
RETOUR D'EXPÉRIENCE DE FORMATION DES ENSEIGNANTS D'INFORMATIQUE : CONSTATS ET ENJEUX DIDACTIQUES

Sébastien JOLIVET¹

IUFE & TECFA, Université de Genève

Patrick WANG²

Haute École Pédagogique du canton de Vaud, Lausanne

Eva DECHAUX³

IUFE, Université de Genève

Résumé. Sur la base d'un retour d'expérience de trois formateurs d'enseignants en informatique en Suisse romande, nous présentons diverses questions soulevées en formation, d'une part en lien avec des aspects curriculaires et d'autre part en lien avec des questions d'enseignement. Pour chaque thème abordé, nous partons de constats qui sont problématisés et nous présentons quelques réponses travaillées en formation, avec, notamment, la mobilisation de différents apports de la didactique. Après avoir précisé différents éléments de contexte, nous abordons les aspects liés à l'introduction de la discipline scolaire informatique ; les conséquences de la diversité des rapports personnels des enseignants stagiaires à la discipline informatique et à son enseignement ; le travail sur les progressions annuelles et les ressources ; l'impact de l'organisation spatiale des classes ; puis nous terminons par des questions en lien avec l'évaluation.

Mots-clés. Didactique de l'informatique, formation d'enseignant, transposition didactique.

Abstract. This article reports on the authors' experience in training secondary computer science teachers in French-speaking Switzerland. We present issues commonly raised during our courses and originating either from curricular or teaching decisions and describe how we tackle them using several concepts drawn from didactics. After a brief description of the cultural context, we treat questions related to: the introduction of this new discipline in secondary education; the diversity of answers regarding what computer science is and what ought to be taught; the construction of an annual plan and design of educational resources; the spatial arrangement in the classroom and its effects on teaching practices; and finally, the several means of evaluating students' works in computer science.

Keywords. Computer science education, teacher training, didactic transposition.

¹ sebastien.jolivet@unige.ch

² patrick.wang@hepl.ch

³ eva.dechaux@unige.ch

Introduction

Dans Declercq (2021), l’auteur s’appuie sur son expérience de formateur en INSPÉ pour dresser un panorama de quelques travaux pouvant être utilisés en didactique de l’informatique dans une perspective de formation. Dans la conclusion, il évoque l’idée que « Pour constituer un corpus plus complet, il conviendrait maintenant de comparer au niveau national les références utilisées en formation d’enseignants d’informatique, et aussi au niveau de la communauté francophone, les hautes écoles pédagogiques en Suisse ou en Belgique ayant déjà une longue expérience de formation d’enseignants du secondaire dans un contexte un peu différent ». Nous proposons de contribuer à cet objectif.

En Suisse romande, l’informatique, en tant que discipline scolaire, a fait son apparition dans les curriculums en 2018, avec un enseignement prévu au degré secondaire (12 – 18 ans). Un besoin important de formation d’enseignants d’informatique s’est donc fait ressentir. Les auteurs de cet article sont directement impliqués dans cette formation dans deux institutions différentes. Cet article vise à rapporter quelques questions soulevées lors de ces formations ainsi que des pistes de réponse à ces questions. Pour cela, nous commençons par préciser des éléments de contexte sur la discipline informatique, son enseignement et la formation des enseignants dans les cantons de Vaud et de Genève, pour que le lecteur puisse identifier les similarités et différences avec d’autres contextes. Puis nous présentons quelques moyens utilisés et mis en œuvre en formation en réponse aux questions inhérentes à l’enseignement de l’informatique. Pour organiser notre propos, nous distinguons deux catégories de questions : celles liées à des aspects curriculaires (section 3) et celles liées à des questions d’enseignement (section 4).

1. Éléments de contexte

1.1 Contexte institutionnel et introduction de la discipline scolaire informatique

Le système éducatif en Suisse romande se décompose en deux degrés : le degré primaire (4 à 12 ans) et le degré secondaire (12 à 18 ans). Le secondaire est lui-même composé du secondaire I (12 à 15 ans) qui s’inscrit dans la continuité du primaire. À l’issue du secondaire I, les élèves sont orientés vers des formations générales ou professionnelles offertes au secondaire II (15 à 18 ans). Pour chaque niveau, un curriculum (nommé plan d’études) est défini au niveau cantonal mais dont les grandes lignes sont dictées au niveau intercantonal voire fédéral.

Des textes de référence (CDIP, 2017 ; CIIP, 2018) ont acté l’introduction de l’informatique comme discipline scolaire aux niveaux secondaire I et secondaire II à compter de la rentrée 2018, chaque canton devant réaliser une mise en œuvre effective au plus tard à la rentrée 2022. Si l’on peut trouver, avant 2018, des cours intitulés cours d’informatique dans certains plans d’études, il s’agissait en fait principalement de cours de bureautique. Ces changements ont induit un fort besoin de formation d’enseignants, en formation initiale et en formation continue.

1.2 Contexte relatif à la formation des enseignants

En Suisse romande, pour pouvoir enseigner une discipline il faut obtenir une certification qui permet ensuite de candidater dans les différents établissements scolaires. Dans le canton de Genève, une même certification permet d’enseigner dans le secondaire I ou le secondaire II à l’issue d’une formation de deux ans. Dans celui de Vaud, deux certifications distinctes sont délivrées à l’issue d’une formation de deux ans pour le secondaire I et d’une seule année pour le secondaire II. Cette certification est délivrée par les différentes institutions de formation mais est ensuite reconnue dans toute la Suisse, au niveau fédéral.

Pour pouvoir s'inscrire à la formation d'enseignant d'informatique il est nécessaire d'avoir un master en informatique, en humanités numériques, en systèmes de l'information ou une formation jugée équivalente. Par ailleurs, différents dispositifs sont proposés, selon le canton, aux enseignants d'une autre discipline, pour leur permettre de devenir aussi enseignant d'informatique tout en prenant en compte leur expérience professionnelle. Dans tous les cas, les futurs enseignants doivent attester de connaissances solides dans un ou plusieurs domaines de la science informatique. Les enseignants stagiaires ont des profils différents que l'on peut regrouper en trois grandes catégories : (1) des stagiaires qui font la formation d'enseignant dans la continuité de leur formation initiale avec ni expérience professionnelle ni expérience d'enseignement (ou limitée à quelques situations de suppléance) ; (2) des stagiaires qui sont déjà enseignants dans une ou plusieurs autres disciplines, avec souvent une expérience de l'enseignement de plusieurs années ; (3) des stagiaires qui ont plusieurs années d'expériences professionnelles dans des domaines proches de l'informatique et pour qui l'objectif de devenir enseignant est une reconversion professionnelle. Dans cette dernière catégorie, plusieurs motivent ce cheminement par une perte de sens dans leur métier et l'espoir d'en retrouver dans le cadre du métier d'enseignant.

