau cadre que nous nous intéressons de plus près à la programmation informatique dès l'école primaire. Selon Claude Terosier créatrice de la société « Magic Makers » qui propose des ateliers de programmation aux enfants, « programmer est accessible aux jeunes enfants car pour eux c'est un jeu de création et de construction, qui plus est sur un support qu'ils adorent, l'ordinateur. Plus ils se familiarisent tôt avec la programmation informatique, mieux ils appréhendent les concepts de logique associés à cette discipline » (Terosier, C. 2015). C'est dans cette ambition que le Lifelong Kindergarten Group du laboratoire Media du MIT (Massachusetts Institute of Technology) a créé un logiciel pour faciliter l'accès à la programmation informatique chez les enfants dans un environnement ludique et intuitif. Ce logiciel, appelé Scratch, permet la création artistique à travers la conception d'histoires interactives, de jeux vidéo ou de films d'animation. Avec Scratch, l'utilisateur devient concepteur. Ce logiciel permet d'apprendre à programmer avec un langage simplifié. Nous voulons amener les élèves à créer un scénario en définissant une série d'algorithmes. « L'enjeu est la créativité, le code est un moyen » (Terosier, C. 2015). Deux interfaces du logiciel sont disponibles, celle sur ordinateur et celle sur tablette et présentent quelques différences.

### 3.2 Scratch sur ordinateur

Avec un groupe d'élèves de CE2-CM1, nous avons utilisé le logiciel sur ordinateur qui est accessible dès 8 ans. L'écran se partage en 3 zones distinctes, une zone où l'on voit ce que nous programmons et où notre animation prend vie au fur et à mesure des instructions que l'on communique à l'ordinateur, une zone où se trouve toutes les séries d'instructions, appelées « blocs », permettant de faire avancer, tourner, sauter son personnage, et une dernière zone où sont glissés tous les blocs afin de définir le script souhaité : par exemple faire avancer de 15 pas, puis tourner de 40°, puis dire bonjour puis se cacher... En assemblant les blocs par un simple clic de souris, les élèves voient immédiatement le résultat de l'enchaînement d'actions qu'ils proposent. C'est visuel et concret car ils voient le personnage choisi s'animer. Ils doivent résoudre des problèmes de logique afin que les actions s'enchaînent dans le bon ordre. Classés par couleur, les blocs sont regroupés par thème, les blocs liés à l'apparence du personnage, ceux liés au mouvement ou encore ceux qui vont

produire du son. Le logiciel dispose d'une bibliothèque de « lutins » (nom donné par Scratch aux personnages) et de fonds d'écran mais l'élève peut également imaginer et créer son propre personnage ou décor à l'aide de la palette d'artiste mise à disposition. De plus, ils bénéficient de tutoriels pour une prise en main rapide de l'outil, et peuvent également résoudre des problèmes basiques d'enchaînements d'actions. L'élève peut insérer autant de lutins qu'ils le souhaitent et les faire interagir entre eux. Pour chaque lutin, un script différent peut être mis en place mais il est également possible de dupliquer les instructions d'un lutin à un autre. En 15 minutes, un enfant de 8 ans comprend comment le logiciel fonctionne, ensuite il ne s'agit que de manipulations et de tests pour comprendre ce que chaque bloc peut permettre de faire.



*Impression écran de l'interface du logiciel Scratch sur ordinateur.*

### 3.3 Scratch junior sur tablette

Le groupe de grande section de maternelle a utilisé Scratch junior, sur tablette, accessible dès 5 ans. Dans Scratch junior, un programme est un ensemble d'actions associées à un objet numérique (personnages, fruits, véhicules). Ce programme s'écrit avec des blocs « actions ». Développé dans l'optique d'être utilisé dès 5 ans, Scratch junior privilégie les symboles pour que la lecture ne soit pas un frein. Les déplacements sont figurés par des flèches, la possibilité d'écrire est symbolisée par un stylo, celle d'émettre un son par un micro, etc.

Les blocs permettent de s'initier aux principaux algorithmes : les actions de lancement, les



boucles itératives, l'utilisation de variables avec l'envoi de message. Seule la structure conditionnelle n'est pas incorporée dans cette application qui est conçue pour un public d'enfants âgés entre 5 et 7 ans. Le logiciel reprend également les codes graphiques d'un objet bien connu des enfants à cet âge, le puzzle. Les blocs s'accrochent entre eux comme les pièces d'un puzzle, par un « cliquer-glisser ». Grâce à la tablette, la difficulté du maniement de la souris est écartée au profit d'une manipulation tactile.

C'est un outil au service de la conception de projet et d'un enseignement dirigé vers une approche créative, on peut d'ailleurs personnaliser ses personnages et son décor.



*Impression écran de l'interface du logiciel Scratch junior sur tablette.*

## Deuxième partie : Scratch Junior en moyenne section

### 1 Formulation de la problématique

L'état de l'art nous fait prendre conscience d'un décalage entre les besoins actuels des sociétés et la formation des nouvelles générations dans le domaine de l'informatique. Alors que l'informatique a envahi les sociétés à tous les niveaux, l'école a longtemps envisagé l'élève dans l'enseignement de l'informatique uniquement comme un usager. L'un des enjeux actuels est d'inclure l'apprentissage de la programmation à l'école. Par ailleurs ce nouveau domaine à enseigner hérite des nouvelles recherches en éducation. L'apprentissage peut se faire par des activités, on peut apprendre à programmer, ou du moins s'y initier, sans passer par des cours théoriques mais en se plongeant dans un univers de création mettant en jeu les connaissances à acquérir. Un tout nouveau logiciel, Scratch permet cet apprentissage par la réalisation, en faisant appel à la créativité des élèves. Sa version junior sur tablette recommandé dès 5 ans, permet d'initier dès le plus jeune âge, en affranchissant de toute lecture, à l'activité de programmation. Ce cadre théorique m'a amené à la problématique suivante : dans quelle mesure est-il possible de proposer une initiation à la planification d'algorithmes dès la moyenne section, afin que les enfants ne se positionnent comme simple utilisateur ?

La réponse possible à cette question était que les élèves seraient en mesure de comprendre les premières logiques de la pensée informatique en utilisant un jeu éducatif adapté à certaines conditions. Les élèves auraient besoin d'abord de passer par une phase débranchée et manipulative, mais aussi, ensuite, d'un scénario et d'un logiciel pour les guider, et enfin d'une forte présence de l'enseignant.

### 2 Méthode

#### 2.1 Participants

L'expérimentation a lieu auprès de 12 élèves d'une classe de moyenne section d'une école classée REP. La classe compte 24 élèves et le niveau est hétérogène. Elle regroupe des élèves très divers : des élèves allophones, un élève en retard de langage aidé d'un AVS, une élève très dissipée, des élèves timides, des élèves très à l'aise à l'oral, et une élève qui doit faire

l'objet d'un prochain passage anticipé. Le niveau en compétences relatives à la numération est bon.

L'activité sur Scratch Junior s'inscrit à double titre dans le programme de Maternelle. D'une part, elle s'inscrit dans la découverte des outils numériques et d'autre part dans la construction de la représentation de l'espace.

## **2.2 Matériel**

### **2.2.1 Les Choix didactiques**

Mes choix ont été guidés par la nécessité de proposer une approche de la programmation cohérente avec les possibilités des élèves et avec les instructions officielles.

Apprendre à programmer n'est pas une compétence attendue de l'école maternelle. Le premier impératif était de rendre cet apprentissage compatible avec les programmes. En ce sens j'ai cherché à lier une compétence attendue au programme et une utilisation possible de Scratch junior. La carte quadrillée et un scénario de déplacement d'un personnage sont apparus dans mes recherches sur les outils de l'enseignement de la représentation de l'espace comme un outil très fréquent et une opportunité pour lier les deux. Il permet d'utiliser les quatre flèches de déplacement.

La construction de l'espace est un vecteur qui permet donc l'initiation à la programmation et d'aborder l'algorithme avec un enchaînement d'actions de déplacements dans un espace quadrillé.

