

Table des matières

Introduction	3
Partie 1 - Numérique et informatique à l'école : définitions, évolutions et perspectives.....	5
<i>1. Numérique et informatique : représentations et définitions des concepts</i>	<i>6</i>
1.1 Qu'est-ce que la science informatique ?	6
1.2 Qu'est-ce ce que le numérique aujourd'hui ?	8
1.3 Relation entre numérique et informatique	9
<i>2. Numérique et informatique : repères historiques sur les conceptions de ce qu'il faut enseigner à l'école</i>	<i>10</i>
2.1 Un mouvement de balancier entre deux conceptions	10
2 .1.1 Années 1970-1980 – Les premiers pas	10
2.1.2 Années 1990-2000 – L'informatique comme outil : une logique d'usage et un régime de la certification sans curriculum	11
2.1.3 Années 2010 – Reconsidération de la question de l'informatique comme objet d'enseignement	11
2.1.4 Les facteurs à l'origine d'une reconsidération de la question de l'informatique à l'école	12
<i>3. Etat actuel de l'enseignement de l'informatique en France et dans des pays voisins « pionniers »</i>	<i>15</i>
3.1 Situation en France	15
3.1.1 Au lycée	15
3.1.2 Au collège	16
3.1.3 A l'école primaire	16
3.2 Situation dans des pays « pionniers » : cas de l'Angleterre et la Suisse francophone	17
3.2.1 Situation en Angleterre	17
3.2.2 Situation en Suisse	18
<i>4. Quelles perspectives pour un enseignement de l'informatique en France dans une optique curriculaire ?</i>	<i>19</i>
4.1 Un enseignement qui se structure, mais la possibilité d'une scission entre le primaire et secondaire	19
4.2 L'importance des ressources pédagogiques pour amorcer le mouvement	19
Partie 2 - Elaboration d'un modèle de scénarisation pédagogique pour l'enseignement de l'informatique à l'école élémentaire dans une optique curriculaire	21
<i>1. Présentation du projet de recherche IE-CARE et questions initiales</i>	<i>23</i>
1.1 Présentation du projet de recherche ANR IE-CARE	23

1.2 Questions initiales de fonds soulevées dans le cadre du projet IE-CARE et réponses apportées	24
1.2.1 Quelle approche de l'informatique ?	24
1.2.2 La question de l'instrumentation.....	26
1.2.3 L'orchestration de la classe en contexte instrumenté par le numérique	28
1.2.4 Un questionnement sur la notion de pensée informatique	29
<i>2. Elaboration et expérimentation d'un scénario-test.....</i>	32
2.1 Postulats de départ.....	32
2.2 Présentation du scénario test.....	33
2.3 Expérimentation du scénario-test	34
2.4 Discussion	36
<i>3. Elaboration d'un modèle de scénarisation pédagogique</i>	36
3 .1 Cadre d'usage et éléments constitutifs du modèle.....	36
3.2 Questions initiales et réponses.....	36
3.3 Modèle de scénario pédagogique : présentation.....	38
3.3.1 Approche pédagogique globale	38
3.3.2 Architecture globale d'un scénario	38
3.3.3 Langage de description	39
3.3.5 Formalisme de diffusion : les graphes d'orchestration	42
3 3.6 Discussion	43
Conclusion.....	45
Bibliographie.....	47
Index.....	1
Annexe 1 - La pensée informatique.....	2
Annexe 2 – Scénario <i>Alphabet de l'informatique avec Ozobot</i>	5
Annexe 3 – Résultats de l'expérimentation du scénario-test.....	15
Résumé	20

Introduction

Alors que plusieurs pays ont introduit la science et la pensée informatique dans leurs curriculums, que le codage a fait son entrée dans les programmes français de 2016 et que les initiatives se multiplient en faveur d'un enseignement de la science informatique en France dès le plus jeune âge, le projet de recherche ANR IE-CARE¹ (2019-2021) associant des chercheurs en informatique et en sciences humaines et sociales se fixe l'objectif de proposer des conditions et des modalités durables pour un enseignement/apprentissage de l'informatique sur les trois premiers cycles de l'école dans une optique curriculaire en abordant la question des conceptualisations pouvant être construites à chaque âge.

Plusieurs obstacles se dressent sur la route d'un tel objectif, aux premiers rangs desquels l'absence d'un curriculum établi des contenus à enseigner en primaire, le manque de culture informatique des professeurs des écoles et l'hétérogénéité des ressources (scénarios pédagogiques et instrumentation) pour soutenir la mise en œuvre de cet enseignement/apprentissage au sein des classes.

Le travail présenté se situe suivant ce dernier axe. Réalisé dans le cadre du projet IE-CARE au sein de l'équipe MOCAH² du LIP6³, il a pour objet l'élaboration d'un modèle de scénarisation pédagogique qui servira de cadre à la production de scénarios d'enseignement/apprentissage de l'informatique.

La construction de ce modèle de scénarisation pédagogique pour initier et soutenir un enseignement/apprentissage de l'informatique à l'école ayant pour objet la construction de conceptualisations dès le plus jeune âge, constitue la question centrale de ce mémoire autour de laquelle s'articulent ses deux grandes parties.

¹ Informatique à l'Ecole : Accompagnement, Conceptualisations, Ressources - <https://anr.fr/Projet-ANR-18-CE38-0008>

² Modèles et Outils en ingénierie des Connaissances pour l'Apprentissage Humain

³ Laboratoire d'Informatique de Sorbonne Université

La première s'attache à définir ce qu'est l'informatique en tant que science et à dégager les leviers et scénarios possibles pour son enseignement sur l'ensemble du cursus scolaire. Elle met à jour les grands principes sur lesquels repose l'informatique, qui l'ont progressivement conduit au statut de science et justifient son enseignement, et pose la question de savoir si cet enseignement/apprentissage est envisageable en France de la maternelle au lycée comme cela est le cas dans d'autres pays.

La seconde partie expose les questions et solutions apportées pour élaborer un modèle de scénarisation pédagogique destiné à l'enseignement/apprentissage de la science informatique à l'école élémentaire ayant pour objectif central la construction de conceptualisations propres au domaine, en rupture avec les pratiques actuelles. Cette partie donne une description du modèle et un exemple de scénario construit dans ce cadre.

Partie 1 - Numérique et informatique à l'école : définitions, évolutions et perspectives

De quelle informatique parle-t-on lorsque l'on parle d'enseigner l'informatique à l'école ? La plupart des personnes disent « faire de l'informatique » lorsqu'elles remplissent un *Doodle* ou qu'elles saisissent un document sur un traitement de texte. Une étude menée en 2016-2017 dans le cadre du projet de recherche ANR DALIE⁴ montre que les représentations de l'informatique des étudiants stagiaires en Master MEEF restent très majoritairement associées aux notions d'outils et d'usages qui relèvent pourtant du champ du numérique. Seul un petit nombre d'entre eux a une conscience de l'informatique comme science du traitement automatisé de l'information (Drot-Delange, 2018). Probablement les évolutions extrêmement rapides de l'informatique au cours de ces soixante-dix dernières années et la pénétration de ses applications dans les sphères personnelles et professionnelles sont-elles à l'origine de ces représentations. Quoi qu'il en soit, ces données engagent à opérer d'emblée une clarification. Qu'est-ce que l'informatique et pourquoi est-elle aujourd'hui considérée comme une science ? En quoi se différencie-t-elle de ce que l'on appelle le numérique et quelles sont les relations entre les deux ?

Dans le champ éducatif, une approche utilitaire de l'informatique laissant de côté la dimension scientifique a longtemps dominé. Mais actuellement, sur la base des arguments d'une informatique devenue une science incontournable, le codage a fait son entrée dans les programmes de l'école tandis qu'un enseignement de la science informatique se structure au lycée et dans une moindre mesure au collège. Ces évolutions marquent-elles la fin d'un mouvement de balancier entre des conceptions divergentes de ce qu'il faut transmettre aux élèves, intégrant une fois pour toutes la science informatique comme un contenu fondamental ? Cet enseignement prendra-t-il la forme d'un curriculum cohérent entre les différents niveaux ayant pour objectif des conceptualisations tel que visé par le projet de recherche IE-CARE, ou des scissions vont-elle s'opérer entre différents niveaux d'enseignement, laissant place à un empilement plutôt qu'à une continuité ?

Ce sont les questions auxquelles je vais apporter des éléments de réponse dans cette première partie.

1. Numérique et informatique : représentations et définitions des concepts

1.1 Qu'est-ce que la science informatique ?

Pour comprendre ce qu'est l'informatique, et ce qui fait sa spécificité, je prends essentiellement appui sur la série de cours donnés par Gérard Berry, titulaire de la chaire *Algorithmes, machines et langages* au Collège de France et promoteur d'un enseignement de l'informatique dès l'école primaire. Intitulées « *Où va l'informatique ?* », ces interventions reprennent le contenu de son ouvrage **L'HYPERPUISANCE DE L'INFORMATIQUE. ALGORITHMES, DONNEES, MACHINES, RESEAUX** (Odile Jacob, 2017).

Pour Gérard Berry, l'informatique constitue une rupture dans l'histoire des découvertes scientifiques et techniques modernes. En effet, la pensée scientifique et technique, qui change en profondeur le monde à partir de la seconde moitié du 18e siècle et permet la révolution industrielle, s'articule autour de découvertes dans trois grands domaines : la matière, l'énergie et les ondes électromagnétiques. A la différence de ces découvertes, l'informatique, qui apparaît au milieu du 20e siècle, n'est pas issue de l'observation et de la compréhension de la nature, mais relève d'une construction, au sens où ses briques de base sont construites de toutes pièces par l'Homme. Surtout, elle s'intéresse à un domaine jusque-là peu abordé, mais qui a un rôle central chez l'Homme : l'information. L'informatique va permettre de la manipuler à de grandes échelles à partir de deux grands principes : sa modélisation sous une forme numérique et son traitement automatique par des algorithmes. Ces idées viennent briser les vieilles limites et donnent accès à des applications qui étaient inenvisageables avec les représentations du 19e siècle, dépendantes des caractéristiques de la matière.

Pour éclairer la rupture apportée par l'informatique, prenons un exemple concret : l'appareil photo. Dans un monde fonctionnant avec les connaissances des 18e et 19e siècles, il serait impossible de faire de bons appareils photo dans des Smartphones. Leurs optiques rudimentaires, sujettes à de fortes distorsions, constituaient un obstacle infranchissable. Dans le monde du 21e siècle, grâce à l'informatique, cela devient possible. Les défauts des images, engendrés par les faibles performances des capteurs, sont corrigés par des algorithmes. Cet exemple illustre la façon dont l'informatique vient briser les vieilles limites : limites matérielles, mais également limites de pensée. Bien plus qu'un outil (l'expression

⁴ Didactique et Apprentissage de l'Informatique à l'Ecole - <https://anr.fr/Projet-ANR-14-CE24-0012>

« outil informatique » encore très répandue reprend les représentations des siècles précédents), l'informatique est une science qui induit une nouvelle façon de penser les choses et dont les applications sont transversales à quasiment tous les domaines.

Cette transversalité prend son origine dans l'universalité des principes sur lesquels repose l'informatique.

- 1- La notion d'information est universelle à tous les domaines. Qu'il s'agisse de géographie, de sciences physiques, d'histoire ou de médecine, toutes les disciplines partagent la même notion de l'information ;
- 2- La notion d'algorithme est elle aussi universelle. Suite précise d'opérations effectuées sur des données en entrée qui fournit des données en sortie (les données en entrée pouvant être utilisées en entrée d'un autre traitement), l'algorithme permet une variété d'applications quasi infinie allant de la compression des images à la gestion du trafic routier en passant par la reconnaissance d'objets ;
- 3- La modélisation de l'information sous une forme numérique est la même pour tous les types de données : texte, image, audio, carte géographique ou séquence du génome sont codés suivant le même modèle ;
- 4- Les machines informatiques fonctionnent toutes sur les principes issus de la machine universelle de Turing : c'est en effet bien le même type de machine (exception faite de sa puissance) qui commande un lave-linge, un véhicule ou sert aux prévisions météorologiques. Autrement dit, la même machine sert pour tout à partir du moment où l'on a mis dedans le bon programme.

Il est important de comprendre que le développement exponentiel des applications de l'informatique repose sur ces quatre caractéristiques et que la diversité des tâches prises en charge par le traitement automatique de l'information constitue à la fois un véritable changement mental et un potentiel de développement considérable : il n'y a en effet pas de limite définie au traitement de l'information. Reproductible, conservable, traitable et communicable de façon quasi instantanée, l'information a des caractéristiques très différentes de la matière et de l'énergie. Seule la puissance de calcul, la capacité de stockage, la vitesse de transfert, tous trois en constante augmentation depuis soixante-dix ans, constituent des facteurs limitant. Si l'étendue des applications de l'informatique n'a cessé de s'étendre, touchant aujourd'hui presque tous les domaines, les potentialités restent vertigineuses.

Alors que les services relèvent d'une logique d'usage, qui se fait à travers des parcours et des interfaces choisis par les concepteurs et qui sont variables dans le temps, l'informatique en tant que science du traitement de l'information repose sur des invariants et des conceptualisations relatives à la modélisation de l'information et à son traitement automatique par des machines qui fonctionnent toutes sur un même principe.

Le domaine des connaissances aujourd'hui couvert par la science informatique est découpé en sous-domaines qui servent à structurer son enseignement. En France, la typologie qui fait autorité est celle proposée par Gilles Dowek, composée de quatre sous-domaines (Dowek, 2011) :

- L'information numérique : elle permet de représenter sous une forme unifiée des informations de nature et de domaine divers : textes, images, sons etc. ;
- Les algorithmes : ils spécifient de façon abstraite, sous la forme d'un enchaînement précis d'opérations, les traitements à effectuer sur l'information ;
- Les langages : ils permettent de traduire des algorithmes abstraits en programmes exécutables par les machines ;
- Les machines informatiques : elles permettent d'exécuter les programmes, de stocker les données et de gérer les communications. Elles comprennent les objets connectés et les réseaux.

1.2 Qu'est-ce ce que le numérique aujourd'hui ?

Le sens du terme « numérique » a évolué au cours des cinquante dernières années, s'enrichissant de nouvelles significations.

Etymologiquement, le terme « numérique » signifie : « qui concerne les nombres ».

Avec les progrès technologiques, son sens s'est enrichi pour désigner toute information (texte, image, vidéo...) modélisée sous la forme d'une suite de nombres et pouvant être traitée par un dispositif informatique ou électronique.

Avec la convergence technologique de l'informatique et des télécommunications, qui a donné naissance à Internet, le terme s'est à nouveau enrichi pour englober les réseaux et les contenus diffusés par ces réseaux.

A partir de 2005-2010, l'arrivée conjointe des terminaux intelligents sans fil et des technologies 2.0 qui ont fait du Web une gigantesque plateforme pour des services interactifs à l'échelle mondiale générant de gigantesques quantités de données sur le comportement des

utilisateurs, la problématique centrale du numérique est devenue celle des usages de services connectés. Aujourd’hui, le terme « numérique », évacuant la dimension technologique de ses débuts, fait essentiellement référence aux services et usages connectés.

1.3 Relation entre numérique et informatique

Il n'y a pas de rupture entre informatique et numérique, mais une continuité. Si le monde numérique, de plus en plus tourné vers une logique de l'usage, se concentre sur l'expérience utilisateur, ses services sont rendus possibles par l'informatique sous la forme d'empilements de programmes qui communiquent entre eux (du programme de bas niveau qui gère la mémoire physique de la machine jusqu'à celui qui gère l'interface) d'une manière transparente pour l'utilisateur. On pourrait dire, en grossissant le trait, que le numérique désigne aujourd'hui le versant visible de l'iceberg (les services), tandis que l'informatique désigne la partie invisible (les processus qui soutiennent le fonctionnement des services).

2. Numérique et informatique : repères historiques sur les conceptions de ce qu'il faut enseigner à l'école

2.1 Un mouvement de balancier entre deux conceptions

Pour aborder l'évolution des conceptions qui ont façonné l'approche de l'informatique dans l'enseignement au cours des cinquante dernières années, je m'appuie sur les données apportées par Georges-Louis Baron, principal spécialiste de la question, et en particulier sur ses propos les plus récents, exprimés lors de la conférence *L'informatique, discipline scolaire : enjeux de savoir et de pouvoir* donnée à l'université de Fribourg le 10 mai 2019.