À l'image de nombreuses formations d'enseignants, celles proposées dans les institutions des auteurs (IUFÉ à Genève et HEP Vaud à Lausanne), reposent sur une double articulation : articulation entre un volet théorique et un volet pratique organisée à l'aide de stages d'enseignement ; articulation, au sein de la formation théorique, entre enseignements dits transversaux et enseignements liés à la didactique de la discipline. Les auteurs de l'article interviennent dans la partie pratique, en observant les stagiaires dans leurs classes, et dans la formation à la didactique de la discipline.

Les enseignants en formation peuvent être affectés sur deux types de stages, parfois ils ne le sont que sur l'un, parfois sur les deux. Le premier est un stage dans lequel l'enseignant a la responsabilité de sa ou ses classes. Pour le second il est accueilli dans une ou plusieurs classes d'un enseignant expérimenté et il vit une alternance de situations d'observation, de co-intervention, de responsabilité supervisée et de responsabilité en autonomie. Ces modalités de stage peuvent exister dans un même groupe d'enseignants en formation et, quel que soit le type de stage, l'objectif est de mobiliser, *in-situ*, les apports proposés lors de la formation théorique, mais aussi de nourrir la formation avec un retour d'expérience et l'analyse de situations vécues.

1.3 Contexte relatif aux conditions d'enseignement de l'informatique

Les conditions matérielles d'enseignement de l'informatique sont plutôt favorables. D'une manière générale, le nombre d'élèves varie entre 6 et 24 par classe ou demi-classe, en fonction du niveau de classe et du canton. Les salles de cours sont, généralement, bien équipées, avec au moins un ordinateur par élève (PC, avec OS Windows ou Linux, ou Mac selon les établissements), un poste pour l'enseignant (parfois tactile), un vidéoprojecteur et parfois un système de pilotage des ordinateurs de la classe. Des différences significatives sont observées dans l'agencement des salles, leur effet est discuté plus en détail en section 4.4.

Dans le canton de Genève, dans les curriculums, l'informatique est présente uniquement en première année du secondaire 1, avec des éléments relatifs à la fois à la science informatique mais aussi d'autres qui relèvent plus largement de l'éducation numérique. Les différents éléments horaires sont synthétisés dans le Tableau 1 (une période est une unité de temps d'enseignement de 45 minutes). Le collège de Genève correspond au lycée général français. Les ECG (écoles de culture générale) proposent à la fois un enseignement général et une option préprofessionnelle.

Tableau 1 : nombre de périodes hebdomadaire d'informatique dans le canton de Genève

	1 ^{ère} année	2 ^{ème} année	3 ^{ème} année	4 ^{ème} année
Secondaire I (12 – 15 ans)	1	0	0	
Secondaire II Collège de Genève (15 – 18 ans)	2	1	1 4 en option complémentaire	0
Secondaire II ECG (15 – 18 ans)	2	2	0	

Ces différents éléments de contexte précisés, nous abordons maintenant divers constats, questionnements et hypothèses que nous avons pu construire sur la base de notre expérience. Les thèmes abordés en formation, et présentés dans la suite, sont la manifestation de questions curriculaires ou de questions d'enseignement. C'est à dire, comme synthétisé par Bosch (2020), des questions en lien avec la *transposition didactique externe* (problèmes curriculaires) et la *transposition didactique interne* (problèmes d'enseignement).

Après un rappel sur ces concepts nous structurons notre propos autour de ces deux processus.

2. Retour sur la notion de transposition didactique

Il y a couramment l'illusion d'une possible quasi-similarité entre le savoir issu des institutions productrices de savoirs (recherche universitaire ou industrielle par exemple), et le savoir à enseigner, puis enseigné. La place importante prise par des universitaires dans les processus ayant mené à l'introduction de l'informatique comme discipline scolaire et à la définition des premiers curriculums, ainsi que l'aspect relativement récent de ces processus, pèse sans doute dans une relative persistance de cette illusion pour la discipline informatique. La transposition des institutions productrices aux institutions d'enseignement est parfois pensée comme une simple question de découpages, de choix de traiter ou non, et de niveau d'approfondissement.

Le processus de transposition didactique, mis en évidence et étudié par Chevallard (1985), montre au contraire que le savoir ne peut pas être le même selon l'institution dans laquelle il est considéré. Lorsque le savoir « passe » d'une institution à une autre on parle de transposition, et, en particulier, lorsque l'institution cible est une institution d'enseignement on parle de *transposition didactique*. Dans notre cas, trois grands types d'institutions sont identifiables : les institutions productrices du savoir lié à la science informatique (universités, centres de recherche publics ou privés...) ; les institutions professionnelles dans lesquelles ce savoir est utilisé et qui sont aussi un lieu de production de savoirs ; les institutions dans lesquelles ce savoir est enseigné. Nous intégrons les institutions professionnelles en raison du nombre important d'enseignants en formation qui ont travaillé dans de telles institutions et de l'impact que cela a sur la formation, comme nous le développons plus loin. Chaque passage d'une institution à une autre implique une transposition du savoir (c'est aussi le cas entre deux institutions de même type).

La transposition didactique est séparée en deux phases selon les institutions et les acteurs impliqués dans le processus transpositif. Tout d'abord la *transposition didactique externe* durant laquelle le savoir à enseigner dans les institutions d'enseignement va être défini à partir du savoir de référence. Ce savoir à enseigner est celui que l'on peut observer dans les curriculums et dans les moyens d'enseignement (manuels, brochures, sites...). Les différents acteurs qui interviennent dans ce processus de définition du savoir à enseigner définissent une entité appelée

noosphère (voir le chapitre 1 de (Kaspary, 2020) pour plus de détails). Celle-ci est composée notamment des experts de la discipline, des décideurs de politique éducative, des professionnels qui vont employer les futurs élèves, des producteurs de moyens d'enseignement, etc., chacun tentant de faire valoir son point de vue lors du processus de définition du savoir à enseigner.