Via cette activité le choix des actions est réduit. Cette contrainte permet d'attirer l'attention sur une catégorie d'action et d'en tirer partie plutôt que de laisser les élèves découvrir un large éventail d'actions sans arriver à n'en saisir aucun sens. De plus l'algorithme de déplacement permet une mise en action évidente visuellement.

Avant d'arriver à l'utilisation du support informatique à proprement parler, il était nécessaire de passer par une étape intermédiaire. Aborder l'algorithme de déplacement chez l'enfant de 5 ans nécessite inéluctablement une première approche sur un support manipulable. En m'appuyant sur les travaux de Jean Piaget sur la construction de l'espace chez l'enfant et les travaux de Berthelot et Salin qui distinguent plusieurs types d'espaces, il est apparu indispensable d'articuler l'expérimentation dans un aller-retour entre les différents types d'espaces. La construction de l'espace est d'abord perceptive puis représentative à partir de 3 ans selon Piaget. Les élèves de maternelle s'inscrivent dans le passage de l'un à l'autre. Il faut donc leur permettre d'appréhender un dispositif perceptif dans un méso-espace, espace

où ils se déplacent, avant d'utiliser un dispositif représentatif sur un écran relevant du micro-espace, espace entièrement visible sans déplacement (feuille A4, maquette, un écran,...). Dans l'adaptation pédagogique imaginée, il convenait d'assurer une continuité entre l'activité débranchée et l'activité sur tablette. J'ai d'abord conçu l'activité sur le logiciel Scratch avant de le transposer dans un espace perceptif. Le recours au quadrillage a fait émerger quelques difficultés lors de la transposition sur Scratch. Si le logiciel offre la possibilité de faire apparaître un quadrillage, la dimension de celui-ci n'est pas personnalisable.

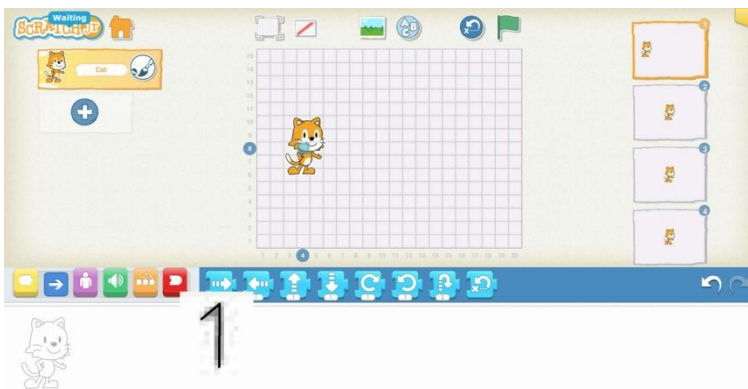


Figure 1 : quadrillage par défaut

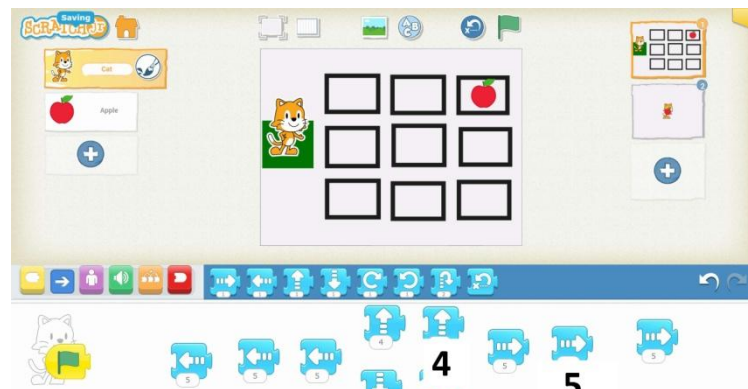


Figure 2 : quadrillage personnalisé

Ses dimensions par défaut ne sont pas accessibles pour des enfants de maternelles. Les diverses activités de moyenne section utilisant un quadrillage en moyenne section ont recours à des dimensions de 2 sur 2 ou 3 sur 3. Il a fallu concevoir un paysage en forme de quadrillage répondant à ses dimensions. Cette adaptation a eu pour conséquence d'adapter les boutons déplacements. En effet les boutons déplacements ont une valeur par défaut qui est à l'échelle du quadrillage par défaut. Pour permettre une animation cohérente du personnage, des boutons déplacements ont été personnalisés à la taille du décor. Ainsi des déplacements verticaux présente la valeur « 5 » et les déplacements horizontaux la valeur de « 4 ». Ces blocs personnalisés sont créés en nombre suffisant pour permettre une latitude suffisante de déplacement. Leur présentation a fait l'objet de nombreuses hésitations. Le dilemme était de rendre visible et accessibles ses blocs tout en n'encomrant pas l'espace dédié aux assemblages en sachant que les blocs personnalisés ne peuvent apparaître que dans cette zone. C'est par tâtonnement qu'il est apparu que les blocs pouvaient être superposés. Cette possibilité a été retenue. On peut ainsi préparer un nombre important de blocs tout en laissant un espace d'assemblage à gauche.

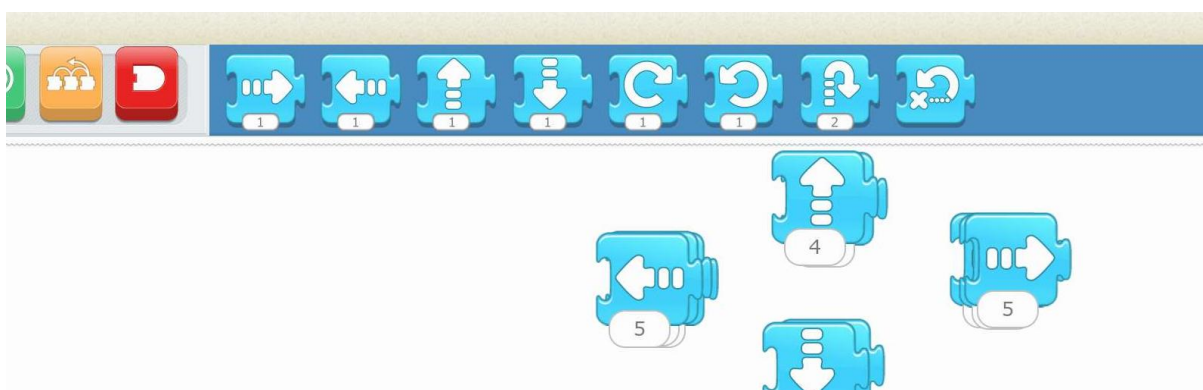


Figure 3 : Disposition superposé

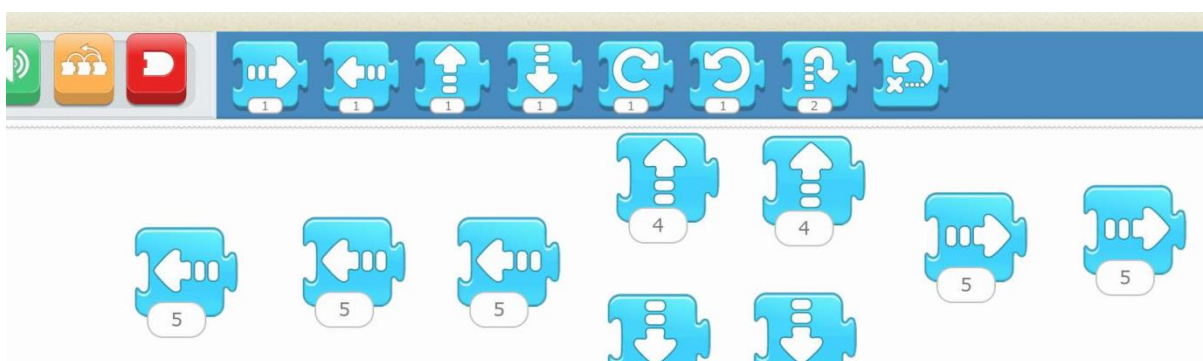


Figure 4 : Disposition juxtaposé

Il s'est également posé un problème de vocabulaire, il n'était pas envisageable de leur dire: vous allez faire un algorithme. Ce mot, pas toujours bien connu des adultes aurait été incompréhensible pour eux. J'ai fait le choix de scénariser l'activité en demandant d'emmener scratch le chat chercher la pomme. Pour justifier l'assemblage des blocs, les exercices évoluent du bloc par bloc à la planification par un petit jeu où il faut laisser « un message » à un enfant caché. Le message est le mot qui me semblait être le plus compréhensible. Dans un second temps, le « message » sert à « dire à la tablette » ce que Scratch doit faire.