La question de l'enseignement de l'informatique, qui remonte aux années 70, se caractérise par un mouvement de balancier entre deux conceptions de ce qu'il faut enseigner. D'un côté, une vision de l'informatique en tant qu'objet d'enseignement qui considère qu'il faut avant tout enseigner les concepts et méthodes de l'informatique. De l'autre, une vision d'une informatique-outil qui promeut la formation à l'usage des outils.

2 .1.1 Années 1970-1980 – Les premiers pas

Les années 70 constituent un période de gestation traversée par les questions de savoir si l'informatique doit être enseignée, quelle informatique enseigner, et à qui. Ces questions, qui ne trouvent pas de réponse stable, en partie parce que l'informatique évolue et produit à un rythme rapide de nouveaux outils et usages qu'il faut prendre en compte, se perpétuent encore aujourd'hui.

Au début des années 70, attentifs à ne pas réduire l'informatique à une technique, des chercheurs comme Jacques Arsac soutiennent l'enseignement d'une « démarche informatique » algorithmique et modélisatrice qu'ils estiment utile partout et tout le temps à la différence des outils qui sont obsolescents.

Cette vision s'impose dans les années 80 avec la mise en œuvre d'un enseignement de l'informatique comme objet, sans création d'une nouvelle discipline. Au lycée, les contenus d'enseignement relèvent d'une transposition didactique des enseignements de licence. A l'école, la démarche soutenue par Seymour Papert et son instrumentation par Logo (langage permettant de programmer un petit robot appelé Tortue Logo), destinée à développer chez les

enfants une manière de penser la résolution de problèmes, est mise en œuvre. En dépit de son adoption par les enseignants, de résultats intéressants sur le plan des apprentissages et d'un succès auprès des élèves, cette vision de l'enseignement de l'informatique comme objet va être supplantée. L'apparition de nouveaux outils (tableurs, traitements de textes) et leur pénétration dans le monde économique contribuent à renforcer progressivement la vision des promoteurs d'une informatique-outil qui pourrait se résumer de la façon suivante : la démarche informatique et la programmation sont pour les professionnels, les usages des outils informatiques sont pour tout le monde.

2.1.2 Années 1990-2000 – L'informatique comme outil : une logique d'usage et un régime de la certification sans curriculum

L'approche d'une informatique-outil qui a pour objectif de former à l'usage des outils regroupés sous l'expression « technologies de l'information et de la communication », s'impose au cours des années 1990 dans presque tous les pays. Intégrée de manière transversale à toutes les disciplines, elle consiste à apprendre à manipuler de façon pertinente des outils numériques logiciels et matériels pour rechercher, organiser, traiter, produire et communiquer de l'information. Cette approche, qui vise le développement de compétences instrumentales sans nécessité de conceptualisation débouche sur la mise en place d'un régime de certification des compétences numériques (B2i puis PIX) comme mode d'évaluation unique.

2.1.3 Années 2010 – Reconsidération de la question de l'informatique comme objet d'enseignement

Supprimé en 1998 (au moment de l'explosion d'Internet), l'enseignement de l'informatique en tant que science ne cesse pas pour autant d'être soutenu par un ensemble d'acteurs, principalement composé de chercheurs en informatique (majoritairement issus de l'INRIA) et d'enseignants du secondaire regroupés au sein de l'association EPI⁵. Selon eux, la littératie numérique actuelle, comprenant l'usage des TIC et l'EMI, ne suffit pas pour les nouvelles générations, pas plus que l'explosion des usages dans la sphère privée : le simple usage du numérique ne permet pas d'acquérir les connaissances et de cultiver les modes de pensée informatique qui lui sont spécifiques et qui seront nécessaires dans la société du 21e siècle. Ils

⁵ Enseignement Public et Informatique - <https://www.epi.asso.fr/>

considèrent l'abandon d'un enseignement de l'informatique comme une erreur et y voient une des raisons du décrochage de la France (et plus généralement de l'Europe) dans l'économie numérique ayant pour conséquence sa dépendance aux industries des Etats-Unis et de l'Asie. Promoteurs d'une large diffusion de la culture informatique, ces acteurs sont à l'origine d'initiatives importantes (*Class Code⁶, 1,2,3 Codez!⁷*) en faveur de l'éducation du grand public à l'informatique et de son enseignement à l'école, développées à la faveur des financements du PIA (Programme d'Investissements d'Avenir). Ils préconisent trois niveaux d'éducation :

- pour le citoyen ;
- pour ceux qui se destinent à un métier où l'informatique joue un rôle central ;
- pour les scientifiques et ingénieurs (déjà, nombre d'établissements de l'enseignement supérieur ont commencé à introduire un cursus d'informatique qu'ils considèrent désormais comme une discipline fondamentale pour tous leurs étudiants de première année).

2.1.4 Les facteurs à l'origine d'une reconsidération de la question de l'informatique à l'école

Le point de vue des promoteurs d'un enseignement de l'informatique en tant qu'objet et au-delà, en tant que nouvelle discipline, a reçu un accueil favorable à partir des années 2010. Le rapport de l'Académie des sciences de 2013⁸ qui préconise avec force un enseignement de l'informatique à tous les niveaux de l'école a joué comme un signal d'alerte. Il met en avant deux arguments pour une reconsidération de la question de l'enseignement de l'informatique en France :

- d'une part, le « phénomène numérique », caractérisé par la rapidité et l'ampleur de sa pénétration représente un secteur économique aujourd'hui majeur dans lequel la France n'est, à l'exception de sa recherche, qu'un acteur mineur. Selon le rapport de l'Académie des sciences précité, si la France veut jouer un rôle majeur dans l'économie numérique, elle doit reconsidérer ses positions sur l'enseignement de l'informatique : « L'Europe et la France en particulier accusent un important retard conceptuel et industriel dans le

⁶ <https://pixees.fr/classcode-v2/>

⁷ <https://www.fondation-lamap.org/fr/123codez>

⁸ L'enseignement de l'Informatique en France – Il est urgent de ne pas attendre (Institut de France, Académie des sciences, 2013)

domaine par rapport aux pays les plus dynamiques, comme les Etats-Unis et certains pays d'Asie. Ce retard est en partie lié aux carences de l'enseignement de l'informatique, resté longtemps au point mort ou réduit à l'apprentissage des seuls usages de produits de base. Un enseignement aussi limité ne saurait permettre de faire basculer notre pays de l'état de consommateur de ce qui est fait ailleurs à celui de créateur du monde de demain. » (Institut de France, Académie des sciences, 2013)

- Un autre argument important en faveur d'un enseignement de l'informatique dès l'école primaire est l'idée que l'informatique repose sur une forme de pensée spécifique qu'il est devenu nécessaire d'intégrer aux cursus scolaires du 21e siècle : la « pensée informatique » ou « pensée computationnelle ». Introduite en 2006 par Jeannette Wing, alors professeure d'informatique à l'université Carnegie Mellon, cette notion a fédéré un mouvement au niveau international qui a débouché sur la production de ressources pédagogiques directement reproductibles dans les classes (activités de l'informatique débranchée). Dans son article de 2006 intitulé « Computational Thinking », Wing présente la pensée informatique comme un savoir fondamental du 21e siècle, qui devrait faire partie des apprentissages de l'école : « La pensée informatique constitue pour nous tous un savoir fondamental, pas seulement pour les informaticiens. Au même titre que la lecture, l'écriture ou l'arithmétique, nous devrions la transmettre à nos enfants. » (Wing, J., 2006). Mais qu'est-ce précisément que la pensée informatique ? Selon Wing, c'est bien plus que l'usage d'un langage de programmation : « La programmation permet d'implémenter la pensée informatique, mais celle-ci ne s'y réduit pas. Penser comme un informaticien va bien au-delà de la capacité à programmer, cela requiert de penser à différents niveaux d'abstraction. » (Wing, J., 2006). Ce que Wing entend par « penser comme un informaticien » (ou « penser informatiquement ») consiste à déterminer si un problème, quel que soit son domaine, peut être traité par un ordinateur et le cas échéant, à le formuler de manière à ce qu'il puisse être traité par un dispositif informatique. Lorsque c'est le cas, « penser informatiquement » consiste à traduire la solution sous une forme qui soit « exécutable » par un ordinateur. Ce processus de « traduction » met en jeu un répertoire de cinq capacités cognitives (dont je fournis une description détaillée incluant des exemples en annexe) : (1) penser l'enchaînement séquentiel des actions pour résoudre un problème (pensée algorithmique), (2) sélectionner les informations pertinentes pour résoudre un problème (abstraction), (3) déterminer quelle solution, parmi plusieurs, est la

mieux adaptée pour traiter un problème (évaluation), (4) décomposer un problème complexe en sous-problèmes simples (décomposition), (5) inférer une solution à un problème général à partir de l'identification de régularités dans des instances ou d'éléments d'une solution existante à un problème analogue (généralisation).

Historiquement, la notion de pensée informatique proposée par Jeannette Wing apparaît comme l'héritière de la « démarche informatique » promue en France des années 70.

Dans son argumentaire pour un enseignement de l'informatique, le rapport de l'Académie des sciences fait explicitement référence à la notion de pensée informatique : « L'informatique dans les sciences devient bien plus qu'un outil de calcul. Elle conduit à une nouvelle forme de pensée, appelée "pensée informatique" (computational thinking en anglais). » (Institut de France, Académie des sciences, 2013).

L'accent mis explicitement sur le développement d'une pensée informatique (en tant qu'ensemble de capacités cognitives mises en œuvre par la démarche informatique) constitue un point important dans la façon de concevoir l'informatique et son enseignement. A l'encontre d'une représentation largement partagée, l'informatique ne se réduit pas au codage qui ne constitue que la phase finale de traduction dans un langage de programmation. Elle est avant tout une activité conceptuelle, qui peut en partie s'enseigner sans ordinateur (approche de l'informatique débranchée qui a pour objet de construire des concepts). Ainsi conçu, l'informatique quitte le statut d'outil pour ce qu'il est en réalité : une science ayant ses spécificités et nécessitant un apprentissage propre.

3. Etat actuel de l'enseignement de l'informatique en France et dans des pays voisins « pionniers »

3.1 Situation en France

3.1.1 Au lycée

L'enseignement au lycée, inspiré des cursus de L1 en informatique, s'organise autour de trois modalités (optionnel, tronc commun, enseignement de spécialité) qui visent toutes des conceptualisations relatives à la science informatique :

- Enseignement ICN (Informatique et Création Numérique) optionnel en classes de seconde, première et terminale qui établit une passerelle entre le numérique et l'informatique en tant que science du traitement automatisé de l'information ;
- Enseignement SNT (Sciences Numériques et Technologiques) dans le tronc commun de seconde qui vise l'acquisition de connaissances dans les domaines de la modélisation de l'information, de l'algorithme et des langages ;
- Enseignement ISN (Informatique et Sciences du Numérique) de spécialité en première et terminale issu d'une transposition du programme de L1 en informatique, destiné aux élèves visant une poursuite d'études dans le domaine scientifique.

La formation des enseignants, majoritairement recrutés au sein des mathématiques et des sciences physiques, met en jeu plusieurs dispositifs et s'appuie largement sur des acteurs de la recherche (INRIA) et de l'enseignement supérieur (départements d'informatique des universités).

Les données récentes concernant l'inscription des élèves aux enseignements de spécialité de premières montrent un démarrage mesuré de l'ISN. Comparé aux demandes des lycéens en direction des spécialités scientifiques traditionnelles (64% de demandes pour les *Mathématiques* et avoisinant les 43% pour la *Physique-Chimie* et les *Sciences de la Vie et de la Terre*), ou même des nouvelles spécialités *Histoire-Géographie, Géopolitique et Sciences Politiques* et *Humanités, Littérature et Philosophie* (choisies par respectivement 33,5% et 18,5 % des lycéens), le succès de la spécialité *Numérique et Sciences Informatiques*, retenue par seulement 8,4 % des lycéens en première à la rentrée 2019, apparaît modeste. (source :

Enseignements de spécialité au lycée : des parcours plus divers et plus adaptés aux profils et aux projets des lycéens⁹. L'ampleur que prendra l'enseignement de la science informatique, qui a fait l'objet d'avancées majeures avec notamment la création d'une discipline en 2019 et d'un CAPES spécifique en 2020, est difficile à prévoir. Pour l'instant, le nombre de postes ouverts à la première session du CAPES ISN est limité à une cinquantaine et seules quelques ESPE (aucune en Ile de France) proposent des préparations au CAPES ISN (avec le soutien des départements d'informatique des universités auxquelles elles sont rattachées).

3.1.2 Au collège

Au collège, l'informatique n'est pas une discipline à part entière. Les textes spécifient qu'« un enseignement de l'informatique (algorithmique et programmation) est dispensé conjointement en mathématiques et en technologie. Il n'a pas pour objectif de former des élèves experts, mais de leur apporter des clés de décryptage d'un monde numérique en évolution constante. Il permet d'acquérir des méthodes qui construisent la pensée algorithmique et développe des compétences dans la représentation de l'information et de son traitement, la résolution de problèmes, le contrôle des résultats. »

Ces objectifs d'apprentissage se situent dans une optique curriculaire de la science informatique visant une articulation collège-lycée. Le problème de la formation des enseignants de technologie et de mathématiques, amenés à enseigner l'informatique sans nécessairement disposer d'une culture informatique, reste une préoccupation.

3.1.3 A l'école primaire

A l'école, l'informatique a fait une entrée dans les programmes de 2016 sous la forme d'une sensibilisation à la programmation. Les textes préconisent explicitement la mise en œuvre d'activités de codage pour déplacer un robot ou un personnage à l'écran dans une approche intégrée aux autres apprentissages, en particulier en mathématiques (repérage et déplacements dans l'espace, construction de figures géométriques).

Les conceptualisations traditionnellement associées à la programmation (structure conditionnelle, itération, notion de variable...) ne figurent pas comme des objectifs de

⁹ Communiqué de presse de Jean-Michel Blanquer du 12/07/2019 :

<https://www.education.gouv.fr/cid143872/enseignements-de-specialite-au-lycee-des-parcours-plus-divers-et-plus-adaptes.html>

connaissance. Et les programmes ne font aucune référence explicite à la pensée informatique. Elle n'est pas pour autant absente des préoccupations. La page d'accueil d'*Eduscol* consacrée au numérique dans le premier degré¹⁰ propose un parcours *M@gistere (Premières activités de programmation pour comprendre le numérique)* élaboré par des acteurs extérieurs à l'Education Nationale (*Class Code* portée par l'INRIA et *D-Clics numérique*¹¹s portée par la Ligue de l'Enseignement)) qui vise à transmettre les fondements historiques de la pensée informatique en préambule à des activités d'introduction à la programmation.

D'une manière générale, l'offre de formation et de ressources pour entrer dans la science et la pensée informatique est essentiellement du côté d'acteurs situés hors du champ de l'Education Nationale : le dispositif *Class Code* (INRIA) propose des formations à distance sous forme de MOOC accessibles gratuitement (*Class Code* annonce « 50000 personnes touchées ») et l'ouvrage **1,2,3 CODEZ !** (Le Pommier, 2016) constitue une référence utilisée dans d'autres pays francophones. Il n'y a cependant pas de données qualitatives sur ce qu'apprennent et réinvestissent les enseignants qui font usage de ces dispositifs.

Il semble que l'on se trouve actuellement dans une situation de transition avec une interrogation sur ce qui doit être enseigné : si des conceptualisations en informatique ne sont pas exclues, elles doivent se construire à partir de situations-problèmes en relation avec d'autres apprentissages, et sont dépendantes des connaissances de chaque enseignant en matière d'informatique.

3.2 Situation dans des pays « pionniers » : cas de l'Angleterre et la Suisse francophone

3.2.1 Situation en Angleterre

Les programmes scolaires britanniques entrés en vigueur en 2014 intègrent un curriculum pour l'enseignement de la science informatique dans le primaire et le secondaire mis en œuvre dans tous les établissements.