Dans un second temps, a lieu une transposition didactique interne. L'institution cible est la classe de l'enseignant, l'acteur principal du processus de transposition est l'enseignant lui-même. Lors de cette phase il va, à partir du savoir à enseigner de l'institution où il travaille, produire le savoir enseigné dans sa classe. Cette transposition didactique interne est le résultat de multiples choix et actions de l'enseignant : définition de sa progression ; sélection, production, adaptation de ressources (Gueudet & Trouche, 2010) ; orchestration (Drijvers et al., 2013) ; etc.

In fine, il y a le *savoir appris* par chaque élève que nous abordons brièvement dans la section 4.5 par le prisme de l'évaluation.

En formation notre objectif est double : 1) donner aux étudiants des moyens d'analyse et de compréhension des effets de ces choix globaux liés à la transposition didactique externe, sur lesquels ils ne peuvent agir directement ; 2) dans le cadre de ces choix, leur donner des moyens pour agir lors du processus de transposition didactique interne qu'ils vont réaliser dans leurs classes.

Ces deux objectifs motivent la structuration de ce texte. Ainsi, dans la section 3, nous présentons l'effet du résultat de la transposition didactique externe sur les conditions d'enseignement et donc sur les besoins de formation. Puis, la section 4 est consacrée à la présentation de divers éléments mobilisables lors de la transposition didactique interne et à leur place dans la formation.

3. Transposition didactique externe et effets sur la formation

L'étude approfondie de la transposition didactique externe des différents concepts de la science informatique nécessite un travail important (analyse épistémologique, analyse de nombreuses ressources, etc.) et relève d'une activité de recherche en didactique de l'informatique, et non pas d'une activité de formation. Cependant ce sont les fruits de ce processus qui définissent, en partie, le cadre d'exercice des enseignants stagiaires, et donc un certain nombre d'enjeux de formation. La présentation de ce processus dans le cadre de la formation est donc intéressante à différents titres. Tout d'abord, elle permet de préciser que le cadre dans lequel l'enseignant va exercer est un construit et pas un absolu. Ainsi, il peut évoluer au fil du temps, être imparfait ou incomplet, refléter divers enjeux sociétaux ou politiques. Il doit donc être analysé, compris et questionné en formation. De plus, étant le fruit de divers choix, les curriculums peuvent entrer en résonance ou en dissonance avec les conceptions et représentations des enseignants. Enfin, l'étude des curriculums produits peut mettre en évidence d'éventuels implicites, trous ou incohérences d'un point de vue didactique. Par conséquent, leur identification permet aux enseignants de mieux comprendre des difficultés d'enseignement et d'apprentissage.

Il est notable qu'en Suisse, comme les plans d'études sont d'un niveau de précision et de prescription assez faible, les établissements scolaires disposent d'une autonomie et d'une souplesse non négligeable dans leur mise en œuvre. Ceci est illustré dans (Dechaux & Jolivet, 2024) à propos du choix des langages de programmations et des moyens utilisés pour leur étude. Par exemple, pour un même niveau d'enseignement, plus de sept langages de programmation sont utilisés dans les différents établissements du canton de Genève. Ceci implique que le savoir à enseigner, d'un établissement à l'autre, est déjà re-précisé par rapport au curriculum et que la transposition didactique externe s'achève donc au sein de l'établissement. Cela a comme conséquence des différences, parfois significatives, dans les environnements de travail des stagiaires. Le prendre en compte est un défi supplémentaire en formation.

Nous revenons maintenant sur quelques éléments caractéristiques de la discipline informatique et du contexte Suisse qui impactent la transposition didactique externe.

3.1 Quel savoir de référence ? Quels curriculums produits ?

Avant de se poser la question de ce que les élèves doivent apprendre, il est nécessaire de définir le périmètre des savoirs de référence considérés. Concernant ce qui doit servir de fondement pour la discipline scolaire informatique, les approches sont diverses et se situent à différents niveaux. Une première question est de savoir s'il faut considérer uniquement la science informatique ou intégrer des éléments de culture numérique au sens large allant de l'utilisation d'outils, comme un navigateur web ou un logiciel de messagerie, à la bureautique. Ce niveau de discussion est par exemple présenté dans Fluckiger (2019). Même si on se limite à la *science informatique* il existe une diversité d'approches. Par exemple celle, souvent retenue dans l'enseignement primaire, de la pensée computationnelle proposée par Wing (2006) se centre plus sur des manières de penser et d'agir. Elle est assez différente de celle de Dowek (2011) qui propose de définir la science informatique par quatre domaines, chacun devant être étudié : les algorithmes, les machines, les langages et les données.

En Suisse, la société suisse pour l'informatique dans l'enseignement (SSIE) s'est réunie à de multiples reprises avec notamment la volonté de fournir « une base de réflexion pour aider les cantons romands à rédiger leur propre plan d'études » (SVIA et al., 2018, p. 7). Elle a proposé un plan d'études qui organise les concepts informatiques en découpant la science informatique en trois thématiques : 1. Algorithmique et programmation ; 2. Information et données ; 3. Systèmes numériques et réseaux. On y retrouve les quatre concepts du modèle de Dowek, dans un ordre et une composition différents, le domaine *langage* étant implicitement présent dans les trois thématiques.

Dans les faits, des arbitrages différents ont été réalisés selon les niveaux d'études. Ainsi dans le secondaire I, le plan d'études romand (CDIP, 2017) qui s'applique en Suisse romande et dans le Tessin définit une discipline intitulée « éducation numérique ». La figure 1 permet de constater que ce curriculum intègre non seulement le domaine de la science informatique mais aussi des éléments relatifs au numérique au sens large et notamment certains éléments relatifs à la bureautique dans la rubrique *usages* avec la création de contenus (le texte des secteurs circulaires est le même que celui des cadres). Historiquement, c'est quasi exclusivement cette partie qui était présente dans les cours intitulés « Informatique ».