### 2.2.2 Choix pédagogiques

L'école ne dispose pas de tablette. Il aurait été possible en réservant dès le début de l'année, avant d'avoir connaissance du thème de notre mémoire, d'obtenir le prêt d'une dizaine de tablettes via le réseau canopé départemental pour quelques semaines. La circonscription de l'école aurait pu mettre à disposition une ou deux tablettes pour une ou deux semaines. Le choix a été fait d'utiliser une tablette personnelle qui présente plusieurs avantages. Premier avantage d'être disponible sur un délai plus long. Ce délai permet

d'ajuster l'activité et de la proposer à des petits groupes d'élèves en APC durant tout un semestre.

Second avantage, la conception de l'adaptation pédagogique de scratch junior n'a pas à être répété sur plusieurs terminaux. Scratch junior n'offre pas encore la possibilité d'importer et d'exporter des projets comme peut le proposer la version de scratch sur ordinateur.

Troisième avantage la supervision de l'enseignant ne se porte que sur une seule tablette. Ne connaissant pas le degré d'autonomie que peuvent atteindre les élèves de moyenne section dans ce dispositif, il vaut mieux prévoir un accompagnement plus élevé que nécessaire plutôt qu'inversement.

Le choix de Scratch parmi les logiciels d'initiation à la programmation repose d'abord sur le fait qu'il s'agit d'un logiciel libre. Cette gratuité permet une accessibilité à tous, si les enfants veulent poursuivre la découverte chez eux. Aussi, de par l'intérêt grandissant pour l'enseignement de la programmation, ce logiciel est promis à un avenir prometteur. Il rencontre déjà un succès auprès des professionnels de l'éducation et est plébiscité par nombre d'entre eux. Une communauté nourrit mois après mois de projets et de situations pédagogiques innovantes (<https://scratch.mit.edu>). Cette notoriété naissante de l'application peut permettre aux enfants d'être dans la suite de leur scolarité à nouveau invité à utiliser Scratch. Pour ma part il s'agissait également d'en avoir une première approche et de poursuivre ensuite son exploration et d'enrichir son utilisation pédagogique à travers les cycles.

## 2.1 Les outils d'analyses

Les outils d'analyses suivants sont utilisés :

- Un relevé de réponse à la question posée au début de la séquence : « à quoi sert une tablette ? ».
- une grille d'évaluation de reconnaissance et de compréhension des blocs.
- une grille d'évaluation de création et de lectures d'assemblages de blocs pour les activités débranchées et branchées. Les blocs « autres » (parler, écrire) font l'objet d'une grille d'évaluation indépendante et le nombre d'exercices réalisés à chaque séance est également noté.
- un relevé de réponse à la question posée à la fin : « à votre avis c'est plutôt vous ou plutôt la tablette qui fait bouger, parler, écrire Scratch ? ».
- diverses observations seront notées selon les difficultés ou facilités rencontrées par les



élèves.

Plusieurs variables sont prévues. La première est le « nombre de blocs ». Dans l'expérimentation la difficulté des exercices dépend du nombre de bloc à assembler. Le nombre maximal de blocs à associer pour un déplacement est de 4. La performance de création des déplacements est mesurée grâce à cette variable.

La seconde variable utilisée est la nature du support. La première séance se déroule sur un support matériel et la seconde séance sur un support numérique, la tablette. Selon la nature du support le nombre d'élève participant entièrement à la séance peut être différent.

La troisième variable est la méthode d'utilisation des blocs. Dans chacune des séances les élèves créent un déplacement soit bloc par bloc soit par la constitution d'un message

### 3 Procédure

#### 3.1 Déroulement de l'étude

Les deux séances sont proposées lors d'APC, à trois élèves, et durent 45 minutes.

Lors de la première séance, l'activité débranchée est mise en place. Un quadrillage de 3 cases sur 3 est matérialisé sur une table à l'aide de feuille. Les flèches et les personnages sont imprimés et découpés sur des feuilles A4. Les flèches sont superposées comme elles le seront dans l'activité branchée.

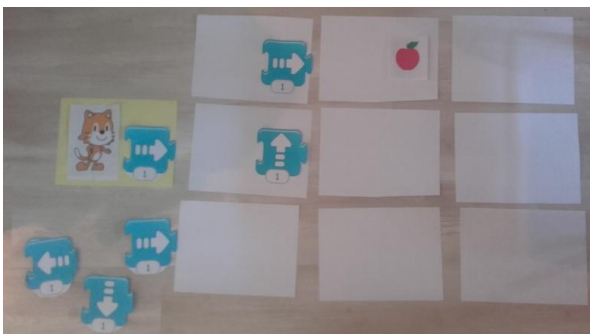


Figure 4 : déplacement bloc à bloc

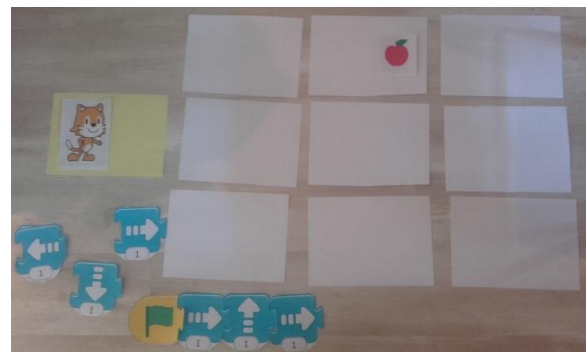


figure 5 : déplacement par message

Dans un premier temps je présente la tablette et je demande ce que l'on peut faire avec une tablette pour eux. Ensuite nous commençons l'activité débranchée par la découverte et la description des éléments visibles sur la table. Puis je leur présente le scénario et je leur montre un

exemple. Lors de ces premiers déplacements les flèches sont posés feuille par feuille. La pomme est éloignée au fur et à mesure des exercices. Ensuite, le jeu évolue après une mise en évidence de la forme en puzzle des blocs et de la possibilité de les assembler en « message », le bloc avec le drapeau est appelé « début du message ». Un enfant doit se cacher avec Scratch et il devra deviner où se trouve la pomme qui sera caché sous une feuille. Un autre enfant doit donc laisser un message. Il a la liberté de concevoir le message en mettant les flèches sur chaque feuille comme précédemment et de les assembler ensuite, ou d'assembler directement le message. L'enseignant est présent pour l'amener à vérifier et à corriger si nécessaire. L'enfant revient et lit le message pour trouver la pomme. La pomme est éloignée de la maison au fur et à mesure des exercices pour agrandir l'algorithme à produire. La séance se termine par une verbalisation de ce qui a été fait.

Lors de la seconde activité, il est rappelé ce que les enfants ont fait la séance précédente et la tablette est allumée. Les enfants doivent découvrir l'environnement du logiciel, le tactile et reconnaître les éléments qu'ils ont déjà rencontrés (le quadrillage, la maison, les personnages, les blocs flèches). Ils doivent ensuite « amener scratch à la pomme de la façon dont il le souhaite ». Ils vont utiliser les flèches par simple pression. Il leur est ensuite rappeler la présence du bloc « début du message » et leur est montré la manœuvre de déplacement des blocs. L'enseignant leur demande d'écrire un message pour que scratch aille à la pomme et leur explique que le message sera « exécuté » en appuyant sur le premier bloc « début du message ». Chacun son tour, les élèves écrivent le message et vérifient en appuyant sur le bouton, se corrigent si besoin. Les autres peuvent l'aider s'il y a une erreur. La pomme est éloignée au fur et à mesure pour agrandir l'algorithme à produire. L'enseignant leur fait découvrir ensuite un nouveau bloc « parler ». Il leur demande quel peut être l'utilité et la signification de ce symbole. Les enfants imaginent un bruit en lien avec le scénario (« j'ai faim », « hum cette pomme est bonne »...), ils l'enregistrent et doivent ensuite le positionner dans la chaîne de l'algorithme en discutant. Un exercice est refait avec ce nouveau bloc. L'enseignant introduit un nouveau bloc « parler » et les élèves imaginent un autre son. Il est introduit au bon endroit du « message ». Un exercice individuel a lieu. L'enseignant leur fait découvrir le bloc « écrire ». Les élèves sont invités à écrire « MERCI » suivi de leur prénom en majuscule ou script. Ce bloc est intégré au message. Un nouvel exercice individuel est réalisé.