L'introduction de cet enseignement initialement soutenu par des chercheurs en informatique, est le résultat d'une démarche systémique prenant en compte les questions de la création d'un

¹⁰ <https://eduscol.education.fr/pid29714/le-numerique-dans-le-premier-degre.html>

¹¹ <https://d-clicsnumeriques.org/>

curriculum, de la formation des enseignants, du déploiement de matériel à grande échelle et de la mise à disposition de ressources pédagogiques, menée sur un temps court :

- 2008 : création de *Computing at School* par des chercheurs en informatique ;
- 2009 : rapport de la *Royal Society* qui préconise l'apprentissage de la science informatique à l'école au détriment de la littératie ;
- 2012 : coordination par la *British Computing Society*, sur demande du Ministère de l'Education britannique, de la création d'un curriculum en science informatique et d'un réseau d'enseignants formés ;
- 2014 : mise en œuvre dans tous les établissements d'enseignement anglais ;
- 2015 : distribution d'un million d'ordinateurs *Micro:Bit*¹² (ordinateur à carte unique) sur une initiative portée par la BBC.

3.2.2 Situation en Suisse

Depuis la rentrée 2018, la Suisse francophone met en œuvre un enseignement de la science informatique basé sur un curriculum spécifique dans le cadre d'une littératie numérique élargie intégrant les usages des technologies de l'information et de la communication et l'éducation aux médias. Un plan de formation progressif des enseignants est engagé et une politique d'équipement harmonisée est en cours de structuration¹³.

Là encore, l'inscription dans les programmes est l'aboutissement d'une action forte des chercheurs en informatique, en particulier de l'université de Lausanne à l'origine du projet Thymio, un robot développé exclusivement pour l'éducation associé à un site proposant des ressources pédagogiques et de l'auto-formation. On retrouve également la volonté de travailler de façon conjointe les axes des contenus à enseigner, de la formation des enseignants, de l'équipement et des scénarios pédagogiques instrumentés.

¹² <https://microbit.org>

¹³

https://www.vd.ch/fileadmin/user_upload/accueil/Communique_presse/documents/20180821_presentation_CP_r_entree_01.pdf

4. Quelles perspectives pour un enseignement de l'informatique en France dans une optique curriculaire ?

4.1 Un enseignement qui se structure, mais la possibilité d'une scission entre le primaire et secondaire

Le mouvement de balancier entre informatique-outil et informatique comme objet d'enseignement, qui se perpétue en France depuis les années 70, semble sur le point de cesser. Les données présentées montrent que l'enseignement se structure dans le secondaire dans une approche top-down inspirée de l'enseignement supérieur avec la création d'une discipline au lycée, associée à la mise en place de formations pour les enseignants, et qu'un curriculum cohérent est envisageable avec le collège. Revenir en arrière semble difficilement envisageable dans un contexte de mise en œuvre par les pays voisins et du fait des enjeux économiques présents et futurs liés aux applications de l'informatique.

A l'école primaire, où la question des contenus à enseigner reste posée, plusieurs options sont possibles. La mise en place d'un curriculum en informatique cohérent sur l'ensemble de la scolarité obligatoire n'est pas acquise : une scission pourrait avoir lieu entre les visions de l'informatique dans le primaire et dans le secondaire. Un élément important pour une continuité entre les niveaux d'enseignement réside dans l'accueil fait à la notion de pensée informatique (au sens défini par Jeannette Wing), porteuse d'une vision unificatrice de l'enseignement de l'informatique. Cette notion reste toutefois sujette à discussion en France. La capacité à l'appréhender et à la mettre en œuvre tout au long du cursus scolaire n'est là non plus pas acquise.

4.2 L'importance des ressources pédagogiques pour amorcer le mouvement

Les regards portés sur l'Angleterre et la Suisse montrent le rôle important joué par des initiatives nées hors du cadre de l'éducation officielle. En Angleterre à travers le dispositif *Computing at School*, en Suisse avec le projet *Thymio*. Dans ces deux cas, la mise à disposition de scénarios pédagogiques reproductibles par les enseignants a constitué un préalable important pour un passage à l'échelle.

En Suisse, le cas de Thymio, introduit dans les classes dès 2012 avant l'entrée de la science informatique dans les programmes, est intéressant. Des études menées auprès d'enseignants précurseurs en matière d'enseignement de la robotique ont permis de cerner les principaux obstacles à sa mise en œuvre. Le travail de (Kradolfer et al, 2014) met en avant trois aspects, cités ici par croissant d'importance :

- les difficultés rencontrées pour disposer de l'équipement ;
- l'absence d'un cadre institutionnel conférant une légitimité à un tel enseignement ;
- et en premier lieu, l'investissement particulièrement important demandé aux enseignants, du fait d'un manque de formation (initiale et continue) et de l'absence d'un répertoire d'activités pédagogiques scénarisées à proposer aux élèves.

En France, le projet ANR DALIE, centré sur les apprentissages possibles à l'école primaire et sur le cadre d'accompagnement des enseignants, a également mis en évidence l'importance de disposer de scénarios pédagogiques. Les études menées ont montré que les élèves étaient capables de construire ou de déboguer des programmes dès la GS et de mobiliser des concepts complexes (comme les « messages » dans *Scratch*) à partir de la fin du cycle 2. Toutefois, ces apprentissages restent conditionnés à un guidage par des scénarios pédagogiques adaptés se déroulant sur un temps long.

Elaborer et tester un modèle de scénarisation pédagogique permettant de générer un répertoire de scénarios à mettre à la disposition des enseignants a constitué le travail d'ingénierie pédagogique présenté dans la seconde partie de ce mémoire.

Partie 2 - Elaboration d'un modèle de scénarisation pédagogique pour l'enseignement de l'informatique à l'école élémentaire dans une optique curriculaire

Cette seconde partie, centrée sur le travail mené au sein du projet de recherche IE-CARE, est l'objet d'une présentation en deux grandes sous-parties.

La première présente les axes de recherche du projet et leurs objectifs en décrivant plus particulièrement l'axe 2 dans lequel prend place mon travail. Ce travail relève du domaine de l'ingénierie pédagogique. Elle consiste en l'élaboration d'un modèle de scénarisation pédagogique servant de cadre pour la production (à venir) de scénarios pédagogiques ayant une double vocation :

- initier un répertoire de scénarios pédagogiques instrumentés mis à la disposition des enseignants pour soutenir un enseignement/apprentissage visant l'appropriation de concepts en informatique par les élèves ;
- alimenter le travail de définition d'un curriculum pour la science informatique à l'école.

La seconde sous-partie décrit les grandes étapes du travail mené et les réflexions qu'elles ont impliquées :

- les questions initiales de fonds émergeant au lancement du projet et les réponses apportées qui constituent des hypothèses de travail pour l'élaboration d'un premier scénario test ;
- la démarche d'élaboration d'un scénario test et son expérimentation auprès d'enseignants avec l'objectif central d'évaluer le degré d'effort nécessaire à son appropriation ;
- la démarche d'élaboration d'un modèle de scénarisation permettant la production d'autres scénarios.

En amont de la présentation de mon travail, il me faut signaler deux points qui ont eu un impact important sur la nature du travail de conception. Ils concernent une reconfiguration des conditions initialement prévues d'élaboration des scénarios :

- Le projet initial prévoit la production des scénarios tests dans le cadre d'ateliers de co-conception avec des enseignants repérés : les scénarios doivent être produits par des enseignants accompagnés par les chercheurs. Cette modalité n'est finalement pas mise en œuvre lors de cette première phase du projet qui correspond à la période de mon stage. Les enseignants de terrain ne seront associés au projet qu'à partir de la rentrée 2019. Ce changement a profondément reconfiguré la nature du travail, impliquant de définir de nouvelles modalités d'élaboration du scénario test et d'évaluation de son appropriation par les enseignants ;
- Le projet initial prévoit également la construction d'un environnement informatique (plateforme collaborative) pour la mutualisation des scénarios produits et devant permettre les échanges entre chercheurs et enseignants de terrain associés au projet. La création de cet environnement a été abandonnée au lancement du projet.

Ces changements ont profondément modifié l'approche de la conception : la démarche de co-conception centrée-utilisateur prévue faisant place à une démarche de « laboratoire » qui sépare les phases de conception et d'expérimentation.

L'absence d'une démarche de co-conception a également impacté le contenu des scénarios. Le développement de « scénarios intégrés à la vie de la classe » qui apparaît comme une approche bien adaptée à l'école primaire, repose en effet sur la possibilité de faire émerger des projets à partir du vécu d'une classe. Il n'a donc pu avoir lieu.

Ces contraintes ont amené à une redéfinition du travail initialement envisagé.

1. Présentation du projet de recherche IE-CARE et questions initiales

1.1 Présentation du projet de recherche ANR IE-CARE

Sur la base du texte déposé à l'ANR, le projet IE-CARE a pour objectif « de proposer des conditions et des modalités durables pour un enseignement de l'informatique à l'école obligatoire en favorisant un point de vue curriculaire centré sur la question de savoir comment articuler différents contenus de manière progressive tout au long des trois premiers cycles scolaires pour constituer un corpus enseignable cohérent avec les attentes des différents niveaux concernés. »¹⁴ Au-delà du cadre de la classe, le projet a pour objet de contribuer au développement d'une culture informatique et technologique partagée par les enseignants, les accompagnateurs et les superviseurs de l'action pédagogique.

La recherche, qui se déroule sur une période allant de janvier 2019 à décembre 2021, se structure suivant trois axes thématiques reliés par un axe transversal :

- L'axe 1 a pour objectif de délimiter un ensemble de contenus enseignables en informatique dans une optique curriculaire sur les trois premiers cycles de l'école ;
- L'axe 2, corrélé à l'axe 1, vise la conception, la modification et l'étude de l'usage de scénarios pédagogiques et de ressources pour soutenir les pratiques d'enseignement et d'apprentissage de l'informatique à l'école ;
- L'axe 3 est centré sur la mise en place d'un cadre d'accompagnement pour les enseignants et les formateurs en informatique ;
- Synthèse des axes 1, 2 et 3, un axe transversal vise à circonscrire, dans une vision curriculaire, ce que pourraient être les contenus d'une informatique enseignable jusqu'à la fin du cycle 3 de l'école et à proposer des environnements utilisables en prenant en compte les obstacles techniques et cognitifs qu'ils pourraient représenter pour les élèves. Responsable de cet axe, Eric Bruillard insiste sur le fait que le projet IE-CARE « a vocation à produire de l'enseignable » sous la forme de scénarios instrumentés mis à disposition des enseignants.

¹⁴ <https://anr.fr/Projet-ANR-18-CE38-0008>

Entre outre, le document déposé à l'ANR précise le champ couvert par l'enseignement de l'informatique :

« Dans le cadre du projet IE-CARE, nous considérons que l'informatique englobe des compétences générales non spécifiques ou restreintes à la programmation, telles que la résolution de problèmes par itération et/ou par essai/erreur, la décomposition de problèmes complexes en problèmes plus simples, la classification et le tri d'objets, l'exploration et la comparaison entre différentes stratégies plus ou moins optimales, la recherche de solutions et l'application d'heuristiques, le traitement et le codage d'informations, le débogage, la compréhension des systèmes distribués (réseaux informatiques), etc. L'enseignement de l'informatique doit permettre l'introduction de ces concepts auprès des jeunes élèves. La tâche transversale a pour objectif de définir un référentiel de ces compétences. »

La conception de l'enseignement de l'informatique initialement portée par IE-CARE s'inscrit donc dans une vision large qui dépasse la programmation, intégrant des conceptualisations propres à la science informatique (modélisation et traitement de l'information) et les compétences plus générales couvertes par la pensée informatique (décomposition d'un problème en sous-problèmes, évaluation d'une solution...). Elle est en phase avec ce qui a été mis en œuvre dans d'autres pays (Angleterre et Suisse notamment) et les projets relayés au niveau international (*CS Unplugged*).

1.2 Questions initiales de fonds soulevées dans le cadre du projet IE-CARE et réponses apportées

1.2.1 Quelle approche de l'informatique ?

Si l'objectif d'un enseignement ayant pour objet la construction de conceptualisations en informatique est partagé parmi les membres du projet, deux conceptions se distinguent concernant les démarches à privilégier et les conceptualisations visées :

Une première vision, qui relève d'une culture de l'enseignement de la technologie, propose « d'ouvrir la boîte noire de l'informatique » en interrogeant les dispositifs technologiques, comme la carte RFID ou le moteur de recherche Google, pour comprendre ce qui se passe à l'intérieur. Il s'agit, suivant les mots d'Eric Bruillard, de « ralentir les technologies dans le cadre de l'éducation pour pouvoir les observer ». Une piste pédagogique propose d'interroger le fonctionnement de Google à partir de l'analyse de ses résultats. Pour ce qui me concerne,

cette approche soulève plusieurs questions : est-ce motivant pour les élèves ? Est-ce en phase avec la culture des enseignants du primaire ? Et jusqu'à quel point cela est-il possible dans la pratique ? En effet, peut-on vraiment savoir comment Google fonctionne à partir de l'analyse de ses résultats ? Ne risque-t-on pas de construire des représentations qui soient du domaine de « l'à peu près » ?

L'idée « d'ouvrir la boîte noire de l'informatique » est néanmoins porteuse de sens. Les progrès réalisés par l'informatique ont conduit à une forme de « naturalisation » des objets et des services numériques. Le Smartphone, l'ordinateur et l'accès instantané à des services connectés sont en effet devenus « des choses naturelles » pour les plus jeunes, induisant une absence de distance (y compris temporelle) qui masque la réalité d'un traitement programmé sous-jacent et empêche la prise de recul. Cependant, à quelques exceptions près, cette approche semble difficilement applicable eu égard à la complexité des dispositifs informatiques et à leur rapide évolution.

Aussi, pour se rendre compte de ce qui se passe, plutôt que de décortiquer de l'extérieur, ne vaudrait-il pas mieux programmer soi-même ? Cette autre approche, basée sur la créativité avec l'informatique dans la lignée des travaux de Marina Bers (Bers, M. 2019) et qui s'incarne dans une pédagogie de projets, est soutenue par certains membres du projet. Permettant d'aborder l'informatique en fonction de sensibilités différentes, laissant une place à l'expression personnelle, et proposant un objectif de production explicite et socialement visible, la pédagogie de projet présente l'avantage d'être intrinsèquement plus motivante. Elle semble également mieux adaptée au contexte puisqu'elle permet d'adapter les projets en fonction de l'instrumentation disponible. Elle comporte cependant le un risque de passer à côté des conceptualisations au profit d'autres attracteurs. Lors des journées EIAH 2019, des participants mettant en œuvre une pédagogie de projet avec leurs élèves témoignent de limites concernant l'évaluation des primaire : ce qui est testé, c'est bien souvent la compétence à mener à bien des projets, mais pas les connaissances et compétences en informatique.

Se lancer dans un projet de création avec l'informatique, qui implique le recours à un langage de programmation simple et l'élaboration d'algorithmes, présente un aspect insécurisant pour beaucoup d'enseignants. Au-delà de la complexification de la gestion de classe induite par l'instrumentation et du risque de déborder sur les plages dévolues aux autres apprentissages, il y a l'anticipation de ne pas pouvoir guider les élèves en raison d'une maîtrise insuffisante des concepts mis en jeu et d'un manque de technicité.

Dans la pratique, nombre de projets de ce type sont encore réalisés dans un cadre prévoyant un accompagnement qui prend la forme de moments de formation ou de regroupements, ou du soutien d'un « pair-expert ». C'est le cas des défis robotiques comme *Robots d'Evian* ((Lazzaroto, 2019). Pour les projets les plus ambitieux, à l'instar de ceux réalisés dans le cadre du dispositif des *Savanturiers*¹⁵, les enseignants bénéficient de formation et de l'accompagnement d'un « expert » (universitaire, professionnel, élève d'école d'ingénieur) qui vient parer aux difficultés conceptuelles et techniques.

Il n'en demeure pas moins que « créer quelque chose » est une dimension constitutive de l'enseignement de l'informatique. Cette approche est d'ailleurs reprise dans les programmes de la spécialité ISN au lycée qui mentionnent qu'au moins 25% du temps doit être consacré à l'élaboration de projets par les élèves. L'option ICN est, quant à elle, largement fondée sur la démarche de projets.