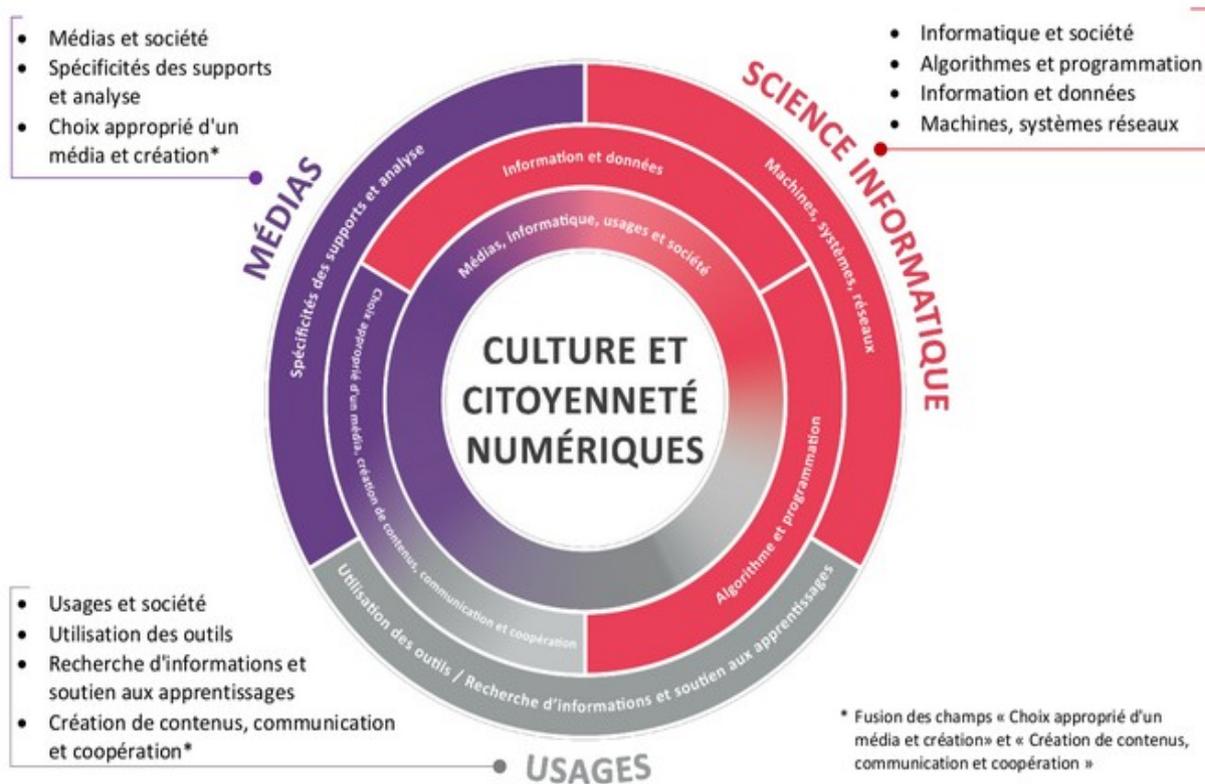


Figure 1 : Extrait des commentaires généraux sur l'éducation numérique pour le secondaire I.

Au niveau du secondaire II, dans les études gymnasiales (équivalent au lycée général en France) on constate que l'on se concentre sur la science informatique et qu'il y a un consensus sur les sujets incontournables. Dans le canton de Genève, le programme (DGES II, 2021, pp. 160-165) contient les thématiques « Information et données » ; « Algorithmique et programmation » ; « Sécurité » et « Bases de données ». Dans le canton de Vaud (DGEP, 2022) on trouve les thématiques « Représentation et structuration de l'information », « Architecture des ordinateurs », « Algorithmique », « Programmation » et « Réseaux ». Il est notable que dans le canton de Vaud s'ajoute, comme thématique à part entière, « Introduction aux enjeux sociaux », alors que dans le canton de Genève on trouve dans une rubrique *savoir-faire transversaux* la « Sensibilisation à différents aspects sécuritaires » et que la question de « l'impact écologique de l'informatique » est un sous-thème de *Informations et données*.

3.2 Quels effets de la transposition didactique externe pour la formation ?

Les évolutions curriculaires importantes mises en évidence dans la section précédente, avec notamment la place « historique » de la bureautique, ne sont pas sans conséquences sur les représentations de certains élèves et/ou enseignants de ce qu'est la science informatique et ce que doit être son enseignement. Cette diversité des représentations impacte aussi la convergence des enseignements proposés d'un enseignant à l'autre, parfois même au sein d'un établissement.

Une conséquence pour la formation est que l'on doit parfois accompagner un sentiment d'incompréhension des enseignants stagiaires vis-à-vis d'enseignants qui enseignent eux aussi l'informatique mais avec une vision différente de ce qu'est l'objet d'enseignement. La mise en évidence du processus de transposition externe et des évolutions curriculaires permet de donner aux étudiants des clés de compréhension de la situation et donc de l'appréhender, en partie au moins, plus sereinement et efficacement.

On peut en particulier revenir sur le fait que lors de la construction des curriculums, une fois le

savoir de référence défini, il est nécessaire d’opérer des choix permettant de définir les *savoirs à enseigner*. Ces choix sont guidés par un certain nombre de contraintes :

La cohérence des savoirs à enseigner pour permettre d’organiser une progression, une programmabilité de leur enseignement. Cette question est abordée dans la section 4.2.

La légitimité du savoir à enseigner dans le sens où il y a effectivement un enjeu d’apprentissage et qu’il présente une utilité.

La non-obsolescence du savoir à enseigner sur un plan moral (on ne va pas promouvoir des savoirs identifiés comme participant au réchauffement climatique) et scientifique (par exemple, sauf à se placer dans une perspective historique, on ne va pas étudier les disquettes comme moyen de stockage de l’information).

Les arbitrages, en termes de légitimité du savoir à enseigner, qui transparaissent dans les curriculums produits, sont parfois sujets à discussion avec les enseignants stagiaires. S’ils sont largement convaincus de la présence de la discipline “Informatique” en tant que discipline obligatoire pour les élèves, ce n’est pas toujours le cas pour la légitimité de certains contenus ou thématiques transversales comme « Informatique et société ». Ce type de discussion permet d’aborder avec les stagiaires le fait, qu’en tant qu’enseignants d’une discipline, ils n’en sont pas moins inscrits dans un système global et qu’ils doivent parfois « faire avec » le résultat d’arbitrages qui ne sont pas interne à la discipline. L’échelle des niveaux de codétermination du didactique (Chevallard, 2002) est un cadre intéressant pour aborder et formaliser cette réflexion, dans le sens où elle permet de penser les effets des différents niveaux, notamment *société, école, pédagogie, et discipline*, sur les conditions et contraintes qui déterminent le cadre dans lequel s’organise l’enseignement et donc sur l’étude didactique à mener.