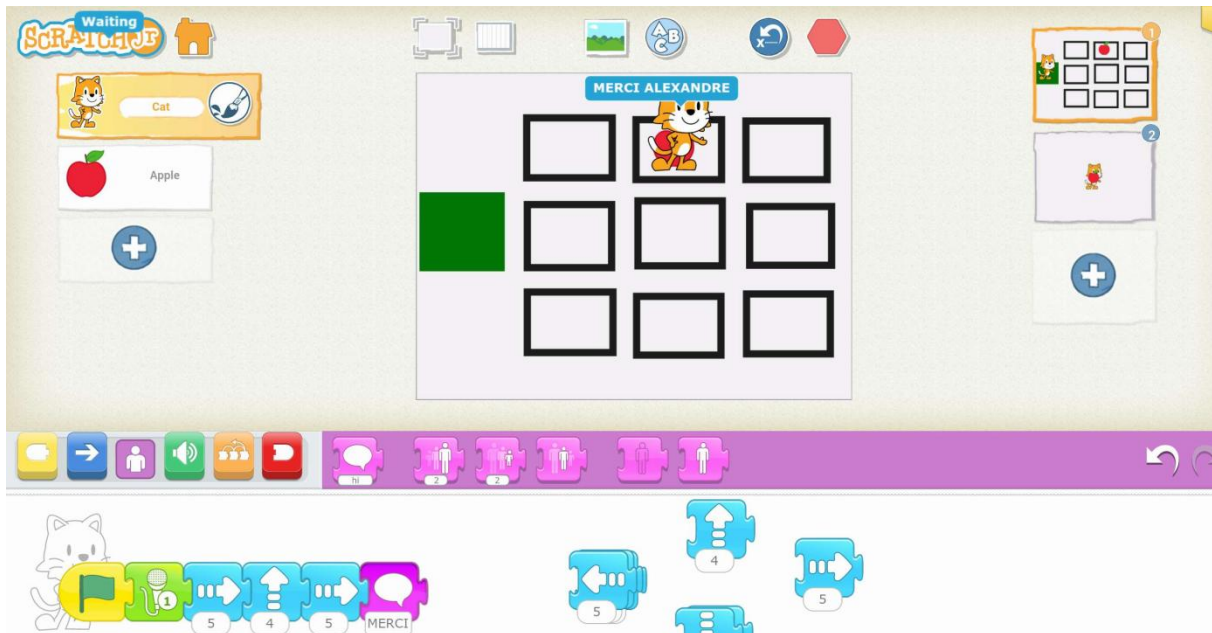
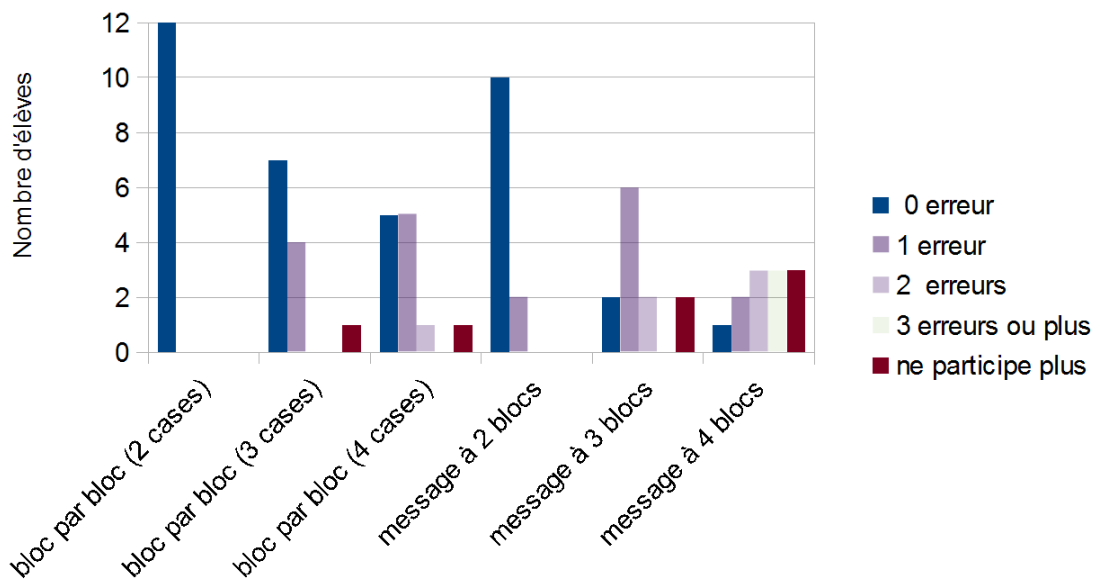


Figure 7 : séquence d'instructions multiples.

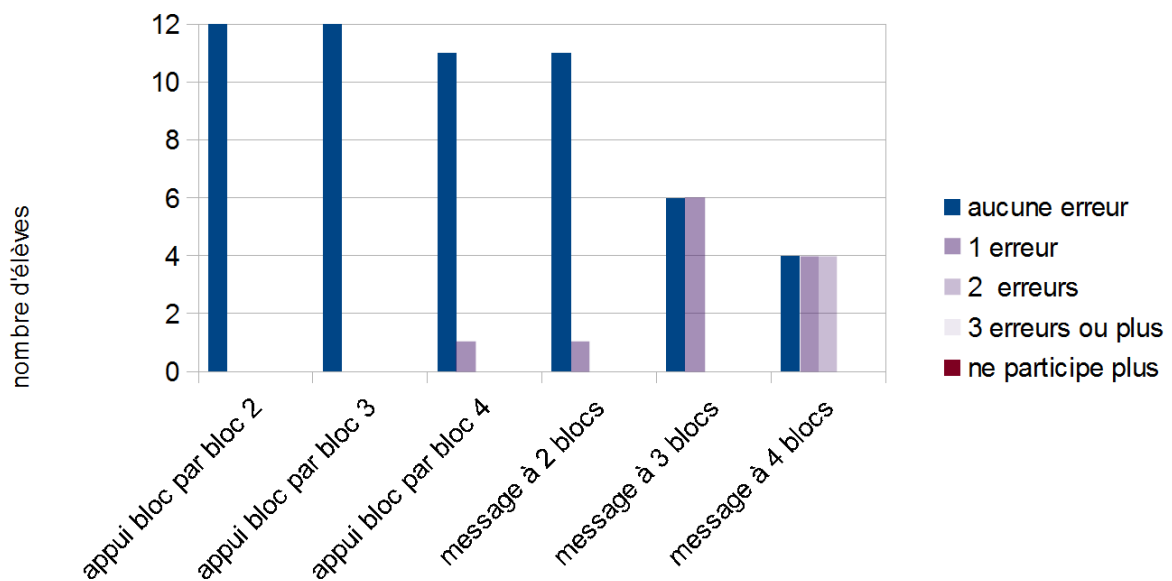
D'autres exercices sont proposés si le temps le permet. Les élèves sont ensuite amenés à verbaliser leur expérience puis la question suivante leur est posée : « à votre avis, c'est plutôt vous ou la tablette qui fait bouger, parler, écrire Scratch ». Les élèves sont invités à justifier leurs réponses.

## 4 Résultats

### 4.1 Des apprentissages en jeu



Graphique 1 : Résultats activité débranchée



Graphique 2 : Résultats activité sur tablette

#### 4.1.1 Blocs par blocs

Sur ces deux premiers graphiques, on peut observer que les exercices de déplacements blocs par blocs sont bien réussis par les élèves aussi bien sur tablette que dans un méso-espace.