Les échanges au sein d'IE-CARE n'ont débouché sur aucun arbitrage clair concernant la démarche (technologique ou informatique) à privilégier en vue de l'élaboration d'un scénario test. Je privilégie pour ma part une approche qui mêle les deux démarches : « faire quelque chose avec l'informatique » (qui se traduit concrètement par l'élaboration d'algorithmes et leur programmation en vue de résoudre un défi, de créer un jeu...) et « construire des conceptualisations en informatique » au cours d'activités d'observation de dispositifs en action (robot, programme, etc.). Je prévois que le scénario test alternera ces deux types d'activités, le réinvestissement et l'évaluation des concepts se faisant au travers d'activités-défis. Les objets connectés (qui offrent la possibilité de scénarios pédagogiques intégrés à la vie de la classe) et la robotique constituent les entrées privilégiées pour mettre en œuvre cette approche.

1.2.2 La question de l'instrumentation

La question de l'instrumentation constitue un des aspects de l'axe 2 du projet IE-CARE qui a pour mission d'émettre des recommandations pour la constitution de « valises numériques et technologiques » permettant la mise en œuvre des scénarios. Le document déposé à l'ANR précise ce qui est visé : « Identifier les ressources techniques et autres qui sont les objets manipulés au sein des activités. Il peut s'agir d'objets tangibles techniques tels que des robots

¹⁵ <https://les-savanturiers.cri-paris.org/a-propos/presentation/>

et des objets connectés, d'objets tangibles non techniques (de nombreux objets de la vie quotidienne peuvent servir à travailler des concepts comme l'algorithme, cet aspect est désigné par informatique débranchée) et d'objets non tangibles tels que des environnements de programmation dédiés aux jeunes élèves. »

A ce stade, les échanges au sein du projet ont peu abordé les questions d'instrumentation. Les objets connectés font l'objet d'un intérêt dans le cadre de scénarios intégrés à la vie de la classe. Cette piste se heurte pour l'instant à deux obstacles :

- la complexité de prise en main : des kits Arduino permettant de réaliser des objets connectés ont été testés dans le cadre d'un atelier interne au LIP6. Si leur utilisation est envisageable, elle se révèle complexe. Ce constat est partagé par des enseignants expérimentés auxquels les kits ont été soumis.
- la nécessité d'un terrain d'expérimentation : l'instrumentation par des objets connectés dans le cadre de scénarios intégrés à la vie de la classe requiert un terrain d'expérimentation qui n'est pas disponible à ce stade du projet.

Sur le terrain, la question de l'équipement est de première importance. Le travail de (Kradolfer et al, 2014) mené en Suisse et déjà cité met en avant la difficulté des enseignants à disposer de l'équipement souhaité comme un obstacle majeur pour mettre en œuvre des projets en informatique. En France, où l'homogénéisation des situations en matière d'équipement numérique est loin d'être atteinte, disposer de l'équipement permettant un apprentissage instrumenté de l'informatique en classe constitue un obstacle pour nombre d'enseignants, particulièrement dans les domaines de la robotique.

Le récent rapport de la Cour des Comptes¹⁶ (Cour des Comptes, 2019) met en évidence un manque de stratégie dans la politique d'équipement (et de formation) numérique. L'initiative laissée aux collectivités locales, sans véritable pilotage du ministère de l'Education Nationale, débouche sur un paysage hétérogène. Ce constat est en phase avec une réalité vécue. Alors que certaines écoles parisiennes sont équipées de plusieurs dizaines de robots et de tablettes dans le cadre des budgets participatifs de la Ville de Paris, de l'autre côté du périphérique, certaines ne disposent d'aucun équipement récent.

Du côté de l'offre, on peut noter que certains instruments éducatifs se sont imposés comme des standards. Dans le domaine de la robotique éducative, Blue Bot en maternelle, Thymio et

¹⁶ Le service public numérique pour l'éducation : un concept sans stratégie, un déploiement inachevé

Ozobot à l'école élémentaire sont les matériels les plus répandus. Du côté des langages de programmation, *Scratch* et *Blockly* (langages dit par blocs en raison de leur interface qui permet de construire des programmes par l'assemblage de « blocs-instructions ») constituent eux aussi des standards pour l'entrée dans le codage. Ces langages (ou leur dérivés basés sur le même type d'interface) permettent également de piloter des robots comme Thymio et Ozobot. Cette relative standardisation de l'offre facilite la construction de scénarios instrumentés, qui sont par définition dépendants de l'instrumentation.

Du côté des objets connectés, la situation est moins avancée. Nous n'avons identifié qu'un seul matériel adapté à des élèves du cycle 3 pour un coût financier raisonnable : le kit *Micro:Bit*. Largement diffusé dans les écoles anglaises, il est encore peu distribué en France. Nous n'avons pas pu nous le procurer et le tester sur la durée du stage.

Prenant en compte ces différents paramètres, je fais le choix d'une instrumentation qui puisse trouver un écho dans les classes sur la base de plusieurs critères :

- l'instrument permet de mettre en œuvre des situations pédagogiques intéressantes permettant l'appropriation de premières conceptualisations en informatique ;
- il est (relativement) répandu ;
- son coût d'acquisition ne constitue pas un obstacle infranchissable ;
- sa mise en œuvre nécessite une logistique légère (transport aisé d'une classe à une autre, temps de recharge court...) ;
- il peut être manipulé par des enfants ;
- sa programmation se fait à travers une interface adaptée aux plus jeunes.

Ces critères ont orienté mon choix vers le robot *Ozobot*¹⁷.

1.2.3 L'orchestration de la classe en contexte instrumenté par le numérique

En lien avec l'instrumentation, la question de l'orchestration de la classe est abordée par des membres du projet qui la considèrent comme l'obstacle majeur à la continuité dans la pratique. Introduit par Pierre Dillenbourg (Dillenbourg, 2013), l'orchestration a trait à la façon

¹⁷ <https://ozobot.com>

dont les enseignants gèrent, en temps réel l'activité pédagogique dans un contexte numérique qui induit à la fois des potentialités et des contraintes nouvelles. Les travaux menés par Pierre Dillenbourg dans le domaine des EIAH visent à prendre en compte, à côté des aspects didactiques, les contraintes très pratiques du fonctionnement de la classe dès la conception des activités avec un objectif double : alléger la charge de l'enseignant (de manière à libérer son attention et à favoriser sa capacité à intervenir de manière adaptée auprès des élèves) et optimiser les potentialités d'exploitation pédagogique de l'instrumentation numérique.

Organiser l'espace classe, s'assurer que chaque élève a avec lui les bons instruments pour travailler, gérer le temps, etc., constituent en effet une part importante du travail des enseignants et de leurs préoccupations et influencent la qualité des apprentissages. Cela est encore plus vrai lorsque les apprentissages sont instrumentés par des dispositifs informatiques qui n'ont, pour la plupart d'entre eux, pas été développés en cohérence avec les contraintes qui sont celles des enseignants et dont la mise en œuvre comporte des risques de complexification de l'activité (pour une plus-value souvent faible). L'orchestration se préoccupe donc de répondre aux contraintes considérées comme de bas niveau (aspects logistiques, ergonomiques, organisationnels) et rejoint, en cela, les préoccupations actuelles concernant la réorganisation des espaces classe ou la création de Learning Labs.

Pour ce qui me concerne, si une part de l'orchestration peut être prise en charge dès le stade de la conception des scénarios, certains aspects relevant des savoir-faire des enseignants ne peuvent être capitalisés et intégrés qu'au moment de l'itération des scénarios en situation réelle. A ce stade, la forme de cette capitalisation reste indéterminée.

1.2.4 Un questionnement sur la notion de pensée informatique

Comme précédemment mentionné, la pensée informatique joue un rôle indéniable dans le mouvement de légitimation d'un enseignement de l'informatique à l'école. Au niveau international, la notion est abondamment discutée : le blog *Computational thinking papers*¹⁸ qui recense les publications sur le sujet, fait état d'environ deux cents articles sur l'année 2018. En France où les publications qui y sont consacrées sont rares et où la notion ne fait l'objet d'aucune synthèse de référence, la notion de pensée informatique ne fait l'objet d'aucun consensus, suscitant même un certain scepticisme. Les échanges au sein du projet IE-

¹⁸ <https://csedresearch.wordpress.com/computational-thinking/>

CARE reflètent cette position, certains membres (en l'occurrence informaticiens) mettant en doute qu'une telle pensée informatique existe. Serait-elle seulement un leurre pour attraper ceux qui sont indécis quant à la « nécessité » de mettre en œuvre un enseignement de l'informatique à l'école ?

Il faut ici souligner que l'idée d'une pensée informatique n'est pas nouvelle. Elle est née dans les années 60 :

« C'est vers 1965 qu'a commencé à émerger l'idée que l'informatique était plus qu'un phénomène technique, et qu'elle reposait sur une forme de pensée originale qui lui appartenait en propre. Mais c'était alors une intuition plus qu'une évidence tirée des faits. » (Arsac, 1981)

Puis elle s'est affirmée au début des années 80 servant déjà d'argument en faveur d'un enseignement de l'informatique :

« L'informatique s'appuie sur des méthodes de pensée originales dont l'apport est enrichissant et la valeur culturelle certaine » (Arsac, 1981).

Son instrumentalisation en tant qu'argument ne signifie pas pour autant qu'elle ne recouvre aucune réalité. Pour donner corps à cette notion sujette à caution, des chercheurs mènent un travail d'explicitation depuis plusieurs années. En France, Pierre Tchounikine, enseignant-chercheur en informatique à l'université Grenoble-Alpes, la décrit comme un ensemble singulier de compétences applicables à la résolution de problèmes : « Savoir décomposer un problème en sous-problèmes plus simples ; savoir réfléchir aux tâches à accomplir pour résoudre un problème en termes d'étapes et d'actions (ce que l'on appelle un algorithme) ; savoir décrire les problèmes et les solutions à différents niveaux d'abstraction, ce qui permet d'identifier des similitudes entre problèmes et, par suite, de pouvoir réutiliser des éléments de solutions. » (Tchounikine, 2017)). Le projet néo-zélandais *CS Unplugged*¹⁹ qui constitue une référence (la publication proposant des activités débranchées reproductibles avec les élèves est traduite en Français sous la direction de l'INRIA²⁰) va plus loin. Les auteurs y présentent la pensée informatique comme la combinaison de cinq compétences cognitives de haut niveau qu'ils décrivent de façon détaillée en illustrant leur propos d'exemples (pour ceux que la langue anglaise décourage, je propose une adaptation française de la description des cinq compétences en annexe 1).

¹⁹ <https://csunplugged.org/en/>

²⁰ <https://interstices.info/wp-content/uploads/2018/01/csunplugged2014-fr.pdf>

Pour ma part, je pense qu'aborder la pensée informatique comme une combinaison de compétences déconnectée de toute exécution computationnelle, en laissant penser que l'on peut la travailler pour elle-même et que ses compétences sont transférables (alors que cette transférabilité n'est pas prouvée) pose problème. J'y vois le risque d'une démarche pédagogique descendante non fondée consistant à partir des compétences de la pensée informatique et à construire en fonction de leur description les exercices pour la mettre en œuvre. Dans le type d'activités qui en résulterait, que pourrait-on évaluer ? L'étude de (Drot-Delange et Tort, 2018) montre que ces compétences, de par leur caractère général, sont difficiles à observer en situation : les enseignants ne voient pas tous la même chose.

J'aurais pour ma part tendance à privilégier une approche ascendante qui consiste à construire les situations pédagogiques sans se préoccuper de la pensée informatique et à voir si et comment elle intervient.

Considérant, à la différence de certains membres du projet IE-CARE, qu'une forme de pensée informatique existe, je pense qu'elle se développe de façon privilégiée chez les élèves de l'école par la confrontation à la machine informatique et à ses modèles d'exécution. L'instrumentation (environnements et objets traitant de l'information de manière automatique) joue à cet égard un rôle important.

2. Elaboration et expérimentation d'un scénario-test

2.1 Postulats de départ

Il existe une tension entre ce que peuvent s'approprier des enseignants qui n'ont pas (ou peu) de culture informatique et l'idée d'un curriculum comprenant des conceptualisations dans le domaine. Les études menées sur le terrain (Yadav, Mayfield, Zhou, Hambrusch, & Korb, 2014) montrent en effet que l'enseignement de l'informatique à l'école primaire représente un défi difficile à relever par les enseignants.

Le projet ANR IE-CARE postule qu'il faut partir de l'état des enseignants. Les ateliers de co-conception ne pouvant avoir lieu, j'ai retenu la solution de partir des séquences pédagogiques diffusées ou partagées sur le Web. On en trouve sur les sites institutionnels (DANE, Réseau Canopé notamment), les sites personnels d'enseignants et sur ceux des constructeurs/diffuseurs de matériels et logiciels.

Les sites institutionnels proposent des progressions pédagogiques sous forme de séquences avec des objectifs d'apprentissage explicites et des témoignages d'enseignants qui relatent leurs projets pédagogiques. Les enseignants partagent des activités qu'ils ont réalisées avec leurs élèves, souvent accompagnées de traces d'apprentissage. Les sites des concepteurs de robots (*Lego, Ozobot, Thymio, BlueBot...*) proposent des ressources visant explicitement des apprentissages en informatique. Elles ont souvent la forme d'activités organisées par difficulté qui, sans être articulées en scénarios pédagogiques, sont reliées à un curriculum en informatique²¹. Parmi ce qui est proposé, certaines activités sont récurrentes et mettent en jeu des conceptualisations informatiques sans les expliciter.

Dans l'objectif de construire un scénario qui vise des conceptualisations en informatique et que les enseignants puissent s'approprier, je fais l'hypothèse qu'il faut partir de ces activités repérées. Il existe en effet de bonnes chances que d'autres enseignants puissent se les approprier.

Ma démarche peut se résumer de la manière suivante :

- Identifier et expliciter les conceptualisations présentes dans les activités repérées ;

- Combiner ces activités (éventuellement en les adaptant) avec d'autres activités (éventuellement nouvelles) pour construire un scénario proposant une progression dans les connaissances.

La construction du scénario respecte le principe suivant :

- Une activité doit comporter au maximum un objectif de connaissance.

Ces quelques éléments constituent la base pour l'élaboration du scénario-test expérimenté auprès de deux groupes d'enseignants.

2.2 Présentation du scénario test

Le scénario-test, intitulé *Alphabet de l'informatique avec Ozobot* (cf. annexe 2) vise l'appropriation de connaissances dans le domaine de l'informatique et le travail de compétences associées à la pensée informatique (abstraction, pensée algorithmique, décomposition, évaluation).

Il s'appuie sur l'usage d'un petit robot nommé *Ozobot* que les élèves observent (construction d'une représentation de son fonctionnement interne) et programment (résolution des défis).

Ce scénario peut être mené dès le début du cycle 3.

Les connaissances en informatique qu'il vise sont les suivantes :

- Capteurs, actionneurs, contrôleur/processeur (leurs rôles) ;
- Mémoire (morte/vive) ;
- Principe du codage, forme et principe d'un langage de codage/programmation ;
- Notion d'instruction, transfert d'une instruction en mémoire, exécution d'une instruction ;
- Exécution de plusieurs instructions en parallèle ;
- Notion de variable (compteur).

²¹ <https://code.org>

2.3 Expérimentation du scénario-test

Démarche : 2 ateliers d'expérimentation d'une partie d'un scénario test

- Atelier 1 : 8 enseignants expérimentés (maternelle/élémentaire) n'ayant jamais mené d'activités d'apprentissage de l'informatique ;
- Atelier 2 : 22 enseignants stagiaires (M2-MEEF - Mention premier degré).

Méthode

- Test du scénario par les participants et questionnaires ;
- Scénario testé : “Alphabet de l'informatique” (Entrer dans l'apprentissage de l'informatique avec un robot éducatif : Ozobot).

Objectif central de l'expérimentation

- Explorer la capacité des enseignants à s'approprier le scénario test.

Atelier 1

Durée : 2 heures (avril 2019) ;

Participants : 8 enseignants expérimentés (maternelle/élémentaire) n'ayant jamais mené d'activité d'apprentissage de l'informatique ;

Déroulement :

- Temps 1 : présentation de ce que recouvrent les 4 domaines de l'informatique selon la typologie de G. Dowek + questionnaire sur le degré de connaissance estimé de chacun de ces domaines ;
- Temps 2 : les enseignants mènent les quatre premières activités du scénario ;
- Temps 3 : questionnaires individuels sur (1) la motivation et le degré d'effort estimé pour mettre en œuvre ces activités et (2) ce qui pourrait diminuer cet effort (formation/scénario pédagogique/autre ?).