Concernant l’obsolescence, l’entrée dans cette réflexion par les stagiaires est souvent liée aux évolutions technologiques plus qu’à l’identification d’éventuelles ruptures conceptuelles. Or si ces dernières peuvent amener à réinterroger en profondeur les savoirs à enseigner, il n’y a pas de raison de se livrer à une course à la nouveauté technologique systématique. Si les illustrations proposées pour différents concepts se doivent d’être réalistes par rapport aux évolutions techniques (par exemple les ordres de grandeur pour le stockage de l’information), il n’en demeure pas moins que les fondements du stockage et de la représentation des données sont stables. Ou encore, le passage du protocole IPv4 à IPv6 modifie-t-il les concepts à travailler lors d’une introduction aux réseaux ? Il est nécessaire d’amener les enseignants stagiaires à avoir une réflexion sur les délimitations et l’intrication entre les avancées technologiques et conceptuelles, pour éclairer leurs choix mais aussi pour leur éviter un sentiment d’instabilité permanente du contenu de leurs cours. Sentiment qui est parfois encore plus fort chez les enseignants imprégnés d’une culture professionnelle qui ont pu être confrontés à des logiques de choix différentes.

Comme nous l’avons mentionné en section 3, les établissements disposent d’une grande liberté dans les choix réalisés pour la mise en œuvre des plans d’études du fait de leurs contenus parfois très succincts et se limitant à la définition de grandes thématiques sans spécifier ni d’éléments précis de contenus, ni de niveaux d’approfondissement des thèmes. Cette liberté implique que les enseignants stagiaires doivent pouvoir s’insérer dans des environnements très variés. Par exemple et selon les établissements, les élèves du secondaire II pourront commencer la programmation avec des langages de programmation par blocs (avec par exemple Scratch, Blockly ou encore Thymio VPL) ou directement avec des langages de programmation textuels (le plus souvent Python ou Javascript). Il est donc nécessaire, en formation, d’intégrer cette diversité importante des expériences vécues, et parfois subies, par les stagiaires. L’absence d’examen terminal national ou cantonal, favorise encore cette dynamique de différenciation inter-établissements.

Nous abordons maintenant les moyens pour agir lors du processus de transposition didactique

interne qu'ils vont réaliser dans leurs classes.

4. Enjeux de formation liés à la transposition didactique interne

4.1 Rapport personnel de l'enseignant à la discipline informatique et stéréotypes autour de l'informatique

Lors de sa formation initiale, puis éventuellement lors de ses expériences professionnelles, l'enseignant stagiaire a construit, au travers des différentes institutions fréquentées, un rapport personnel (Chevallard, 2003) à la discipline informatique, à son enseignement voire à l'enseignement en général. Les échanges et observations nous ont permis de dresser les constats suivants sur des éléments ayant participé à la construction de ces rapports personnels.

Les enseignants stagiaires n'ont pas vécu *l'informatique discipline scolaire* en tant qu'élèves. Ils n'ont comme référence d'enseignements vécus à l'école que ceux d'autres disciplines.

Si tous les stagiaires ont une formation initiale validée par l'institution comme suffisante pour intégrer la formation d'enseignant en informatique, elle peut être très différente d'un étudiant à l'autre. D'une part il y a ceux qui ont étudié l'informatique comme objet d'étude et ceux pour qui l'informatique a plutôt été un outil dans le cadre de leur formation initiale. D'autre part les contenus des cursus universitaires suivis peuvent être significativement différents.

De nombreux enseignants stagiaires ont une ou plusieurs expériences professionnelles dans le milieu de l'informatique (développeur, ingénieur réseau...). Ils ont ainsi déjà vécu le fruit d'une transposition entre leur(s) institution(s) de formation et leur(s) institution(s) professionnelle(s). Partant d'un même savoir de référence cette transposition professionnelle peut donner lieu à des résultats nettement différents quand il s'agit d'une transposition didactique.

Un certain nombre de stéréotypes relatifs à l'informatique traversent la société, et sont donc potentiellement partagés, voire véhiculés, par les élèves et/ou les enseignants. Ainsi, par exemple, l'image de l'informaticien ou du geek, homme au teint pâle travaillant (ou jouant) seul devant son écran et imposant, par son habileté, sa volonté à sa machine (et potentiellement au-delà) peut représenter un attrait pour certains élèves mais aussi être répulsive pour d'autres. Du côté des enseignants, une idée répandue est que les élèves sont des *digital natives* avec un rapport bien établi à l'artefact ordinateur.

Nous associons ces constats à différentes problématiques de formation.

Des enseignants ne se sentent pas qualifiés pour enseigner certains aspects du curriculum et/ou ne perçoivent pas la légitimité de certains thèmes, tout du moins dans le cadre d'un cours d'informatique. On peut constater des difficultés pour certains enseignants à former leurs élèves à la citoyenneté numérique par exemple, ou encore aux bonnes pratiques en termes de sécurité informatique. Ainsi, les résultats d'une enquête menée auprès de 25 enseignants, en poste dans une autre discipline et suivant une formation continue pour enseigner l'informatique, montrent que s'ils sont majoritairement convaincus que la thématique « Informatique et société » doit être enseignée au secondaire II, ils sont beaucoup plus partagés sur le fait que cet enseignement soit fait par des enseignants d'informatique. La discussion qui a immédiatement suivie cette enquête indique que ces enseignants ne se sentent pas capables de mener un tel enseignement du fait d'un manque de formation sur cette thématique.

L'absence de « vécu d'élève » par rapport à la discipline informatique a plusieurs

conséquences :

Une première rencontre avec la matière au niveau universitaire, et de manière « choisie », provoque parfois l'illusion que l'informatique est une matière *a priori* intéressante pour la totalité des élèves.

Il n'y a pas de vécu du résultat d'une transposition didactique dans une institution d'enseignement scolaire qui permet de matérialiser l'écart entre les savoirs universitaires et/ou professionnels qui sont les seuls auxquels l'étudiant a été confronté.

Enfin, cette absence de vécu en position d'élève, amplifie encore la difficulté, assez fréquente chez les enseignants « anciens bons élèves » d'une manière générale, à anticiper les difficultés que peut rencontrer un public scolaire par rapport à cette discipline. Cela n'aide pas les enseignants stagiaires à identifier *a priori*, ou hiérarchiser, des éléments du savoir à travailler comme pouvant être des obstacles.