Il y a une progression observable entre les deux graphiques. Le nombre de fautes commises

diminue dans les déplacements blocs par blocs entre la première et la seconde séance. On constate que sur un déplacement utilisant trois blocs le nombre d'élèves commettant une erreur évolue de quatre à zéro. Cette diminution est aussi observable sur les déplacements utilisant trois blocs lors desquels le nombre d'élèves faisant au moins deux erreurs passe de six élèves à un élève. Ces changements sont significatifs, un apprentissage est clairement réalisé.

#### 4.1.2 Planification d'un message

Le nombre de fautes augmente lors de la constitution de messages aussi bien avec la manipulation de blocs- feuilles que sur tablette. A partir de la constitution d'un message nécessitant trois blocs papier, seulement deux élèves réussissent sans commettre d'erreur. Lors de l'exercice suivant, seulement un élève y parvient.

Avec l'activité sur tablette on remarque un niveau de réussite sans erreur assez faible. Pour trois blocs associés, seulement la moitié réussit, et pour quatre blocs associés, le nombre descend à quatre.

La phase d'assemblage de blocs pour créer les déplacements paraît être un exercice difficile pour les élèves. On peut expliquer l'émergence de cette difficulté d'abord par la manipulation de plusieurs éléments en mémoire de travail, ensuite et surtout, par le passage à l'abstraction. En effet assembler un message nécessite que les élèves se représentent mentalement le déplacement du personnage au fur et à mesure de l'ajout de bloc dans le message. Il est à noter qu'il n'a pas été précisé aux élèves qu'ils ne pouvaient pas déplacer le personnage. Ce passage par l'abstraction a été naturellement choisi par les élèves bien que porteur de difficulté.

Toutefois le nombre de faute sur la constitution des messages lors de la seconde séance diminue. Les élèves semblent donc s'améliorer et apprendre. Trois fois plus d'élèves réussissent à constituer des messages à trois blocs sans erreurs, et le taux de réussite est quatre fois plus élevé pour les messages à quatre blocs. On peut penser que d'une part les élèves sont en train d'apprendre à constituer des messages et d'autre part que passer par l'abstraction est plus facile sur un micro-espace où le balayage visuel est de moins grande amplitude.

## 4.2 Comparaison des supports

Les exercices sont mieux réussis sur tablette que sur le support papier.

Même si on peut expliquer que l'amélioration des résultats est le reflet de leur apprentissage, il est possible de penser que l'augmentation est amplifiée par l'utilisation de la tablette. La différence de réussite aux exercices de déplacement bloc par bloc est très marquée. Seulement un élève commet une erreur lors du dernier exercice à quatre blocs, alors que dix élèves ont commis au moins une faute lors de ces exercices en activité débranchée. On peut expliquer cela par le fait que la tablette prend en charge le déplacement du personnage et que les élèves n'ont pas à déplacer les blocs sur les feuilles. Ils doivent se concentrer uniquement sur le bloc à choisir avant chaque déplacement.

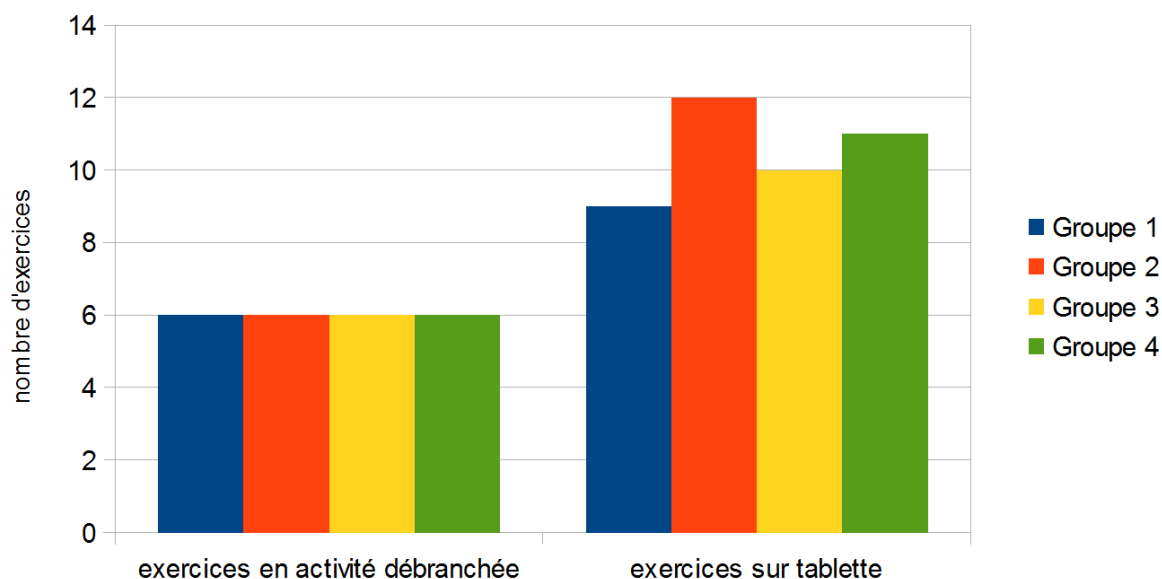
Les résultats montrent que les élèves ont progressé lors de la constitution de message sur la tablette. Le nombre d'élèves qui ne commettent aucune faute n'est cependant pas majoritaire. Dans ce cas de figure l'attention des élèves portait sur les mêmes caractéristiques en phase débranchée qu'en phase sur tablette. Sur les deux supports les élèves avaient à déplacer des blocs et les assembler entre eux. La différence intervenait surtout au moment de la vérification où l'élève appliquait son algorithme soit en déplacement manuellement soit en déléguant la tâche à la tablette.

## 4.3 Participation

On remarque que lors de la phase débranchée, un élève a cessé de participer lors des exercices bloc par bloc, et que trois élèves n'ont pas participé aux exercices de constitution de blocs. On pourrait penser que la difficulté de l'exercice est la cause du désintérêt de ces élèves, en particulier lors du recours à l'abstraction.

Ce ne semble pas être le principal facteur puisque l'on remarque, à la fois qu'un élève avait cessé de participer lors de la première série d'exercice, plus simple, et aussi qu'aucun élève n'a cessé de participer lors des deux séries d'exercices sur tablette. La tablette a peut-être été porteuse d'un facteur d'attractivité dont a été dénué l'activité débranchée pour ces élèves.

#### 4.4 Plus d'exercices sur tablette



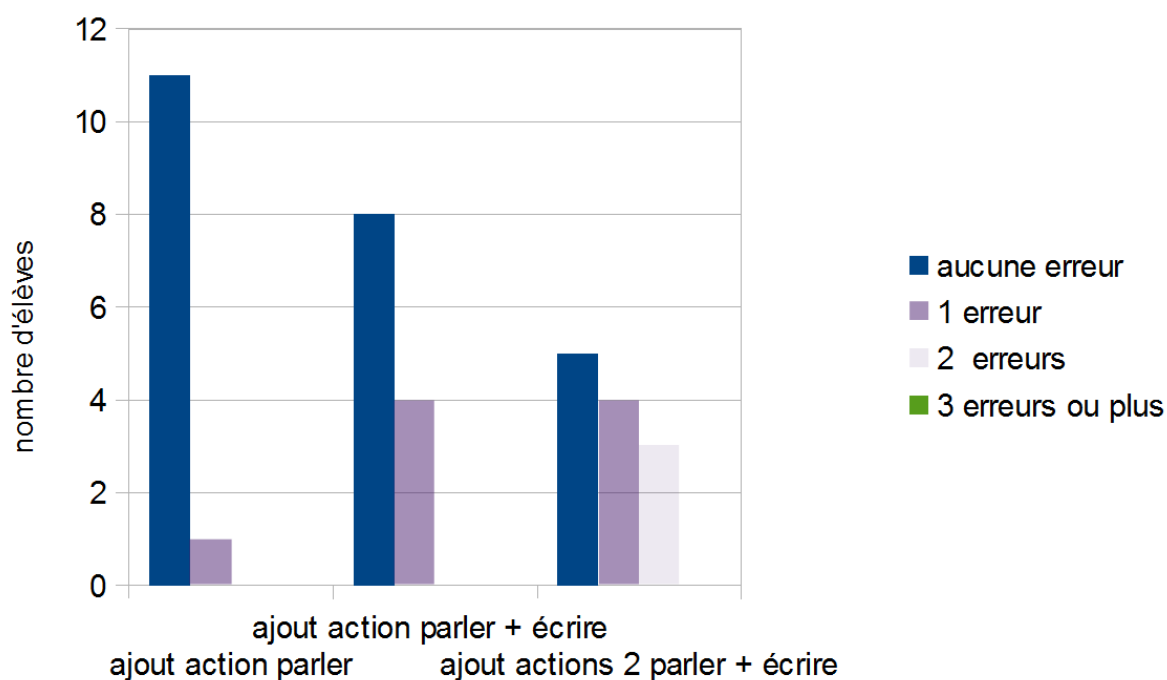
**Graphique 3 : Nombre d'exercices réalisés selon le support**