Atelier 2

Durée : 2 heures (mai 2019) ;

Participants : 22 enseignants stagiaires (M2-MEEF - Mention premier degré) ;

Déroulement

- Temps 1 : présentation de ce que recouvrent les 4 domaines de l'informatique selon la typologie de G. Dowek ;
- Temps 2 : enseignants répartis en 5 groupes pour mener les 4 premières activités du scénario “Alphabet de l'informatique” - 1 robot/groupe - 1 enseignant/groupe découvre le scénario pendant 15 minutes avant le faire mener aux autres membres du groupe ;
- Temps 3 : les groupes mènent le scénario avec possibilité de poser des questions au concepteur ;
- Temps 4 : questionnaires individuels sur la connaissance des enseignants dans les 4 domaines de l'informatique + motivation et degré d'effort estimé pour s'approprier et mettre en œuvre le scénario en classe + intérêt de disposer de scénarios pédagogiques en général et pour l'informatique en particulier + caractéristiques d'un “scénario idéal”.

Bilan synthétique des ateliers

Les enseignants :

- Disent réutiliser des scénarios pédagogiques faits par d'autres en les adaptant à leur contexte ;
- Estiment qu'il est utile de disposer de scénarios pédagogiques pour l'informatique (pour se lancer) ;
- Souhaitent des scénarios qui combinent manipulation, questionnement, observation, hypothèses et validation de la part des élèves ;
- Souhaitent des scénarios en pédagogie de projet avec des objectifs explicites et une identification des compétences travaillées ;
- Estiment que l'effort d'appropriation du scénario-test est modéré (scénario accessible) ;
- Font part d'un souci sur la mise à disposition et la fiabilité de l'instrumentation.

2.4 Discussion

Ces deux expérimentations n'ont qu'une portée limitée, mais elles montrent qu'il est possible à des enseignants de primaire qui ne sont pas formés à l'informatique et qui ne l'ont jamais enseignée de s'approprier un scénario comportant des conceptualisations simples du domaine. Par ailleurs, si l'approche pédagogique (pédagogie active intégrant des situations-problèmes) est plébiscitée, les enseignants font émerger une lacune du scénario : il manque un objectif explicite de réalisation qui permet de donner sens à l'acquisition des connaissances. Cette observation a été prise en compte pour l'élaboration du modèle de scénarisation pédagogique.

3. Elaboration d'un modèle de scénarisation pédagogique

3 .1 Cadre d'usage et éléments constitutifs du modèle

Proposé sous une forme conceptuelle (le développement d'un environnement informatique pour assister la conception n'est pas envisagé par le projet IE-CARE), le modèle est prévu pour servir de cadre à la production de scénarios d'apprentissage de l'informatique à l'école, en particulier dans le contexte d'ateliers de co-conception. Il constitue à la fois une méthodologie, un cadre de réflexion pour la production et un formalisme de description en vue d'une réutilisation des scénarios dans d'autres contextes que ceux dans lesquels ils ont été conçus.

Ce modèle comprend :

- une présentation générale : approche pédagogique et architecture globale d'un scénario ;
 - une méthodologie de conception : série d'étapes pour aboutir à un scénario validé ;
 - un langage de description : composé de descripteurs relatifs aux activités pédagogiques (objectifs didactiques, stratégies pédagogiques, mode d'organisation de la classe...) ;
 - un formalisme de représentation : proposition d'un mode de représentation graphique pour l'expression des scénarios.
-

3.2 Questions initiales et réponses

Parmi les questions qui se posent au départ de la conception du modèle :

1. Quels niveaux de granularité ?

2. Quels descripteurs ?
3. Quelles possibilités d'adaptation du scénario au contexte ?
4. Quel formalisme pour visualiser le déroulement du scénario et les possibilités d'adaptation qu'il intègre ?

Les réponses, relatives à la granularité du scénario, au langage de description et à la formalisation des scénarios, ont été construites à partir d'une analyse de modèles existants et de l'analyse de travaux portant sur les méthodologies et formalisme et de scénarisation pédagogique instrumentée par les TIC, en particulier ceux de (Pierre Dillenbourg, 2015) et de (Vassilis Komis, 2013). Les possibilités d'adaptation du scénario ont été élaborées à partir d'échanges avec les chercheurs en EIAH de l'équipe MOCAH du LIP6 (Amel Yessad et Matthieu Muratet) et des travaux de Pierre Dillenbourg (Dillenbourg, 2015).

1. Un niveau de granularité unique est retenu : l'activité didactique définie comme un ensemble organisé de tâches poursuivies par l'enseignant et les élèves pour atteindre un objectif spécifique (détecter une représentation, motiver les élèves, introduire une nouvelle notion, consolider un concept, évaluer une compétence, etc.).
2. Un ensemble de descripteurs est défini (cf. 3.3.3). Il constitue une proposition susceptible d'évoluer au cours de la suite du projet.
3. Les possibilités d'adaptation constituent une demande forte de l'équipe du LIP6 qui souhaite augmenter la robustesse du scénario. Elle est présente dans le modèle dans le descripteur « Typologie des interventions extrinsèques ». Ce terme technique nécessite quelques explications.

Au cours d'une activité, un groupe peut-être bloqué, un autre avancer beaucoup plus vite que les autres. Dans les deux cas, les élèves peuvent attendre cinq ou dix minutes l'intervention de l'enseignant ou la fin de l'activité. Pour répondre à cette situation, fréquente dans les classes, le modèle intègre des propositions d'interventions pour favoriser l'adaptation de l'activité à l'état des élèves. Ces interventions doivent pouvoir être mises en œuvre aisément dans un contexte de classe. Cela implique qu'elles soient en lien étroit avec l'activité en cours (ne pas nécessiter une re-contextualisation) et ne nécessitent au maximum qu'une instrumentation légère (une carte ou une feuille à distribuer avec la nouvelle consigne). A titre d'exemples, il peut s'agir d'une complexification ou d'une simplification de la tâche, d'une reformulation ou de l'apport d'un exemple pour éclairer un concept sous un jour nouveau, etc.

4. Deux formalismes sont envisagés : la fiche pédagogique organisée en rubriques (correspondant aux descripteurs) et un formalisme graphique reprenant le modèle des graphes d'orchestration (Dillenbourg, 2015). Cette seconde option, dont je donne un aperçu pour le scénario-test, nécessite le scénario à travers l'outil auteur associé aux graphes d'orchestration.

3.3 Modèle de scénario pédagogique : présentation

3.3.1 Approche pédagogique globale

Issus des réflexions qui ont précédé, trois grands principes guident l'approche pédagogique soutenue par le modèle :

- Une pédagogie active ;
- Un objectif de réalisation explicite à l'issue du scénario (résoudre un défi, fabriquer un objet, etc.) ;
- Un apport de la connaissance « juste à temps ».

3.3.2 Architecture globale d'un scénario

Un scénario comprend deux phases :

1. La réalisation d'un projet guidé ayant pour objectif l'acquisition de connaissances/compétences relatives à la science informatique.

Cette phase vise à l'acquisition de connaissances/compétences par les apprenants et leur mise en réussite à travers un guidage fort et une granularité fine des activités et suivant deux principes :

- Une activité ne peut apporter plus d'une nouvelle connaissance/compétence ;
- Les compétences/connaissances sont réinvesties au fil du scénario.

2. La réalisation d'un projet choisi permettant le réinvestissement des connaissances/compétences de la première phase.

La seconde phase propose aux apprenants de réaliser un projet parmi un ensemble typologie de projets proposés qui permet de réinvestir des connaissances/ compétences de la première phase. Contrairement à la première où les élèves sont fortement guidés, ils doivent opérer des choix et les justifier au moment de la présentation de leur projet.

3.3.3 Langage de description

Le langage de description est composé d'un ensemble de descripteurs permettant de caractériser une activité. L'ensemble des descripteurs n'est pas nécessaire pour décrire une activité. Le choix d'utiliser ou non un descripteur appartient au concepteur.

Intitulé	Type d'activité didactique
Commentaire	<i>Le modèle définit 5 types d'activités didactiques (objectifs).</i>
Descripteurs	<ul style="list-style-type: none"> - Préparation cognitive (pré-requis), motivation - Introduction (nouvelle notion) - Consolidation (reformulation) - Evaluation - Métacognition, synthèse (retour sur ce qui a été fait confrontation des représentations)

Intitulé	Objectif central de connaissance/compétence en science informatique
Commentaire	<i>Explicit l'objectif visé par l'activité dans le domaine de la science informatique. Dans le modèle, une activité ne doit pas viser plus d'un objectif.</i>
Descripteurs	Champ à compléter par le concepteur

Intitulé	Compétences du domaine de la pensée informatique
Commentaire	<i>Liste les compétences du domaine de la pensée informatique mises en jeu par l'activité.</i>
Descripteurs	<ul style="list-style-type: none"> - Décomposition - Pensée algorithmique - Généralisation, repérage de motifs - Evaluation - Abstraction, modélisation

Intitulé	Modalité d'organisation de la classe
Commentaire	

Descripteurs	<ul style="list-style-type: none"> - Tâche individuelle - Tâche en groupe - Tâche en groupe classe
--------------	---

Intitulé	Durée estimée
Commentaire	
Descripteurs	Champ « texte »

Intitulé	Mode
Commentaire	
Descripteurs	<ul style="list-style-type: none"> - Branché - Débranché

Intitulé	Instrumentation de l'activité
Commentaire	
Descripteurs	<ul style="list-style-type: none"> - Instrumentation élèves - Instrumentation enseignant

Intitulé	Déroulement
Commentaire	<i>Décrit les phases de déroulement de l'activité et les consignes associées.</i>
Descripteurs	Champ « texte »

Intitulé	Orchestration
Commentaire	<i>Décrit les solutions pratiques élaborées par les enseignants pour optimiser la gestion de l'activité.</i>
Descripteurs	Champ « texte »

Intitulé	Stratégie(s) d'apprentissage
Commentaire	<i>Une même activité peut faire appel à plusieurs stratégies d'apprentissage.</i>
Descripteurs	<ul style="list-style-type: none"> - Résolution de problème - Questionnement - Déstabilisation - Transmission (leçon, cours magistral) - Confrontation des représentations - Coopération

Intitulé	Interventions extrinsèques
Commentaire	<i>Décrit les propositions d'interventions pour favoriser l'adaptation de l'activité à l'état des élèves.</i>
Descripteurs	<ul style="list-style-type: none"> - Activité d'approfondissement (augmentation de la complexité de la tâche) - Activité d'aide (diminution de la complexité de la tâche) - Récompense (badge, etc.) - Intervention enseignant (reformulation, illustration par l'exemple, etc.)

Intitulé	Données en entrée
Commentaire	<i>Formulation explicite de la tâche à mener et/ou production issue d'une autre activité.</i>
Descripteurs	Champ « texte » et/ou document joint

Intitulé	Données en sortie
Commentaire	<i>Production d'élèves ou traces d'activités</i>
Descripteurs	Champ « texte » et ou document joint

3.3.4 Méthodologie d'élaboration d'un scénario à partir du modèle

La méthodologie proposée par le modèle de scénarisation (appliquée pour l'élaboration du scénario-test) propose de partir d'activités existantes (repérées ailleurs ou déjà menées) et de

les articuler dans un scénario en identifiant ce qu'elles mettent en jeu dans les domaines de la science et de la pensée informatique. Elle s'articule suivant les étapes suivantes :

1. Identifier des activités existantes ;
2. Identifier les concepts/connaissances mis en jeu dans les activités ;
3. Identifier les compétences relatives à la pensée informatique mises en jeu dans les activités ;
4. Construire un scénario pédagogique en articulant activités reprises, adaptées et nouvellement conçues ;
5. Formaliser le scénario dans un graphe d'orchestration en décrivant les activités (durée, objectif, instrumentation....) et les liens entre les activités (opérateurs) ;
6. Ajouter des descripteurs permettant d'analyser les activités et la dépendance entre les activités ;
7. Expérimenter le scénario avec des élèves ;
8. Récupérer les traces, les données en sortie ;
9. Analyser : identifier les risques d'échecs et les causes.

3.3.5 Formalisme de diffusion : les graphes d'orchestration

Je donne ici un exemple de formalisation du scénario *Alphabet de l'informatique avec Ozobot* inspiré des graphes d'orchestration proposés par Pierre Dillenbourg (Dillenbourg, 2015).

Sur cet exemple, les activités sont représentées par les nœuds du graphe. Le graphe permet de visualiser le déroulement du scénario et les possibilités d'adaptation d'une activité.

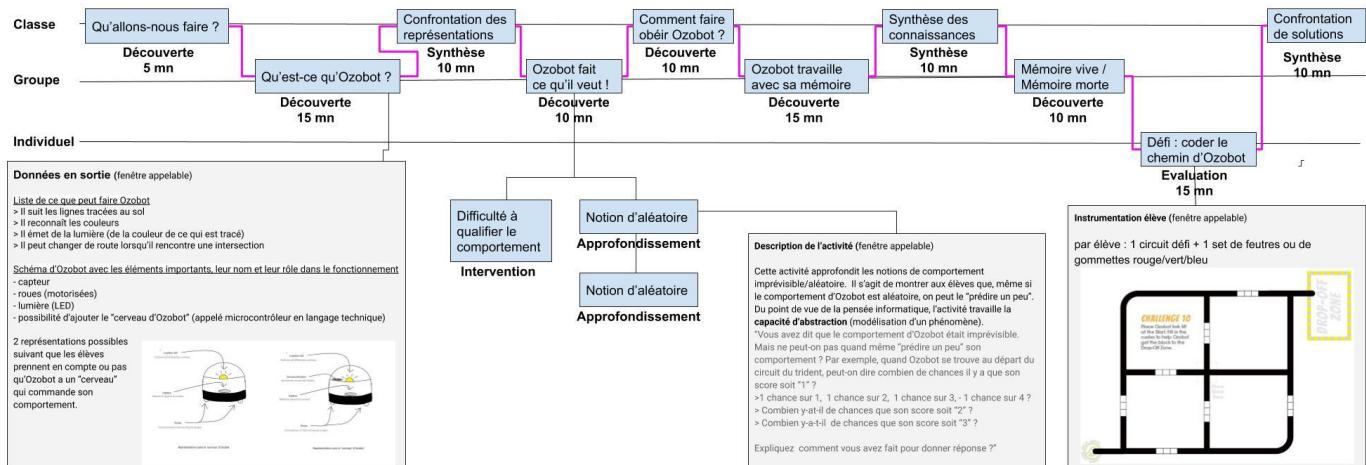


Fig. 1 – Exemple de formalisation du scénario dans un graphe d'orchestration (Olivier Brunet)

3 3.6 Discussion

Dans la plupart des activités que j'ai analysées, ou lors d'ateliers de formation à la robotique pédagogique, je remarque que les objectifs d'apprentissage mentionnés par les enseignants ne font que rarement référence à des conceptualisations en informatique. D'autres objectifs sont cités : la résolution de problèmes, la manipulation d'un robot, la coopération entre élèves, etc. Ces constats font écho aux observations rapportées dans le cadre du projet DALIE :

« Dans les discours des enseignants de notre corpus, l'idée de "faire faire quelque chose à une machine" revient systématiquement lorsqu'il s'agit de réfléchir à la notion de programmation d'un objet appelé le plus souvent robot. L'un des principaux objectifs selon ces enseignants est de faire prendre conscience aux élèves que la programmation est a priori initiée par les humains. Pour aborder cet objectif, les situations de manipulation de l'objet en lien avec une résolution de problèmes sont privilégiées par les enseignants. Ces approches pédagogiques sont en lien avec la représentation qu'ils se font de la pensée informatique qui d'après eux se construit chez les élèves en leur permettant de manipuler des robots, les observer, sur un temps durant lequel l'élève aura une posture de chercheur. Grâce aux investigations et à leurs programmations, ils devraient comprendre que l'objet peut être modelé par eux » (Voulgre, 2017).