L'imprégnation par une culture professionnelle implique que de nombreux implicites se sont installés dans leur rapport à la discipline à enseigner. Ceci concerne à la fois les aspects conceptuels de leur domaine d'expertise, mais aussi les aspects liés à l'utilisation d'un environnement technique (par exemple, les environnements de développement intégrés). Il en résulte que certains objets d'enseignement, ou l'effort d'appropriation nécessaire de certains environnements, semblent transparents ou naturalisés (Coulange & Robert, 2015) pour certains enseignants et ne peuvent donc être identifiés comme source de difficultés par les élèves.

Les enseignants et les élèves ne sont pas étanches aux stéréotypes évoqués plus haut. Ceci peut influencer à la fois les contenus disciplinaires abordés, les modalités d'apprentissage proposées, ainsi qu'un engagement des élèves et des dynamiques de classe fortement marqués par le genre (Centre Hubertine Auclert, 2022). Des *a priori* existent également sur la manière de motiver les élèves : pour beaucoup d'enseignants d'informatique créer un jeu vidéo est stimulant pour tous les élèves, ce qui n'est pas vrai. De même l'image de l'élève *digital native*, conduit à surestimer les habiletés techniques avec l'ordinateur et, corrélativement, à sous-estimer une nécessaire formation. Or, s'ils ont un rapport au numérique (Fluckiger, 2019), pour de nombreux élèves celui-ci a principalement lieu avec leurs smartphones ou tablettes tactiles. L'identification et l'utilisation de matériels aussi basiques que clavier, souris, écran, unité centrale, ne sont pas des évidences pour de nombreux élèves. Il en est de même pour des utilisations « basiques » d'un ordinateur (sauvegarder ou trouver un fichier...).

L'identification de ces problématiques nous a conduit à mettre en place les actions suivantes dans le cadre de la formation et/ou du suivi de stage.

Faire prendre conscience aux enseignants des différences entre le savoir de référence et le savoir à enseigner, mais aussi avec le savoir professionnel. En particulier, ce qui peut être un critère de choix pertinent dans le monde professionnel ne l'est pas forcément pour l'enseignement.

Redonner une visibilité aux difficultés épistémologiques et techniques qui ont été masquées par l'expérience professionnelle, ou la représentation des élèves *digital natives*, et mettre en évidence la nécessité d'en faire des objets d'apprentissage. Il s'agit de souligner une forme d'amnésie du spécialiste.

Insister sur le fait que l'informatique est une matière comme une autre et qu'il n'y a pas de raison qu'elle rencontre une adhésion *a priori* des élèves.

Proposer des ressources, notamment d'auto-formation, relatives à certains sujets du plan d'études comme la culture numérique ou les impacts environnementaux du numérique mais absents des formations initiales des stagiaires. À titre d'exemple on peut signaler la plateforme Modulo (Petreska von Ritter et al., 2023), développée dans le canton de Vaud.

Inciter les étudiants à s'appuyer sur diverses ressources, qu'elles soient de type MOOC ou intervenants spécialistes d'un thème comme cela peut être possible lors d'évènements particuliers (semaine des Médias par exemple).

À l'occasion du travail spécifique sur les plans d'études, une réflexion spécifique peut aussi être réalisée sur diverses modalités permettant d'intégrer ces thèmes dans les plans de cheminement des enseignants, c'est-à-dire l'organisation temporelle et conceptuelle de l'enseignement.

4.2 Du plan d'études au plan de cheminement, un travail important

Comme nous l'avons mis en évidence dans la section 3, la transposition didactique externe produit des curriculums qui définissent pour l'enseignant un cadre global dans lequel il met en œuvre son enseignement. Si les curriculums doivent notamment assurer une programmabilité des savoirs à enseigner, entre le curriculum (en particulier le plan d'étude) et l'organisation de l'enseignement sur une année scolaire (souvent appelé plan de cheminement), il y a un travail important. Or nous avons fait le constat que ce travail est globalement peu réalisé par les enseignants stagiaires, qui se contentent souvent de partager équitablement le temps annuel en autant de séquences que de grands thèmes du plan d'études et de planifier ces séquences dans l'ordre de rédaction du plan d'études. Divers constats expliquent que ce travail de production du plan de cheminement soit aussi délicat, bien que fortement nécessaire.

S'il existe une certaine liberté au niveau des établissements (voir section 3.2), ceux-ci produisent aussi des contraintes internes comme la planification et la définition de contenus pour des évaluations communes, voire de plans de cheminement qui s'imposent à l'ensemble des enseignants de la discipline dans l'établissement.

Le manque de repères pour l'enseignant lui permettant d'identifier *a priori* les difficultés majeures du plan d'études (voir section 3.1), ne l'aide pas à analyser les rapports (temps relatif, prérequis, articulations...) entre les différents thèmes du plan d'études.

Les ressources disponibles sont rares et souvent locales du fait de l'absence d'industrie du manuel scolaire en Suisse. Combiné à la nouveauté de la discipline, cela conduit à un manque de recul vis-à-vis des plans de cheminements conçus.

À nouveau nous relierons ces constats à diverses problématiques explorées en formation.

Les choix réalisés au niveau de l'établissement peuvent générer des tensions entre les enseignants d'informatique « informaticiens » (les stagiaires notamment) et enseignants d'informatique « non informaticiens ».

Les choix réalisés dans les établissements, notamment des langages de programmation ou des environnements de développement, ne sont pas nécessairement guidés par des considérations didactiques, mais parfois plus par les goûts personnels de certains enseignants et/ou des contraintes techniques internes à l'établissement.

La faible quantité de moyens d'enseignements disponibles oblige souvent les enseignants à partir d'une quasi-feuille blanche pour construire leur plan de cheminement et conduit souvent à des résultats très proches du plan d'études. La charge de travail que cela représente et la faiblesse des ressources amènent souvent les enseignants à opter pour des solutions de simplicité en s'appropriant rapidement la première ressource trouvée.

Face à ces problématiques nous mettons en place à la fois des réponses pratiques et l'introduction de certains concepts pour accompagner le travail à réaliser.