Sur le graphique 3, on peut constater qu'un plus grand nombre d'exercice par élève a pu être fait sur tablette. En phase débranchée, seuls les deux séries d'exercices sur le déplacement du personnage ont été réalisés avec les quatre groupes, trois en bloc par bloc, et trois en message, soit six au total.

Le nombre d'exercices réalisés sur tablette n'est pas identique selon les groupes et est supérieur à celui de l'activité débranchée. Les exercices supplémentaires réalisés sont d'abord au nombre de trois, ceux avec les blocs parler et écrire, qui étaient ajoutés à un message constitué. Le groupe 1 a réalisé ces trois exercices uniquement supplémentaires lors de l'utilisation de la tablette. Les autres groupes ont eu le temps de faire davantage d'exercices, le groupe 2 a pu faire trois exercices supplémentaires, le groupe 3, un et le groupe 4 deux. Non-évalués ces exercices ont permis d'aborder d'autres blocs, agrandir, disparaître, ... Il a été possible avec le groupe 2 en particulier de perfectionner le scénario en faisant grandir Scratch à son arrivé sur la pomme, et surtout de créer une séquence parallèle de la pomme qui disparaissait au contact de Scratch. Le logiciel offre une multitude de possibilités pour enrichir la situation avec des élèves en réussite.



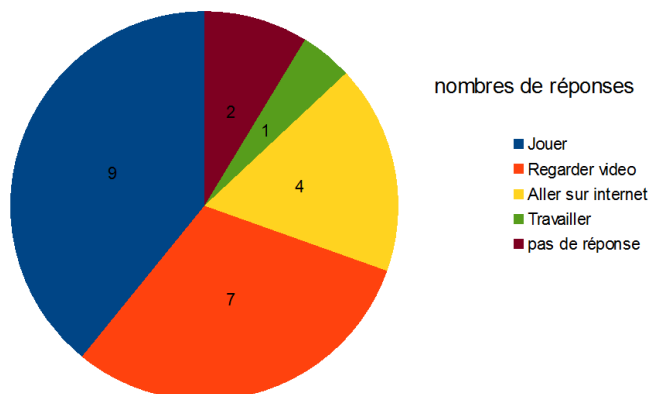
#### 4.5 La diversité des blocs



**Graphique 4 : Résultats séquence avec actions « autres »**

Lors de la phase débranchée, des blocs actions différents des déplacements ont été utilisés. Ces blocs ont été intégrés à un message de déplacement déjà élaboré. On peut donc considérer qu'il s'agit de l'association d'un élément à un autre, d'un groupe de déplacement à une nouvelle action. La première donnée en abscisse peut correspondre à un niveau de difficulté équivalent au message à deux blocs, ... Dans cette comparaison entre les trois constitutions de messages, les élèves réussissent au moins aussi bien avec l'assemblage de blocs d'actions différents du déplacement. En comparant le graphique 4 avec le graphique 2, on constate que huit élèves réussissent sans erreur contre six pour l'association de trois éléments, et cinq contre quatre pour l'association de quatre éléments. Les actions qui font appel à d'autres compétences de la part de l'élève, écrire en majuscule, enregistrer sa voix, permettent aux élèves de donner un sens supplémentaire à l'exercice et d'augmenter leur attention et leur implication.

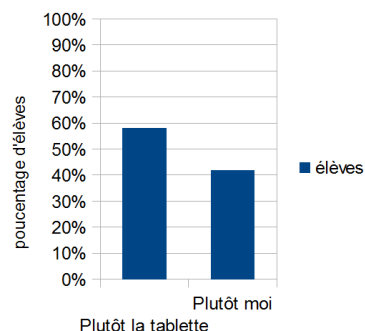
## 4.6 Conception de l'outil numérique



**Graphique 5 :**

**Résultats à la question :**

« Que peut-on faire avec une tablette ? »



**Graphique 6 :**

**Résultats à la question :**

« Selon toi, c'est plutôt toi ou la tablette qui fait bouger, parler, écrire Scratch ? »

L'un des enjeux de cette séquence, au-delà de l'initiation implicite à l'algorithme, est de faire prendre conscience de la posture consciente adoptée face aux nouvelles technologies. J'ai testé par des questions si la conception des élèves évoluait et s'ils pouvaient se considérer comme des créateurs plutôt que de simples usagers. A la question initiale sur la fonction d'une tablette, les résultats montrent que seul un élève associe la tablette à une fonction de travail. Cette fonction de travail n'est certes pas forcément synonyme d'une posture de créateur. On peut travailler sur des logiciels en tant qu'utilisateur mais c'est la seule qui l'inclut. Quatre élèves pensent que la tablette peut servir à aller sur internet, une fonction qui peut à la fois servir le travail ou le divertissement. La tablette est essentiellement envisagée comme un outil de divertissement avec les fonctions de vision de films et de jeu identifiées respectivement par sept et neuf élèves.

A l'issue de la séquence 42% des élèves répondent que les mouvements qu'effectue le personnage sont plus de la responsabilité des élèves que de la tablette. Ce pourcentage est élevé quand on le compare au nombre d'enfants qui pensait que l'on pouvait travailler avec une tablette. Ces données ne sont pas directement comparables mais chacune de ces réponses est connotée de l'idée d'agir sur la tablette.

## 5 Discussion

### 5.1 Confrontation aux hypothèses

L'expérimentation a mélangé trop de variables pour pouvoir mettre en avant des résultats évidents. Malgré de nombreux défauts, je vais essayer d'exploiter les résultats obtenus et tenter d'apporter quelques éléments de réponses aux hypothèses de cette étude.

#### 5.1.1 Apprentissage réalisé

Des apprentissages ont lieu dans la conception de séquence d'actions. Dans les exercices bloc par bloc la réussite devient très élevée et pour ce qui est des exercices de constitution de message, il n'y a pas de réussite élevée mais une nette amélioration des résultats. Cet apprentissage semble donc accessible à des élèves de moyenne section.

#### 5.1.2 Nécessité de la phase débranchée

Il est difficile de tirer des conclusions sur la nécessité de la phase débranchée avec l'expérimentation réalisée. Elle peut paraître avoir sa place lorsque l'on voit l'amélioration des résultats entre les deux séances mais il est impossible de dire si sans la phase débranchée et manipulatoire l'amélioration aurait été identique ou différente. J'ai pu observer que le repérage dans l'espace semble plus difficile dans un méso-espace avec l'amplitude de balayage des yeux. N'est-ce pas cependant une phase indispensable malgré la difficulté pour ensuite pouvoir passer à un balayage visuel plus fin, comme l'enfant évolue d'une motricité de déplacement à une motricité fine.

#### 5.1.3 Utilité d'un scénario et d'un environnement personnalisable.

Plus le scénario était élaboré, plus les taux de réussite et de participation étaient élevés. Assez pauvre sur le support débranché, le scénario s'est enrichi de sons et d'écritures avec l'utilisation de Scratch. Si les taux de réussite peuvent être expliqués par d'autres facteurs que la tablette, le taux de participation lui indique un attrait supplémentaire des élèves.