Dans les cadres décrits ci-dessus, l'élève travaille des compétences voisines de la pensée informatique, mais il n'aborde pas la façon dont l'information est codée et traitée (science informatique).

Dans le modèle de scénarisation proposé, l'objet est au contraire d'asseoir des conceptualisations en informatique. Pour cette raison, les préconisations du modèle placent l'objectif de connaissance en informatique au centre de l'unité de base du scénario (l'activité représentée par un nœud sur le graphe) et la construction du scénario est articulée suivant une progression didactique en informatique.

La construction de ces conceptualisations est rendue possible par l'instrumentation : lorsqu'un enfant manipule un robot, il manipule déjà une machine informatique universelle. A travers elle, il devient possible de lui faire aborder les principes fondamentaux de la modélisation et du traitement de l'information par l'informatique.

Le scénario-test, *Alphabet de l'informatique avec Ozobot* (cf. annexe 2), vise ainsi l'appropriation d'un ensemble de connaissances en informatique dans trois des domaines de la typologie de Dowek (Dowek, 2011) : machines informatiques, langages de programmation, algorithmique.

Selon le modèle de scénarisation proposé, ces connaissances sont à réinvestir dans des situations complexes de projet lors de la seconde phase du scénario.

Conclusion

La science informatique va-t-elle trouver une place durable dans le cursus scolaire français ? Il est raisonnable de le penser. Ses caractéristiques fondamentales, qui ouvrent à des applications dans presque tous les domaines, font qu'elle est aujourd'hui devenue un élément incontournable de notre culture.

Sera-t-elle enseignée selon un curriculum cohérent de la maternelle à la terminale avec des conceptualisations à tous les niveaux ? Il est difficile de le dire. Au stade actuel, une scission entre les approches de l'informatique au primaire et au secondaire semble probable : un modèle avec des « cours d'informatique » au secondaire et un enseignement intégré aux autres apprentissages en primaire pourrait s'imposer, avec le risque de « faire de l'informatique » à l'école primaire sans que les élèves ne construisent des conceptualisations propres au domaine.

Au rang des obstacles exprimés à un enseignement visant des conceptualisations en informatique en primaire figurent « le trop jeune âge des enfants » et « le manque de culture informatique des professeurs des écoles ». Des expérimentations montrent pourtant que les élèves sont capables de s'approprier des conceptualisations propres au domaine dès la Grande Section et que les professeurs des écoles sont aptes à mettre en œuvre des situations d'apprentissage pertinentes s'ils disposent d'un guidage pédagogique adapté et de l'instrumentation nécessaire.

Travailler à développer des ressources pédagogiques en phase avec ce que peuvent s'approprier les enseignants et permettant des conceptualisations en informatique a été l'axe principal de mon action au cours de ce stage, aboutissant à la proposition d'un modèle de scénarisation qui servira de cadre pour la production de scénarios destinés à l'enseignement/apprentissage de l'informatique dans le cadre d'IE-CARE.

Je regrette pour ma part que ce développement n'ait pu se dérouler dans un cadre de co-conception associant des enseignants de terrain tel qu'initialement prévu. Il me semble que le modèle proposé souffre quelque peu d'une conception « en laboratoire ». La seconde phase du

projet IE-CARE qui débute à la rentrée 2019 permettra des ajustements. J'y prendrai part dans la mesure de mes possibilités.

Si la co-conception avec les enseignants de terrain n'a pas été possible, ce stage m'a permis de confronter mon approche à celle d'enseignants et de chercheurs travaillant avec conviction sur le même sujet. C'est dans cet enrichissement mutuel, à côté de l'acculturation aux méthodologies de recherche, du développement de mes compétences en ingénierie pédagogique et de l'approfondissement des approches des EIAH que je situe mes acquis. Ils sont plus larges si je considère cette année de Master qui m'a permis de prendre du recul et d'ouvrir de nouvelles perspectives professionnelles.

Bibliographie

Arsac, J. (1981). Annexe 1. L'éducation et l'informatisation de la société. Rapport au Président de la République sous la direction de Jean-Claude Simon, *La Documentation française*, 152-165.

Baron, G.-L. (1990). Note de synthèse. *Revue française de pédagogie*, 92(1), 57-77.

Baron, G.-L. (2014). Elèves, apprentissages et numérique : regard rétrospectif et perspectives. *Recherches en éducation*, 18, 91-103. [Consulté le 10 mai 2019]. <http://www.recherches-en-education.net/IMG/pdf/REE-no18.pdf>

Baron, G.-L., Bruillard, E. (2001). Une didactique de l'Informatique, *Revue française de Pédagogie*, n°135, p.163-172. [Consulté le 10 mai 2019]. http://ife.ens-lyon.fr/publications/edition-electronique/revue-francaise-de-pedagogie/INRP_RF135_23.pdf

Baron, G-L (2019). L'informatique, discipline scolaire : enjeux de savoir et de pouvoir. Conférence donnée le 10 mai 2019 à l'université de Fribourg [Consulté le 6 juillet 2019]. <https://www.lip-unifr.ch/2019/05/16/le-3e-anniversaire-du-lip-en-images/>

Berry, G. (2017). L'hyperpuissance de l'informatique. Algorithmes, données, machines, réseaux, Paris, Odile Jacob

Bers, M. U., Flannery, L., Kazakoff, E. R., & Sullivan, A. (2014). Computational thinking and thinkering: Exploration of an early childhood robotics curriculum. *Computers & Education*, 72, 145–157

Bers, M. (2017). Coding as a Playground : Programming and Computational Thinking in the Early Childhood Classroom, London, Taylor & Francis Ltd

Bers, M. (2018). La programmation en tant que place de jeu développementale : la pensée informatique et la robotique dans la petite enfance. In Parriaux, G., Pellet, J-P., Baron G-L.,

Bruillard, E., Komis, V. De 0 à 1 ou l'heure de l'informatique à l'école, Actes du colloque DidaPro 7 – DidaSTIC. Lausanne, Suisse. Peter Lang, 2018

Carron, T., Houzet, G., Abed, H., Pernelle, P-J., Lainé, S. (2018). Teaching digital literacy : the outcomes from a learning lab, Journal of electrical engineering, Vol 6 (2), 75-84 [Consulté le 30 avril 2019]. <https://hal.archives-ouvertes.fr/hal-01889025>

Cour des Comptes (2019). Le service public numérique pour l'éducation : un concept sans stratégie, un déploiement inachevé, Rapport de la Cour des Comptes, Juillet 2019. [Consulté le 12 juillet 2019]. <https://www.ccomptes.fr/system/files/2019-07/20190708-rapport-service-public-numerique-education.pdf>

De la Higuera, C., Canellas, C., Roche, M. (2017). Class'Code, Objet de Recherche. ORPHEE Rendez-vous - Apprentissage Instrumenté de l'Informatique, Font-Romeu, France. [Consulté le 11 juillet 2019]. <https://hal.archives-ouvertes.fr/hal-01772175/document>

Dillenbourg, P. (2013). Design for classroom orchestration, Computers & Education 69 485–492. [Consulté le 25 mars 2019] https://edisciplinas.usp.br/pluginfile.php/2745230/mod_resource/content/7/Dillenbourg-CE-A8Extra.pdf

Dillenbourg, P. (2015). Orchestration Graphs : Modeling Scalable Education, Lausanne, EPFL Press

Doueihi, M. (2013). Qu'est-ce que le numérique ?, Paris, PUF

Dowek, G.(2011). Les quatre concepts de l'informatique. Sciences et technologies de l'information et de la communication en milieu éducatif : Analyse de pratiques et enjeux didactiques, Patras, Grèce. pp.21-29. [Consulté le 20 juillet 2019]. <https://edutice.archives-ouvertes.fr/edutice-00676169>

Drot-Delange, B. (2018). Reconfiguration de l'enseignement de l'informatique à l'école primaire : quelle conscience disciplinaire chez les professeurs des écoles stagiaires ?,

Recherches en didactiques 2018/1 N° 25 | pages 27 à 40. [Consulté le 13 juillet 2019].
<https://tel.archives-ouvertes.fr/tel-02185302/document>

Drot-Delange, B., & Tort, F. (2018). Résolution de défis et pensée informatique. Présenté aux 10e rencontres scientifiques de l'ARDIST, Saint-Malo. [Consulté le 5 juillet 2018]
<https://hal.archives-ouvertes.fr/hal-01851772/document>

Fluckiger, C. (2019). Une approche didactique de l'informatique scolaire, Rennes, Presses Universitaires de Rennes

Institut de France, Académie des sciences, (2013). L'enseignement de l'informatique en France – Il est urgent de ne plus attendre. [Consulté le 13 juillet 2019].
<https://www.academie-sciences.fr/fr/Rapports-ouvrages-avis-et-recommandations-de-l-Academie/l-enseignement-de-l-informatique-en-france-il-est-urgent-de-ne-plus-attendre.html>

Komis, V., Tzavara, A., Karsenti, T., Collin, S. & Simard, S. (2013). Educational scenarios with ICT: an operational design and implementation framework. In R. McBride & M. Searson (Eds.), Proceedings of SITE 2013--Society for Information Technology & Teacher Education International Conference (pp. 3244-3251). New Orleans, Louisiana, United States: Association for the Advancement of Computing in Education (AACE). [Consulté le 11 juillet 2019]. <https://www.learntechlib.org/primary/p/48594/>

Kradolfer, S., Dubois, S., Riedo, F., Mondada, F., Fassa, F. (2014). A sociological contribution to understanding the use of robots in schools: the Thymio robot - ICSR 2014, LNAI 8755, 217-228. [Consulté le 20 avril 2019].
<https://infoscience.epfl.ch/record/200503?ln=fr>

Lazzarotto Chessel, F. (2019), Former à la programmation en primaire, une form'action : Robots d'Evian 2015-2018, Didapro 7 – In Parriaux, G., Pellet, J-P., Baron G-L., Bruillard, E., Komis, V. De 0 à 1 ou l'heure de l'informatique à l'école, Actes du colloque DidaPro 7 – DidaSTIC. Lausanne, Suisse. Peter Lang, 2018

Martinez-Emin V., Pernin J-P., Guéraud V. (2011). Scénarisation pédagogique dirigée par les intentions. In Sciences et Technologies de l'Information et de la Communication pour l'Éducation et la Formation, volume 18, 2011. TICE. pp. 195-227. [Consulté le 16 juillet 2019]. https://www.persee.fr/doc/stice_1952-8302_2011_num_18_1_1024

Papert, S. (1981). *Jaillissement de l'esprit ordinateurs et apprentissage*. Edition 1999. Paris, Flammarion

Rogalski, J. (1987). Acquisition de savoirs et savoir-faire en informatique. Cahiers de Didactique des Mathématiques, 43

Tchounikine, P. (2017). Initier les élèves à la pensée informatique. <http://lig-membres.imag.fr/tchounikine/PenseeInformatiqueEcole.pdf>

Touloupaki, S., Baron, G-L., Komis, V. (2018). Un apprentissage de la programmation dès l'école primaire : le concept de message sur ScratchJr. Didapro 7 – DidaSTIC. De 0 à 1 ou l'heure de l'informatique à l'école. Lausanne, Suisse. [Consulté le 5 juillet 2019]. <https://hal.archives-ouvertes.fr/hal-01753125/document>

Villiot-Leclercq, E. Jean-Pierre David, J-P., 2007). Le formalisme des Pléiades pour la conception et l'adaptation de patrons de scénarios pédagogiques, INRP, 2007. [Consulté le 27 juillet 2019]. <https://hal.archives-ouvertes.fr/hal-00161569/document>

Villemonteix, F. (2016). Quel enseignement de l'informatique à l'école primaire ? Le point de vue d'acteurs de l'accompagnement des pratiques pédagogiques. Didapro 6 – DidaSTIC. Quelles éducations au numérique, en classe et pour la vie. Namur, Belgique. [Consulté le 10 juillet 2019]. <https://didapro6.sciencesconf.org/76425/document>

Villemonteix, F., Beziat J., Baron, G-L. (2016). L'école primaire et les technologies informatisées : des enseignants face aux TICE, Villeneuve d'Ascq, Presses Universitaires du Septentrion

Voulgre, Emmanuelle (2017). “Cinq instruments issus de l'ANR DALIE pour accompagner et guider le processus de formation des enseignants et formateurs au cours d'un projet en

robotique “, issus de l’analyse des données de terrain, Site Yvelines, in Livre blanc de l’ANR DALIE, PDF, 25p., Laboratoire Éducation, Discours et Apprentissages (EDA), EA 4071, Université Paris Descartes. [Consulté le 12 juillet 2019].
<https://drive.google.com/file/d/1UiP0cx3CNFh4AaurldOGjvU5ROUdn4nv/view>

- Wing, J. (2006). Computational thinking, Communication of the ACM, Vol 49, No. 3, 33-35. [Consulté le 15 juillet 2019]. <https://www.cs.cmu.edu/~15110-s13/Wing06-ct.pdf>
- Yadav, A., Mayfield, C., Zhou, N., Hambrusch, S. & Korb, J.T. (2014). Computational Thinking in Elementary and Secondary Teacher Education. ACM Transactions on Computing Education, Vol. 14-1

Index

- apprentissage de l'informatique, 3, 34, 36, 45
conceptualisations en informatique, 17, 24, 25, 26, 28, 32, 43, 45
école primaire, 32
graphes d'orchestration, 2, 38, 42
informatique à l'école, 1, 5, 23, 30
instrumentation, 2, 3, 10, 25, 26, 27, 28, 31, 35, 37, 42, 43, 45
modèle de scénarisation, 1, 2, 3, 4, 20, 21, 36, 41, 43, 44, 45, 20
numérique, 1, 2, 5, 6, 7, 8, 9, 11, 12, 15, 16, 17, 18, 27, 28, 29, 47, 48, 50, 15, 16
orchestration de la classe, 2, 28
pédagogie active, 38
pédagogie de projet, 25, 35
pensée informatique, 2, 3, 11, 13, 14, 17, 19, 24, 29, 30, 31, 33, 39, 42, 43, 47, 49, 50, 2, 9, 20
projet de recherche ANR, 1, 3, 5, 23, 20
robotique pédagogique, 43
scénarisation pédagogique, 1, 2, 3, 4, 20, 21, 36, 37, 20
science informatique, 1, 3, 4, 5, 6, 8, 15, 16, 17, 18, 20, 21, 24, 38, 39, 43, 45, 2, 15, 20

Annexe 1 - La pensée informatique

Présentation des compétences couvertes par la pensée informatique au sens défini par Jeannette Wing

Cette présentation s'inscrit dans le cadre défini par Jeannette Wing (Wing, 2006). Elle a été élaborée sur la base du travail de vulgarisation destiné aux enseignants dans le cadre du projet *CS Unplugged (What is computational thinking ? - <https://csunplugged.org/en/computational-thinking/>)*.

La pensée algorithmique

L'algorithmique (ou pensée algorithmique) est le cœur de la pensée informatique et de la science informatique. Ecrire un algorithme, c'est décrire pas-à-pas la façon de résoudre un problème ou d'exécuter une tâche de façon à ce que la résolution du problème ou la réalisation de la tâche puisse être exécutée simplement en suivant les étapes de l'algorithme. Sans qu'il y ait besoin de réfléchir !

Les algorithmes sont déjà présents à l'école. Ainsi, lorsqu'un enseignant apprend aux enfants à faire l'addition posée de 2 nombres a et b, il leur fournit un algorithme, c'est-à-dire une description du processus découpée en étapes élémentaires :

1. écrire les nombres a et b l'un en dessous de l'autre en alignant verticalement les chiffres des unités, les chiffres des dizaines, etc.
2. additionner le chiffre des unités du nombre a avec le chiffre des unités du nombre b...

Cet exemple montre plusieurs choses concernant les algorithmes :

1. il existe plusieurs manières de réaliser une tâche ou de résoudre un problème et donc plusieurs algorithmes pour un même problème ou une même tâche
2. écrire un algorithme oblige à s'interroger sur la façon dont on réalise la tâche
3. un même algorithme permet de traiter des instances différentes : dans notre cas, l'algorithme continue à fonctionner si on change les nombres a et b à additionner.

L'abstraction

L'abstraction est la capacité qui nous permet de construire une image simplifiée des choses, en enlevant tout ce dont nous n'avons pas besoin dans une situation donnée.