Proposer aux étudiants un travail spécifique d'étude du curriculum, en partant d'un niveau de granularité élevé vers un niveau de détail important, pour faire émerger des relations entre les objets de savoir à travailler. Ce travail peut déboucher sur une organisation du plan de cheminement moins centrée sur des blocs monolithique, par exemple en traitant certains points plus au fil de l'eau. A minima il améliore la capacité des enseignants à réagir à la réalité de la classe en évaluant mieux le poids respectif des différentes notions et les liens qui peuvent être faits entre elles. Ce travail d'analyse permet aussi d'identifier certains éléments qui ne figurent pas directement dans le plan d'études mais qu'il est essentiel de faire apparaître dans son plan de cheminement.

Introduire en formation la dialectique outil-objet au sens de Douady (1986), permettant aux étudiants d'envisager plusieurs rencontres d'un même objet de savoir et donc de repenser leur plan de cheminement. Cela permet aussi de travailler aussi sur les raisons d'être de certains objets. Par exemple, des fonctions déjà définies vont pouvoir être utilisées comme outil lors des premiers travaux de programmation, avant de devenir objet d'étude lorsqu'il va s'agir d'en concevoir de nouvelles, pour ensuite reprendre le statut d'outil permettant de concevoir ou améliorer certains programmes plus complexes.

Mettre à disposition des ressources diverses. En effet, les moyens d'enseignement, du type manuels scolaires ou brochures, proposent un premier niveau d'interprétation du curriculum (ordre des chapitres, choix de certains vocabulaires, liens effectués entre différentes notions, niveau d'approfondissement...) et constituent des supports intéressants dans la mesure où ils peuvent élargir le champ des possibles pour l'enseignant. De plus, la diversité des interprétations présentes dans les différentes ressources génère un questionnement, laissé à la charge des enseignants, sur les raisons de ces différences.

4.3 Du plan de cheminement aux ressources pour faire travailler les élèves

Une fois le plan de cheminement produit, un travail important de l'enseignant est de chercher, sélectionner, produire, adapter, des ressources pertinentes pour sa mise en œuvre. À nouveau, plusieurs éléments sont à prendre en compte en formation pour aider et former les stagiaires.

La faible quantité de moyens d'enseignements disponibles et le faible niveau d'indications dans les curriculums ont déjà été évoquées dans les sections précédentes.

Le côté récent de la discipline fait qu'il n'y a pas de forte culture partagée d'établissement, voire parfois au contraire des tensions dans l'interprétation de ce qu'il faut faire.

Enfin, faire travailler les élèves en informatique c'est non seulement « faire de l'informatique » mais aussi travailler des « façons de faire de l'informatique ».

Ces éléments ont des conséquences directes sur l'activité des enseignants en formation, qui soulèvent notamment les problématiques suivantes.

Tout comme pour la construction des plans de cheminement le peu de matériaux disponibles pour comparer différentes ressources conduit parfois à adopter sans recul ce qui est disponible ou à produire ses propres ressources à partir d'une feuille blanche.

Les enseignants, notamment ceux issus du monde professionnel, ont tendance à oublier que certaines façons de faire usuelles en informatique ne font pas partie de la culture (scolaire) des élèves et ne sont pas nécessairement identifiés dans les plans d'études.

C'est notamment le cas lorsqu'il s'agit de travailler en groupe sur un projet informatique, qui nécessite à la fois de travailler la gestion de projet et potentiellement l'utilisation d'outils spécifiques (par exemple Git, un système décentralisé de gestion de versions de documents, par exemple des programmes).

Pour répondre à ces difficultés nous mettons en place les éléments suivants en formation.

Tout d'abord il s'agit de donner aux étudiants des moyens d'analyser, adapter ou produire leurs propres ressources. À la fois, comme précisé en section 4.2, en mettant à disposition une diversité de ressources, mais aussi en leur présentant divers concepts issus de la didactique d'autres disciplines (Jolivet et al., 2022) pour les outiller pour ce travail de production de ressources.

Travailler la notion de contrat didactique (Brousseau, 1986) et en particulier souligner qu'un certain nombre d'éléments du contrat qu'ils souhaitent construire (il est autorisé, voire attendu, d'utiliser l'aide des logiciels, de chercher des ressources de type tutoriel en ligne, d'utiliser une documentation, de prendre pour ne pas refaire...) peuvent être significativement différents des pratiques usuelles dominantes.

Participer, en tant que formateurs, à la construction de ressources d'enseignement mises à la disposition de la communauté enseignante (Petreska von Ritter et al., 2023).

Favoriser une culture de mutualisation et de partage entre enseignants, par exemple avec des modalités de formation permettant de faire vivre aux étudiants l'apport du partage.

4.4 Effets de l'organisation spatiale des salles de classe

Le contexte matériel favorable, présenté dans la section 1.3, évite d'avoir à gérer en formation des questionnements du type « comment faire lorsqu'il n'y a qu'un ordinateur qui fonctionne pour 2 ou 3 élèves ? » *A contrario*, il peut freiner l'émergence spontanée de certaines questions relatives à l'alternance entre des activités avec et sans ordinateurs ou encore, la pertinence selon la situation et les objectifs d'organiser des situations de collaboration entre élèves et de ne pas les cantonner à un face à face permanent avec leur ordinateur.

Lors de l'observation en classe des stagiaires, nous avons pu constater les éléments suivants :

Une part importante du temps des élèves se déroule avec utilisation de l'ordinateur, en particulier au secondaire I où les rares exceptions sont par exemple pour présenter les différents composants de l'ordinateur. Au secondaire II, certains éléments du plan d'études impliquent davantage de moments sans ordinateur.

L'organisation spatiale des salles (figure 2) induit des différences de pratiques. Cependant, en l'absence d'une organisation spatiale affordable, certaines pratiques ne sont pas ou peu explorées.

S'il y a présence systématique d'un vidéoprojecteur, avec une surface de projection pas toujours très ergonomique, celle d'un tableau est rare.

Les questions de formation que cela soulève sont en particulier les trois suivantes.

Comment faire que la ou les pratiques ne soient pas largement prédéterminées par l'organisation spatiale des classes et le mobilier disponible ?

Comment mettre dans le champ de réflexion des stagiaires le fait que l'utilisation systématique de l'ordinateur n'est pas une évidence ?

Comment alterner, y compris au sein d'une même séance, diverses modalités : avec ou sans l'ordinateur ; avec des rôles différents pour l'ordinateur ; travail individuel, en groupe ou collectif ; etc. ?