L'intérêt des enfants semble se constater à travers les bons résultats sur l'ajout de blocs « autres », qui permettent l'enrichissement du scénario et font appel à d'autres compétences.

Parmi ces autres compétences, la créativité a une place de choix. Un élève qui avait le moins participé lors de l'activité débranché, a radicalement changé son comportement en découvrant l'activité sur tablette. Personnaliser les couleurs des personnages et des objets, enregistrer ses

propres sons, imaginer son scénario l'a plongé totalement dans l'activité et le nombre d'exercice qu'il a réalisé est l'un des plus élevé.

#### **5.1.4 Utilisation du logiciel**

Outre la possibilité de faire appel à la créativité, le logiciel semble être un véritable atout. C'est sûrement le résultat le plus probant de cette étude. A travers le nombre d'exercices réalisés dans le même temps imparti selon les deux supports, le logiciel apparait comme étant un outil bien plus efficace. La rapidité des déplacements des blocs, lorsqu'ils étaient maîtrisés, a permis aux élèves d'effectuer les exercices plus vite que sur le support de l'activité débranchée. Il faut ajouter à cela la vérification offerte par l'animation de la séquence réalisée qui prend en charge cette tâche à la place de l'élève et qui lui permet de se concentrer sur les rectifications à apporter.

#### **5.1.5 Présence de l'enseignant**

Sa présence n'a pas été évaluée mais ce sont mes différentes observations qui m'amènent à justifier le dispositif en petit groupe encadré. Les élèves sont confrontés à une tâche à laquelle ils sont complètement novices. Le meilleur choix pédagogique dans ce cas est celui de l'enseignement explicite. L'autonomie des élèves n'est donc pas envisageable en deux séances, et lors de ces dernières le rôle de l'enseignant, induit une présence continue, pour expliciter toutes les objectifs, les stratégies de résolution.

Les difficultés rencontrées par les élèves lors de l'utilisation du tactile renforce l'idée de la présence de l'enseignant. L'utilisation du tactile, qui est un atout du logiciel Scratch sur tablette, affranchissant les élèves de l'utilisation de la souris s'est révélée être problématique pour certains élèves. Ces derniers ne parvenaient pas à déplacer les blocs soit à cause d'appuis discontinus, soit de déplacements trop rapides. Par ailleurs des appuis non-centrés, entraînant une ouverture du pavé tactile pour configurer la valeur du déplacement, ont également bloqué des élèves dans leur résolution des exercices. La présence de l'enseignant s'est avérée alors indispensable pour proposer l'aide adaptée à chacun de ces problèmes rencontrés. Il aurait été possible pour rendre plus autonome les élèves de prévoir un temps d'apprentissage, en amont de la séquence, sur l'utilisation du tactile et les déplacements des blocs du logiciel.

#### **5.1.6 Posture face à l'outil numérique**

Là encore, rien ne peut être affirmé à cause du manque d'homogénéité des questions mises en

place. Les résultats semblent démontrer qu'une proportion non négligeable d'enfants comprend à la fin de la séquence le rôle qu'ils ont tenu face à l'outil numérique alors qu'au début un seul estimait que l'on pouvait travailler avec une tablette.

## 5.2 Limites et perspectives

La variable des différents supports auraient pu être mieux exploitée pour aboutir à une comparaison plus précise des supports. Pour mieux évaluer les apports du logiciel il aurait été possible que deux groupes soient séparés lors de la deuxième séance avec deux dispositifs différents s'inscrivant dans un micro-espace. Un groupe aurait réinvesti les connaissances dans les exercices mis en place dans cette expérimentation, sur un support numérique, et un autre groupe dans des exercices similaires sur une feuille A4, sur un support matériel.

On aurait alors pu comparer les résultats aux exercices, le nombre d'exercices réalisés de ces deux groupes pour constater les effets de la tablette. Cette mise en place n'aurait cependant été possible que sur des algorithmes de déplacements.

La nécessité de l'activité débranchée pour les élèves de moyenne section aurait pu être testée en constituant également deux groupes. Un des groupes aurait suivi le protocole de cette expérimentation et l'autre groupe n'aurait pas fait cette activité débranchée. Il aurait été directement soumis aux activités sur tablette. Ce second groupe aurait constitué un groupe témoin. La différence de réussite aux exercices aurait mis en évidence la nécessité ou non de cette activité débranchée.

Il n'est pas démontré non plus clairement la nécessité du scénario pour guider les élèves dans leurs apprentissages. Il aurait fallu la-aussi créer des sous groupes où les enfants auraient appris soit au sein du scénario proposé ici, soit en l'absence de scénario.

Une autre limite de l'expérimentation porte sur l'étude de la conception que l'on a de soi face à l'outil numérique. Les deux questions et les réponses n'étant pas identiques, il est assez difficile de les comparer et de mettre en avant une évolution dans cette conception. Une même question aurait dû être posée au début et à la fin. La première question qui aurait pu être posée après une démonstration par le maître de l'utilisation du logiciel : « qui fait bouger Scratch ? Plutôt le maître ou plutôt la tablette ? ».

Le nombre de séances trop faible et la petite taille de l'échantillon sont également deux autres limites à cette étude. Tiré des jugements sur un apprentissage en deux séances peut paraître précoce. Le faible nombre d'élèves ayant participé à l'expérimentation constitue un échantillon trop faible et ne peut être représentatif de la population, de ce fait les résultats ne peuvent pas être généralisables. Ces limites étaient cependant difficilement franchissables par le besoin de la présence continue de l'enseignant pour un petit groupe d'élève.

### 5.3 Autres problèmes rencontrés

La disposition des blocs en superposition a permis une transposition mais elle n'est pas exempte de défaut. La proximité des blocs a conduit les élèves à assembler par inadvertance certains blocs créant des chaînes qui pouvaient bloquer l'activité. Le désassemblage ne peut se faire que par retrait du bloc le plus à droite. Si l'on sélectionne un autre bloc on active le déplacement de tout ou partie du message assemblé. Tous les blocs d'une séquence pouvaient se retrouver assemblés, immobilisés et inutilisables.

La position de la maison a mis en échec plusieurs élèves. Le premier déplacement ne pouvait être qu'« avancer » pour aller de la maison au quadrillage. Or le premier réflexe souvent des élèves était d'ajuster la position de scratch sur le plan de la verticalité. Scratch se retrouvait alors hors quadrillage et l'absence de case a conduit à bon nombre d'erreurs. Les enfants considéraient que suite à un premier mouvement vers le bas ou le haut, scratch se situait dans la case du bas ou du haut de la première colonne et c'est ce qu'ils faisaient lors de l'activité débranchée où ils déplaçaient manuellement le personnage. Lors du passage sur la tablette, ils ont pu se rendre compte de la réalité du déplacement et constaté leur erreur d'appréciation.

Mon intention en mettant la maison hors du champ de déplacement, à l'écart était de bien matérialiser le bloc départ. J'avais également matérialisé ce bloc en colorant la case maison en vert. Il aurait peut être été plus judicieux de n'utiliser qu'une seule méthode de matérialisation, la coloration et de placer la maison à l'intérieur du champ de déplacement en rajoutant deux cases, au dessus et en dessous, autour de la « maison ».

Le regard de certains parents sur cette activité a été surprenant. En prolongement il est proposé à des parents, s'ils disposent d'une tablette, d'installer le logiciel et de permettre à leur enfant de réinvestir les connaissances et de continuer l'exploration de Scratch. Certains parents étaient ouvertement récalcitrants et considéraient qu'il ne fallait pas exposer davantage les enfants aux écrans qu'ils ne s'y exposaient déjà. Malgré les avoir informé qu'il s'agissait non

pas d'exposer davantage mais différemment les enfants aux écrans à travers un outil pédagogique, le discours semblait ne pas changer. Il aurait alors fallu présenter aux parents ce travail lors d'une réunion, ou dans le cahier de vie pour qu'ils en comprennent la valeur pédagogique et ne pensent pas qu'il s'agit d'une exposition divertissante supplémentaire à des médias numériques.