Lorsque nous avons un problème à traiter, abstraire consiste à identifier les éléments importants pour traiter le problème, puis à construire un modèle (une représentation simplifiée) limité aux éléments importants. Par exemple, pour savoir si l'on pourra caser les meubles de notre actuelle maison dans notre futur appartement, peu importe de connaître la couleur des meubles ou celle de la pièce. Seules la dimension et à la forme des meubles et de la pièce sont importantes pour résoudre ce problème. Une fois que l'on a identifié cela, on construit un modèle : on trace le plan de la pièce et on dessine et découpe la forme des meubles en respectant l'échelle. Ce modèle omet des tas d'aspects de la réalité, mais il est suffisant pour résoudre notre problème.

Dans notre vie courante, nous utilisons souvent des modèles simplifiés de la réalité (par exemple, le plan d'une ville ou celui d'un réseau de transports) car ils facilitent notre action : ils nous permettent d'étudier les différentes options qui s'offrent à nous sans avoir besoin que les choses ne se déroulent en vrai et nous aident à prendre des décisions (comment organiser les meubles dans mon futur salon, quel est l'itinéraire le plus court pour aller à mon RDV, etc.).

La décomposition

La décomposition consiste à fragmenter un problème (ou une tâche) en mini-problèmes (ou mini-tâches), plus faciles à résoudre. Nous faisons souvent cela dans la vie. Par exemple, lorsque nous préparons un gâteau, nous décomposons la préparation en étapes :

- 1- Mettre la farine, le sucre, les œufs, le beurre dans un grand bol
- 2- Mélanger jusqu'à obtenir une pâte homogène
- 3- Verser la pâte dans un moule
- 4- Mettre au four durant 30 mn

Etre capable de décomposer un problème en sous-problèmes est essentiel pour pouvoir écrire des algorithmes destinés à être traités par des machines informatiques. Cela est dû au fait que les ordinateurs ne peuvent comprendre qu'un petit nombre d'instructions élémentaires. Par exemple, pour programmer le déplacement d'un petit robot qui ne comprend que les quatre instructions “avance de n cm”, “pivoté à droite”, “pivoté à gauche”, “arrête-toi”, il faut décomposer son chemin en instructions élémentaires :

- 1- avance de 30 cm
- 2- pivoté à droite
- 3- avance de 10 cm

4- pivote à gauche, etc.

La généralisation

La généralisation est associée à la reconnaissance de motifs et de similarités. C'est un moyen de résoudre rapidement de nouveaux problèmes en réutilisant les solutions de problèmes déjà résolus. Par exemple, en écrivant un programme pour dessiner des triangles et des carrés, les élèves remarquent qu'il y a une relation entre le nombre de côtés et la valeur des angles. Aussi, pour écrire rapidement un programme permettant de dessiner un autre type de polygone, il leur suffit de réutiliser cette relation.

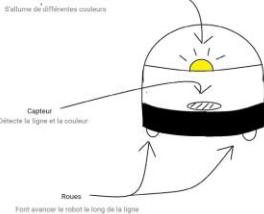
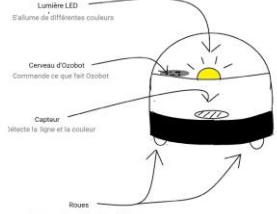
L'évaluation

L'évaluation consiste à identifier les solutions possibles à un problème et à choisir la plus appropriée. Dans le cas où la solution est un algorithme, l'évaluation peut porter sur sa vitesse d'exécution, sur sa consommation de ressources, ou plus fondamentalement, sur sa capacité à traiter tous les cas possibles du problème. Simuler le déroulement de l'algorithme dans un certain nombre de cas ou prouver qu'il fonctionne dans tous les cas par un raisonnement logique sont deux manières d'évaluer. Quel que soit le mode d'évaluation retenu, il fait principalement appel au raisonnement logique.

Annexe 2 – Scénario *Alphabet de l'informatique avec Ozobot*

Activité 1 : Qu'est-ce qu'Ozobot ?

Nom descripteur	Valeur descripteur
Typologie	<p>Introduction <i>> Observation du comportement du robot, repérage des éléments qui participent à ce comportement, et lien entre les éléments du robot et son comportements</i></p>
Organisation de la classe	groupe (2-5)
Objectif de connaissance	<p><u>> Connaissances minimales issues de l'observation visées à l'issue de l'activité</u> - Ozobot a un capteur (situé en dessous), des roues (motorisées), une lumière (située au-dessus) - Ozobot peut détecter qu'il y a une quelque chose de tracé et peut détecter la couleur de ce qui est tracé grâce à son capteur situé en dessous</p> <p><u>> Hypothèses plus avancées sur le fonctionnement d'Ozobot pouvant être élaborées par raisonnement</u></p> <ul style="list-style-type: none"> - Le capteur envoie des informations au cerveau électronique d'Ozobot. Grâce à ces informations, le cerveau électronique commande les roues (leur vitesse) et décide de la couleur de la lumière. - Si Ozobot peut changer de direction, c'est parce que ses roues ne tournent pas toujours toutes les deux à la même vitesse
Durée estimée	15 mn
Instrumentation élèves	par groupe : 1 robot Ozobot Bit + 1 grande feuille de papier blanc (A3 min) + 1 set de gros feutres (noir + couleurs)
Description activité	<p>t1 > Chaque groupe reçoit le matériel t2 > L'enseignant présente Ozobot et l'activité « Ozobot est un robot capable de suivre un chemin qui a été tracé sur une feuille et de s'allumer de différentes couleurs. C'est pour le moment la seule information que je vous donne à son sujet. Votre première mission est d'en découvrir plus sur lui. Faites-le fonctionner, observez-le et faites la liste de tout ce qu'il peut-faire.” t3 > Les élèves observent le fonctionnement d'Ozobot et notent tout ce qu'il peut faire (cf. Données en sortie). t4 > “Votre seconde mission est de faire un schéma d’Ozobot avec les parties qui lui permettent de suivre un chemin tracé et de s'allumer de différentes couleurs, de donner un nom à chacune de ces parties et d'expliquer à quoi elle sert.”</p> <p>t5 > Les élèves réalisent un schéma devant contenir les éléments importants d’Ozobot (cf. Données en sortie), un nom associé à chaque éléments et son rôle dans le fonctionnement d’Ozobot t6 > L'enseignant peut demander aux groupes s'ils pensent que les différentes parties du robot se transmettent des informations.</p>
Données en entrée	> Instrumentation élèves

	<p>> Consigne de l'enseignant sur l'activité</p>
Données en sortie	<p><u>Liste de ce que peut faire Ozobot</u></p> <ul style="list-style-type: none"> > Il suit les lignes tracées au sol > Il reconnaît les couleurs > Il émet de la lumière (de la couleur de ce qui est tracé) > Il peut changer de route lorsqu'il rencontre une intersection <p><u>Schéma d'Ozobot avec les éléments importants, une dénomination et leur rôle dans le fonctionnement d'Ozobot</u></p> <ul style="list-style-type: none"> - capteur - roues (motorisées) - lumière (LED) - possibilité d'ajouter le "cerveau d'Ozobot" (appelé microcontrôleur en langage technique) <p>Exemples de représentations de l'architecture d'Ozobot.</p> <p>2 représentations sont possibles suivant que les élèves prennent en compte ou pas qu'Ozobot a un "cerveau" qui commande son comportement.</p> <div style="display: flex; justify-content: space-around; align-items: center;"> <div style="text-align: center;">  <p>Représentation sans le "cerveau" d'Ozobot</p> </div> <div style="text-align: center;">  <p>Représentation avec le "cerveau" d'Ozobot</p> </div> </div> <p>- informations transmises du capteur aux roues - informations transmises du capteur à la lumière LED</p>
<p>Interventions possibles</p> <p>> Peut-être enrichi par les enseignants qui mettent en oeuvre le scénario</p>	<p>> <u>Cas où les élèves ne repèrent pas le capteur et son rôle</u></p> <ul style="list-style-type: none"> - Comment Ozobot peut-il savoir qu'il y a un chemin tracé sous lui ? A-t-il un œil ? Où est situé cet œil ? - Ozobot reconnaît-il les couleurs ? Comment savez-vous qu'il les reconnaît ? Où est l'œil qui lui permet de voir les couleurs ? <p><u>Cas où les élèves veulent expliquer comment Ozobot fait des virages alors que ses roues ne pivotent pas</u></p> <ul style="list-style-type: none"> - Faire appel à une l'image d'une autre situation : vous êtes sur un bateau que vous voulez faire avancer. Pour ça, vous avez deux rames qui sont accrochées au bateau. Avec ces rames, vous ne pouvez faire qu'un seul mouvement (le mimer). Si vous faites le même mouvement avec les deux rames en même temps, le bateau va tout droit. Comment faire pour faire aller le bateau sur la droite ou sur la gauche ? <p><u>Cas où des élèves s'interrogent sur les composants électroniques qui sont visibles sous la coque</u></p> <ul style="list-style-type: none"> - Dire simplement qu'il s'agit du cerveau d'Ozobot (c'est lui qui commande Ozobot) et que l'on peut le mettre dans le schéma

Traces d'activité	Aucune
-------------------	--------

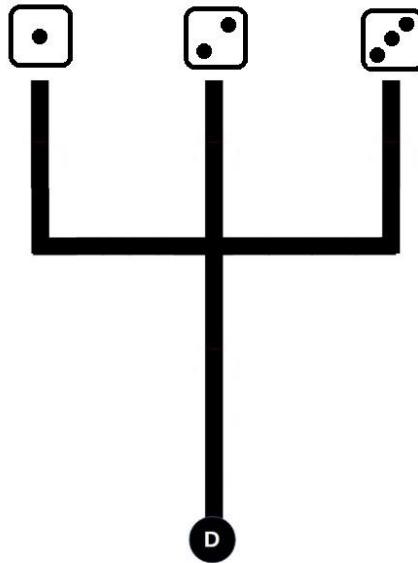
Activité 2 : confrontation des représentations et synthèse

Nom descripteur	Valeur descripteur
Typologie	Métacognition > <i>Synthèse concernant l'architecture et le fonctionnement d'Ozobot à partir de la confrontation des représentations issues de l'activité 1</i>
Organisation de la classe	Classe entière
Objectif de connaissance	> <u>Connaissances minimales visées à l'issue de l'activité issues de l'observation</u> Un consensus sur ce que les élèves savent de l'architecture et du fonctionnement d'Ozobot : - Grâce à son capteur situé en dessous, Ozobot peut détecter une route tracée au sol et détecter sa couleur. - Lorsque Ozobot détecte une route tracée au sol, il la suit. - Lorsque Ozobot détecte une couleur, il allume sa lumière de cette couleur > <u>Hypothèses plus avancées sur le fonctionnement d'Ozobot pouvant être élaborées par raisonnement</u> - Les connaissances précédentes + Ozobot a un "cerveau" qui reçoit les informations du capteur. C'est le "cerveau" d'Ozobot qui commande la lumière et les roues
Durée	mn
Instrumentation élèves	
Description activité	t1 > Chaque groupe présente son schéma de l'architecture et du fonctionnement d'Ozobot t2 > L'enseignant organise la synthèse à partir de la confrontation des représentations
Données en entrée	> Consigne de l'enseignant sur l'activité > Ce qui a été produit par les élèves lors de l'activité précédente : - liste de ce que peut faire Ozobot - schéma de l'architecture d'Ozobot
Données en sortie	
Interventions possibles > <i>Peut-être enrichi par les enseignants qui mettent en oeuvre le scénario</i>	
Traces d'activité	

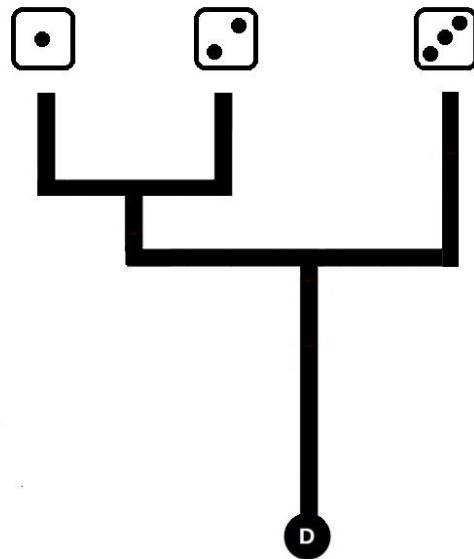
Activité 3 : Peut-on prédire le chemin d’Ozobot ?

Nom descripteur	Valeur descripteur
Typologie	<p>Introduction <i>> Introduction de la notion de comportement aléatoire/imprévisible par l'observation du comportement du robot lorsqu'il a le choix entre plusieurs options. Cette activité constitue une préparation à l'introduction de la notion de programmation/codage : pour rendre comportement du robot prévisible, il faut le programmer/coder.</i></p>
Organisation de la classe	groupe (2-5)
Objectif de connaissance	<p>> <u>Connaissances minimales visées à l'issue de l'activité issues de l'observation</u> On ne peut prévoir le comportement d’Ozobot lorsqu'il a le choix entre plusieurs chemins : son comportement est imprévisible/aléatoire.</p> <p>> <u>Hypothèses plus avancées sur le fonctionnement d’Ozobot pouvant être élaborées par raisonnement</u> Il y a un programme dans Ozobot qui décide où Ozobot doit aller / Il y a un programme dans la mémoire d’Ozobot qui décide où il doit aller.</p>
Durée	mn
Instrumentation élèves	par groupe : 1 robot Ozobot Bit + 1 grande feuille de papier blanc (A3 min) sur laquelle est tracé le “circuit du trident”
Description activité	<p>t1 > Chaque groupe reçoit ou trace d'après un modèle dessiné au tableau le “circuit du trident”</p> <p>t2 > L'enseignant demande de faire effectuer à Ozobot 10 parcours sur le circuit du trident et de noter à chaque fois le score d’Ozobot (1, 2 ou 3) : « Vous allez tenter d'en savoir plus sur Ozobot. Et notamment : que se passe-t-il si Ozobot a le choix entre plusieurs routes ? Peut-on prédire ce qu'il va faire ?</p> <p>Faites-lui parcourir le circuit du trident (au moins) 10 fois de suite et notez à chaque fois son score. Par exemple, s'il va sur le 1 la première fois, et sur le 3 la seconde fois, notez :</p> <p>score 1 > 1 score 2 > 3 Etc. Trouvez un adjectif qui décrit le comportement que vous avez observé.”</p>
Données en entrée	<p>> Consigne de l'enseignant sur l'activité</p> <p>> Instrumentation élève</p>
Données en sortie	<p>Un adjectif qui résume le comportement d’Ozobot : - imprévisible - aléatoire</p> <p>Car Ozobot va soit à droite, soit à gauche, soit tout droit, sans que l'on sache comment il prend ses décisions</p>

<p>Interventions possibles <i>> Peut-être enrichi par les enseignants qui mettent en oeuvre le scénario</i></p>	<p>> <u>Cas où les élèves ne trouvent pas d'adjectif pour qualifier le comportement d'Ozobot</u> Pour faire émerger les qualificatifs “aléatoire” ou “imprévisible”, l’enseignant peut insister sur la métaphore du dé ou de la pièce de monnaie dont on ne peut prédire le tirage à l’avance.</p> <p>> <u>Approfondissement Activité 2 (niveau 1)</u> Cette activité approfondit les notions de comportement imprévisible/aléatoire. Il s’agit de montrer aux élèves que, même si le comportement d’Ozobot est aléatoire, on peut le “prédire un peu”. Du point de vue de la pensée informatique, l’activité travaille la capacité d’abstraction (modélisation d’un phénomène). “Vous avez dit que le comportement d’Ozobot était imprévisible. Mais ne peut-on pas quand même “prédire un peu” son comportement ? Par exemple, quand Ozobot se trouve au départ du circuit du trident, peut-on dire combien de chances il y a que son score soit “1” ? <ul style="list-style-type: none"> - 1 chance sur 1 - 1 chance sur 2 - 1 chance sur 3 - 1 chance sur 4 Combien y-at-il de chances que son score soit “2” ? Combien y-a-t-il de chances que son score soit “3” ? Expliquez comment vous avez fait pour donner réponse ?”</p> <p><u>Approfondissement Activité 2 niveau 2</u> “Sur le circuit du trident, Ozobot a donc : <ul style="list-style-type: none"> - 1 chance sur 3 de faire “1” comme score - 1 chance sur 3 de faire “2” comme score - 1 chance sur 3 de faire “3” comme score Je voudrais qu’Ozobot ait plus de chances de faire “3” comme score... parce que ce c'est un plus gros score ! Comment faudrait-il modifier le circuit du trident pour que Ozobot ait plus de chances de faire “3”, que “1” ou “2” ? Pouvez-vous vérifier que votre solution fonctionne ?”</p>
Traces d’activité	> Notes prises par les groupes : série des directions prises par le robot et caractérisation du comportement par les élèves
Documents associés	- Circuit du trident



- Circuit du trident modifié (solution de l'activité d'approfondissement niveau 2)



Activité 4 : Peut-on donner des instructions à Ozobot ? Et comment ?