## **6 Conclusion**

Les résultats ne sont pas aussi précis qu'attendus, et il est difficile de mesurer les effets des différents choix opérés. Les hypothèses étaient trop ambitieuses par rapport à l'expérimentation. Cependant la situation s'est révélée efficace sur le plan des apprentissages, et la séquence utilisée mériterait d'être utilisée à nouveau en lui aménageant quelques modifications. Il s'agit en particulier de la faire évoluer pour conduire les élèves à explorer ensuite le logiciel avec davantage d'autonomie, objectif vers lequel tend l'enseignement explicite. Par ailleurs cette étude m'a permis de comprendre les objectifs d'apprentissage de la programmation à l'école primaire et m'a formé à l'utilisation d'un outil pédagogique qui peut être utilisé à travers tous les cycles. Enfin, l'initiation à la programmation en moyenne section a révélé requérir un encadrement privilégié et un investissement de temps qui se justifient totalement lorsque l'on souhaite réduire les lacunes en ce domaine vécues par les dernières générations d'écoliers.



## Bibliographie

- Audureau, W. (2016). *On a testé... Cubetto, le robot jouet qui apprend aux enfants à programmer*. Article du journal Le Monde repéré à [http://www.lemonde.fr/pixels/article/2016/05/06/on-a-teste-cubetto-le-robot-jouet-qui-apprend-aux-enfants-a-programmer\\_4914784\\_4408996.html](http://www.lemonde.fr/pixels/article/2016/05/06/on-a-teste-cubetto-le-robot-jouet-qui-apprend-aux-enfants-a-programmer_4914784_4408996.html)
- Berry, G. (12/03/2014). *Enseigner la science informatique à l'école*. Emission : Rue des Ecoles sur France Culture.
- Bihouée, P. & Coliaux, A. (2011). *Enseigner différemment avec les TICE*. Paris : Editions d'Organisation Eyrolles
- Bourassa, B., Serre, F. & Ross, D. (1999). *Apprendre de son expérience*. Canada : Presses de l'Université du Québec
- Curien, N. & Muet, P-A. (2004). *La société de l'information*. Paris : La Documentation française
- Davidenkoff, E. (2014). *Le Tsunami numérique*. Paris : Stock
- Devillard, A. (2015). Le Pouvoir des algorithmes. *Sciences et avenir*, n° 819, (pp. 30 - 36)
- Direction du numérique pour l'éducation du Ministère de l'Education Nationale (2015). Le numérique, les apprentissages et la réussite de tous les élèves. Repéré à <http://forum.ecolenumerique.education.gouv.fr/debat/le-num%C3%A9rique-les-apprentissages-et-la-r%C3%A9ussite-de-tous-les-%C3%A9l%C3%A8ves>
- Flichy, P. (2010). *Le sacre de l'amateur. Sociologie des passions ordinaires à l'ère du numérique*. Paris : Le Seuil - La République des idées
- Julo, J. (2002). Des apprentissages spécifiques pour la résolution de problèmes ? *Grand N*, n° 69, (pp. 31 - 52)
- Lifelong Kindergarten Group Repéré à <https://ilk.media.mit.edu/>
- Minichiello, F. (2014). L'enseignement du code à l'école. Repéré à <http://www.ciep.fr/revue-internationale-deduction-sevres/pedagogie-et-revolution-numerique/lenseignement-code-a-lecole>
- Ministère de l'Education Nationale. Bulletin officiel spécial du 26 novembre 2015.
- Ministère de l'Education Nationale (2016). Refonder l'école numérique. Repéré à <http://www.gouvernement.fr/action/l-ecole-numerique>
- Naughton, J. (2012). Why all our kids should be taught how to code. Repéré à <http://www.theguardian.com/education/2012/mar/31/why-kids-should-be-taught-code>
- Pasquinelli, E., Zimmerman, G., Bernard-Delorme, A., & Descamps-Latscha, B. (2013). *Les écrans, le cerveau... et l'enfant..* Paris : Le Pommier
- Perucho, I. (2016). *Programmer avec Scratch Atelier Canopé des Yvelines*
- Porter, I (2015). L'école à l'heure de la programmation. Repéré à <http://www.ledevoir.com/societe/actualites-en-societe/458435/l-ecole-a-l-heure-de-la-programmation>
- Scott, J. (2012). Bien commencer avec Scratch - Introduction à l'informatique. Repéré à [http://classetice.fr/IMG/pdf/starting\\_from\\_scratch\\_tutor\\_fr\\_mcc\\_v0.pdf](http://classetice.fr/IMG/pdf/starting_from_scratch_tutor_fr_mcc_v0.pdf)
- Tchounikine, P. (2016). *Initier les élèves à la pensée informatique et à la programmation avec Scratch*. Grenoble : Université Grenoble-Alpes
- Terosier, C. (2015). L'ordinateur un crayon pour créer *Bulletin de la société informatique de France*, n° HSI

Vassiliou, A. & Kroes, N. (2014). Vassiliou urges Education Ministers to help kids crack the code. Repéré à [http://europa.eu/rapid/press-release\\_IP-14-887\\_en.htm](http://europa.eu/rapid/press-release_IP-14-887_en.htm)

Wikipédia (2016). Société de l'information. Repéré à [https://fr.wikipedia.org/wiki/Soci%C3%A9t%C3%A9\\_de\\_l'information](https://fr.wikipedia.org/wiki/Soci%C3%A9t%C3%A9_de_l'information)

Wing, J. (2006). Computational thinking *Communications de l'ACM*, n° 3

Computer sciences unplugged. Repéré à <http://csunplugged.org>

Année universitaire 2015-2016

**Master 2 Métiers de l'enseignement, de l'éducation et de la formation**  
**Mention Premier degré**

Titre du mémoire : **Initiation à la programmation informatique à l'école  
maternelle avec Scratch Junior dès la moyenne section**

**Auteur** : Nathanaël Crauet avec la collaboration de Marie Baatsch.

**Résumé** : Il existe un décalage entre les besoins actuelles des sociétés et la formation des nouvelles générations dans le domaine de l'informatique. L'un des enjeux actuels est d'inclure l'apprentissage de la programmation à l'école. Cet apprentissage peut se faire sans cours théoriques mais par des activités qui mettent en jeu les connaissances à acquérir dans un environnement créatif. Un tout nouveau logiciel, Scratch le permet. Sa version junior sur tablette recommandée dès 5 ans, permet d'initier dès le plus jeune âge. Avec l'utilisation de cette application j'ai testé dans quelle mesure il est possible de proposer une initiation à la planification de séquences d'actions dès la moyenne section, avant que les enfants ne commencent à se positionner comme simple utilisateur. Cette expérimentation vise à mesurer les apprentissages effectués et à tester différents paramètres. Il apparaît que des apprentissages se réalisent grâce à la présence d'une activité débranchée, d'un enseignement explicite, d'un recours aux aspects créatifs du logiciel et d'une présence continue de l'adulte. Les résultats qu'amènent cette étude ne sont que parcellaires. La taille de l'échantillon et la durée des apprentissages sont trop faibles. Plusieurs paramètres sont testés simultanément par manque de moyen. Les résultats auraient donc besoin d'être affinés.

**Mots clés** : Programmation, Scratch, tablette, activité débranchée, maternelle, MS, initiation, algorithme.

**Summary** : There is a gap between the current needs of society and the new generations computing knowledge. One of the current challenges is to include learning programming in school. This learning can be done without lectures but through activities in a creative environment.

A new software, Scratch allows it. Its junior version of tablet is recommended from 5 years. With this application I tested how far it is possible to offer an introduction to planning of action sequences to pre-school children, before child begin to position itself as a simple user. This experiment aims to measure the learning acquired and test different parameters. It appears that apprenticeships are realized thanks to an unplugged activity, explicit instruction, a creative use of aspects of the software and a continued presence of the adult. The results of this study are only patchy. The sample size and duration of apprenticeships are too low. Several parameters are tested simultaneously by lack of means. The results would therefore need to be refined.

**Keywords**: program, programming, tablet, pre-school, Scratch, introduction, unplugged activity.