Nom descripteur	Valeur descripteur
Typologie	Introduction > <i>Introduction de la notion d'instruction, de programme et de langage de programmation</i>
Organisation de la classe	classe
Objectif de connaissance	> <u>Connaissances minimales visées à l'issue de l'activité issues de</u>

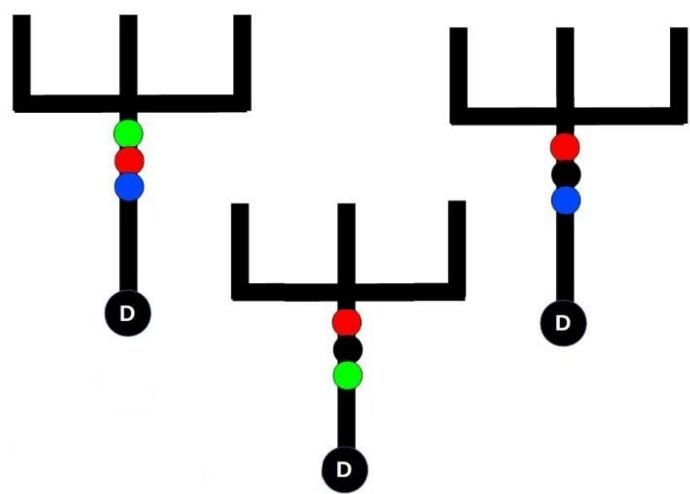
	<p><u>l'observation</u></p> <p>Pour qu'Ozobot suive le chemin que l'on a décidé, il faut lui donner des instructions.</p> <p>Pour cela, il faut un langage spécial que peut comprendre Ozobot : un langage de programmation.</p> <p>Le langage compris par Ozobot est fait de codes couleurs..</p> <p>Une code couleur correspond à une instruction.</p>
Durée	mn
Instrumentation élèves	
Instrumentation enseignant	<ul style="list-style-type: none"> - 1 Ozobot - La table des Ozocodes
Description activité	<p>Cette activité demande aux élèves, à travers un cheminement de questions, d'émettre des hypothèses sur la façon de programmer Ozobot.</p> <p>t1: On fait émerger la nécessité de pouvoir donner des instructions à Ozobot si on veut le diriger Ozobot.</p> <p>Connaissance importante : notion de programme, de programmation “Un robot qui va où il veut, c'est drôle mais ce n'est pas très pratique. S'il est impossible de prédire sa route, on ne pourra lui confier aucune tâche (par ex. transporter quelque chose d'un endroit à un autre). C'est bien embêtant ! Mais il doit bien y avoir un moyen de faire aller Ozobot où l'on veut. Que nous faudrait-il pour cela ?”</p> <p>> pouvoir lui envoyer des instructions, le programmer, le coder... sont des réponses probables des élèves.</p> <p>t2 : On amène les enfants à établir le lien entre ce que peut faire Ozobot (reconnaître les couleurs détectées par son capteur) et le moyen de lui communiquer des instructions</p> <p>“Il faudrait (effectivement) pouvoir lui envoyer des instructions, le programmer, le coder. Mais comment communiquer avec lui ? Si quelqu'un veut nous donner des instructions, il doit utiliser un langage que nous connaissons : des mots en français car nous sommes capables de les reconnaître et de les comprendre.</p> <p>Ozobot est-t-il capable de reconnaître quelque chose ?</p> <p>> les couleurs</p> <p>Mais cela ne suffit pas. Il faut aussi que les instructions arrivent jusqu'à notre cerveau. Nous, nous avons nos oreilles (pour entendre) et nos yeux (pour lire) les instructions. Ozobot a-t-il quelque chose qui remplace les yeux ou les oreilles ?</p> <p>> un capteur</p> <p>On pose la question “Alors, à quoi pourrait ressembler une instruction pour Ozobot et comment lui communiquer ?</p> <p>On laisse les élèves émettre des hypothèses que l'on teste avec un Ozobot</p> <p>> Ce pourrait être des instructions avec des couleurs.: par exemple “Si Ozobot rencontre un trait bleu, alors il doit tourner à gauche” ou “Si Ozobot rencontre un trait rouge, alors il doit tourner à gauche”...</p>

	<p>t3 : on dévoile la table des Ozocodes.</p> <p>“ Ozobot se programme avec des codes de couleurs.C'est son langage de programmation. Chaque code correspond à une instruction. vous allez tester...”</p>
Données en entrée	<ul style="list-style-type: none"> > Consigne de l'enseignant sur l'activité > Savoir qu'Ozobot : <ul style="list-style-type: none"> - a un capteur qui lui permet de suivre les lignes tracées et de repérer les couleurs - > Avoir observé qu'on ne peut prédire la direction qu'Ozobot lorsqu'il arrive à une intersection
Données en sortie	
Interventions possibles <i>> Peut-être enrichi par les enseignants qui mettent en oeuvre le scénario</i>	
Traces d'activité	

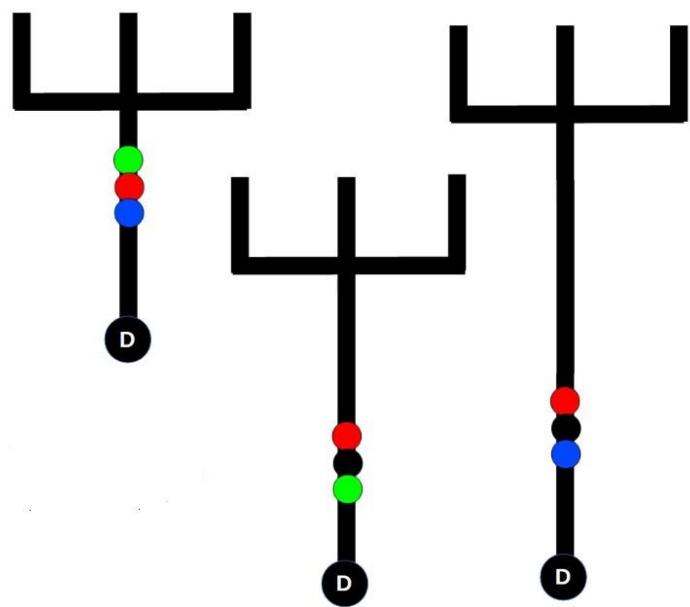
Activité 5 : Ozobot a de la mémoire

Nom descripteur	Valeur descripteur
Typologie	Introduction <i>> Introduction du rôle de la mémoire : le robot met les instructions qu'il reçoit en mémoire jusqu'à ce qu'il les exécute. Un fois qu'il a exécuté une instruction, il l'oublie.</i>
Organisation de la classe	groupe (2-5)
Objectif de connaissance	<u>> Connaissances minimales visées à l'issue de l'activité issues de l'observation</u> Ozobot a une mémoire. Lorsqu'il reçoit une instruction, il la met dans sa mémoire jusqu'à ce qu'il l'exécute puis il “l'oublie”.
Durée	mn
Instrumentation élèves	pour l'ensemble de l'activité - par groupe : 1 robot Ozobot Bit t1- par groupe : grande feuille de papier blanc (A3 min) sur laquelle sont tracés les “circuits du trident avec les codes (“tourner à gauche”, “tourner à droite”, “aller tout droit”) collés aux intersections t2- par groupe : grande feuille de papier blanc (A3 min) sur laquelle sont tracés les “circuits du trident avec les codes (“tourner à gauche”, “tourner à droite”, “aller tout droit”) éloignés des intersections
Instrumentation enseignant	

Description activité	t1 : l'enseignant distribue les circuits du trident 1. “Trouvez et notez la signification de chacun des codes.” t2 : l'enseignant distribue les circuit du trident 2. “Vous allez faire des prédictions ! Que va -t-il se passer si l'on éloigne les codes des intersections ? Notez votre prédition et testez avec Ozobot ? Cela vous apprend-il quelque chose de nouveau sur Ozobot ?” > Cela montre qu'Ozobot se souvient du code jusqu'à l'intersection : Ozobot a donc une mémoire dans laquelle il met les codes jusqu'à ce qu'il les exécute. t3 : “S'il y a une autre intersection derrière la première, que va-t-il se passer ? Ozobot va-t-il exécuter le même instruction une seconde fois ou l'aura-t-il ’oubliée’ ? Dessinez un circuit pour savoir ce qui se passe.”
Données en entrée	> Consigne de l'enseignant sur l'activité > Savoir que : - on peut donner des instructions à Ozobot en plaçant des Ozocodes (codes couleurs qu'il comprend) sur son chemin
Données en sortie	
Interventions possibles > <i>Peut-être enrichi par les enseignants qui mettent en oeuvre le scénario</i>	<u>Activité de déstabilisation</u> “Vous allez faire des prédictions sur ce qui va se passer!” Une portion avec une seconde instruction qui suit la première suivie d'une portion de circuit sans instruction Qu'est-ce qui va se passer à la première intersection ? Et à la seconde ? Noter sa prédition ?
Traces d'activité	
Document associé	Circuits du trident avec codes collés



Circuits du trident 2 avec codes à distances variables



Annexe 3 – Résultats de l’expérimentation du scénario-test

Enquête auprès de 22 enseignants en formation initiale

M2 - MEEF - Mention Premier Degré
ESPE de l’académie de Versailles - site d’Antony
6 mai 2019

DEGRÉ DE CONNAISSANCE PERSONNEL ESTIMÉ ET OPINION SUR L’ENSEIGNEMENT DE L’INFORMATIQUE À L’ÉCOLE

Comment estimez-vous votre degré de connaissance dans ces quatres domaines de la science informatique ?

	1 (bas)	2	3 (moyen)	4	5 (élevé)
Les machines informatiques	3	5	4	9	1
La pensée algorithmique	5	4	8	4	1
Les langages informatiques	8	6	5	3	
L’information numérique	5	3	9	3	2
TOTAL	21	18	26	19	4

Selon vous, faut-il enseigner la science informatique à l’école ?

Oui	20
Non	0
Je ne sais pas	2
Total	22

Le(s)quel(s) de ces quatres domaines de la science informatique vous semble(nt) abordable(s) et pertinent(s) pour les élèves de l’école élémentaire ?

Les machines informatiques	15
La pensée algorithmique	16
Les langages informatiques	15
L'information numérique	11

OPINION SUR L'UTILISATION DE SCENARIOS PEDAGOGIQUES PRODUITS PAR D'AUTRES

Vous arrive-t-il d'utiliser (éventuellement en les adaptant) des scénarios pédagogiques créés par d'autres pour mener vos séquences d'apprentissage ? Réponse ouverte

Oui	20
Non	2
Pas de réponse	
Total	0

Compléments de réponse spontané

- J'utilise des scénarios pédagogiques, mais en les adaptant (8)
- 2 participants mentionnent la source des scénarios pédagogiques qu'ils utilisent : *La Main à la Pâte* (1 participant), *CS unplugged* (1 participant)

Dans le cas précis de la mise en oeuvre d'un enseignement de l'informatique à l'école, la mise à disposition de scénarios pour supporter les enseignants vous paraît-elle nécessaire ? Réponse ouverte

Oui	22
Non	0
Pas de réponse	0
Total	22

Compléments de réponse spontané

- Nécessaire pour se lancer car mes connaissances actuelles sont insuffisantes (6 participants)

OPINION SUR LE SCÉNARIO PROPOSÉ

En tant qu'enseignant, trouvez-vous le scénario qui vous a été proposé motivant ? En quoi ?

Réponse ouverte

Oui	21
Non	1
Pas de réponse	0

Aspects motivants du scénario	Total
L'approche qui combine manipulation, questionnement, observation, hypothèses, validation	13
L'usage d'un robot est motivant en lui-même pour les élèves	4
La façon d'aborder les langages informatiques et le fonctionnement du robot	3
Le découpage du scénario en activités	2
Les activités favorisent la coopération	1

Le scénario proposé vous semble-t-il adapté aux élèves de cycle 3 de l'école élémentaire ?

Oui	22
Non	0
Pas de réponse	0

Remarques spontanées sur l'adéquation du scénario / niveau d'enseignement	Total
Le scénario pourrait aller plus loin	3
Le scénario est un peu difficile	1

**EFFORT ESTIMÉ ET OBSTACLES POUR S'APPROPRIER
ET METTRE EN OEUVRE LE SCÉNARIO****Comment estimatez-vous votre degré d'effort pour vous approprier ce scénario ?**

Degré d'effort estimé	Total
1 (faible)	1
2	2

3 (modéré)	17
4	0
5 (élevé)	2

Quels sont les principaux obstacles à votre appropriation de ce scénario ?

Réponse ouverte > regroupement des réponses par similarité

Obstacles à l'appropriation du scénario	Total
Aucun	12
La prise en main du matériel (robot)	5
Une des activités (“utiliser un langage restreint pour décrire le comportement d’Ozobot”) est trop compliquée	6
Trouver un moyen (projet) de l’Intégrer dans ma programmation	2

Comment estimatez-vous votre degré d'effort pour vous le mettre en oeuvre en classe ?

Degré d'effort estimé	Total
1 (faible)	1
2	3
3 (modéré)	15
4	1
5 (élevé)	2

Quels sont les principaux obstacles à la mise en oeuvre de ce scénario en classe ?

Réponse ouverte > Regroupement des réponses par similarité

Obstacles à la mise en oeuvre du scénario en classe	Total
Avoir le matériel à disposition	10
Organisation de la classe	4
Savoir utiliser la matériel	2
Fiabilité du matériel	2

Appropriation du scénario trop longue	2
Pré-requis des élèves	2
I'intégration dans des programmes déjà denses	1

PISTES D'AMÉLIORATION DU SCÉNARIO

Quels aspects du scénario proposé faudrait-il améliorer ?

Réponse ouverte > Regroupement des réponses par similarité

Pistes d'amélioration du scénario proposé	Total
Proposer ce scénario dans le cadre d'un projet concret avec un objectif explicite	5
Intégrer des défis à l'intérieur du scénario	4
Améliorer la séquence "utiliser un langage restreint pour décrire le comportement d'Ozobot"	4
Proposer des exemples compréhensibles par les élèves pour illustrer les notions difficiles (ex : notion de comportement aléatoire)	2
Proposer des supports de plus grande taille	2

Quels sont pour vous les caractéristiques d'un scénario pédagogique idéal ?

Réponse ouverte > Regroupement des réponses par similarité

Caractéristiques d'un scénario idéal	Total
Met les élèves en situation active de recherche et leur permet de construire leurs savoirs	7
Clarté des activités et de leur enchaînement, clarté des savoirs travaillés	5
S'articule autour d'objectifs de réalisation explicites et motivants pour les élèves	4
Donne du sens aux apprentissages (pourquoi on fait ça)	4
Ludique	2
Met les élèves en réussite dans leur apprentissage	1
Favorise la collaboration	1
Nécessite peu de matériel	1

Résumé

A l'heure où plusieurs pays ont introduit la science et la pensée informatique dans leurs programmes de la maternelle au lycée, le projet de recherche ANR IE-CARE (Informatique à l'Ecole – Conceptualisations, Accompagnement, Ressources) interroge les conditions de la mise en œuvre d'un enseignement de la science informatique sur les trois premiers cycles de l'école française. Mené dans le cadre de ce projet, le travail présenté concerne l'élaboration d'un modèle de scénarisation pédagogique servant de cadre à la production de scénarios pédagogiques pour l'enseignement de la science informatique à l'école dans une optique qui dépasse la pratique du codage et vise des conceptualisations dans le domaine de la science informatique.

.

.