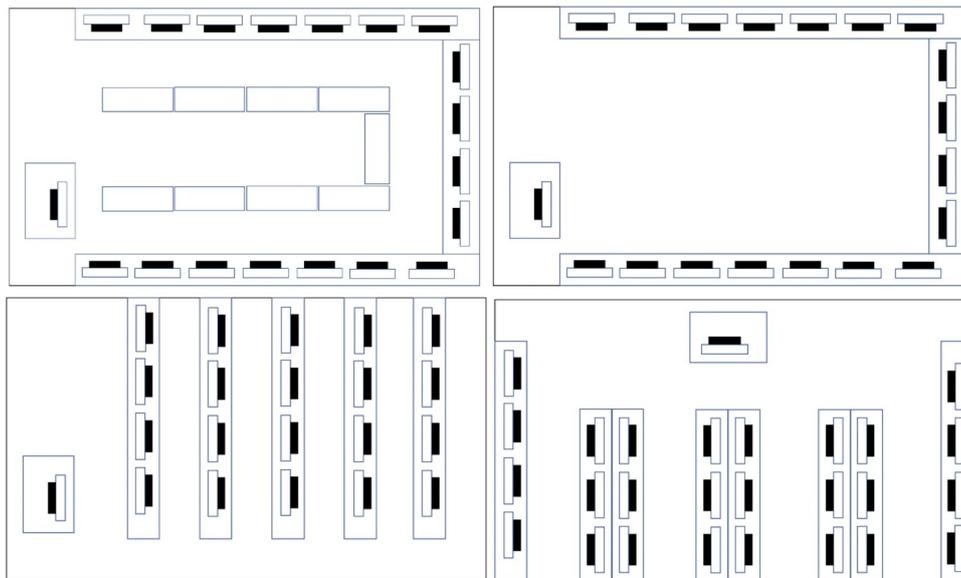


Figure 2 : Illustration des principales organisations observées, le nombre de postes variant de 12 à 24.

En formation la réponse s’oriente dans trois directions.

Organiser le partage de pratiques différentes et proposer des pratiques parfois usuelles dans le monde professionnel (*pair programming*). Un des enjeux est alors de réfléchir sur l’adaptation de ces pratiques dans différentes configurations matérielles. À titre d’exemple, nous avons pu observer des modalités qui favorisent les situations de formulation entre élèves et d’autres qui dissocient, dans une même activité, des temps de recherche sans ordinateur et des temps de vérification - validation à l’aide de l’ordinateur. Pour la première il s’agit d’une situation où deux élèves disposent de deux ordinateurs mais chacun a une fonction particulière : programmer sur l’un - chercher sur l’autre. Le programmeur doit formuler des demandes précises au chercheur, qui doit en retour formuler des aides précises aussi. Pour la seconde, un enseignant avait conçu une situation lors de laquelle les élèves devaient réaliser un travail d’analyse de différents morceaux de code sur papier, mais disposaient de différents ordinateurs sur lesquels les morceaux de code étaient implémentés pour pouvoir tester leurs hypothèses.

Proposer des activités d’informatique débranchée (par exemple tirées ou inspirées de Bell & al. (2002) ou Vincent (2017)). Un enjeu fort étant alors de travailler sur la création de liens et d’articulations entre les situations débranchées et branchées. Il est à noter que dans ce domaine, comme pour beaucoup d’autres questions, il y a relativement peu de travaux qui permettent d’avoir des « certitudes » sur les apports respectifs des approches branchées et débranchées. On peut cependant signaler l’article de Sigayret et al. (2022) autour d’enjeux liés à la pensée computationnelle.

Mettre en évidence, qu’au-delà des facilités induites par l’organisation spatiale de la salle, diverses modalités et pratiques sont possibles et présentent divers intérêts. Le lecteur intéressé par l’impact de l’organisation spatiale sur l’activité en classe peut se référer au travail de Lermigeaux-Sarrade (2018) et aux références qu’il contient.

4.5 La question de l’évaluation en informatique

Les moments didactiques, tels qu’introduits par Chevillard (1999), sont un moyen intéressant pour aider les stagiaires à penser la diversité des tâches à réaliser pour étudier une notion. Parmi ces moments, celui de l’évaluation est vécu par les étudiants comme posant un problème

particulier. Si cette question soulève des questions indépendantes de la discipline (finalité, dimension sociale, équité et égalité...), nous abordons ici les aspects liés à l'informatique. Nos observations et questions soulevées par les stagiaires nous amènent aux constats suivants.

Les activités pratiques font assez majoritairement l'usage de machines, mais les évaluations, et en particulier les évaluations groupées et communes à plusieurs classes d'un même établissement, se font sur papier.

Le contrat didactique peut se trouver rompu lorsque l'enseignant encourage l'utilisation de ressources externes pour chercher des informations puis les interdit lors des évaluations.

La mise en place de modalités d'enseignement du type « projet à réaliser en groupe » amène à des interrogations sur les manières d'évaluer individuellement dans un projet collectif.

Est-il raisonnable de donner du travail hors la classe et de l'évaluer, notamment en raison de l'accès inégal des élèves à un ordinateur ?

Il est dès lors utile de traiter des questions suivantes en contexte de formation :

Est-ce la même chose de traiter une tâche donnée sur papier ou sur machine ? Les savoirs mobilisés sont-ils effectivement les mêmes ? S'entraîner dans un environnement permet-il de réussir dans l'autre ?

Une question connexe est de savoir si on peut évaluer un travail sur machine ? Et si oui, comment ? Le « bon fonctionnement » est-il un critère suffisant de validation ? Le non-fonctionnement est-il nécessairement un critère d'échec ou de pénalisation ?

Peut-on donner et évaluer des devoirs à la maison, sachant que tous les élèves n'ont pas un accès égal à un ordinateur chez eux et que cela risque de creuser les inégalités sociales ?

Comment permettre l'utilisation de ressources externes dans le cadre des évaluations ?

Plusieurs pistes sont discutées et explorées en formation proposées pour proposer des éléments de réponse à ces questions :

Le travail sur machine, avec un rendu limité à la version finale, peut masquer le processus de résolution d'un exercice. Il est donc important de pouvoir récolter des informations aussi bien sur étapes du travail que sur le résultat final. Diverses solutions permettent de répondre à ce besoin, par exemple, un programme peut être sauvegardé et versionné à chaque exécution. Il est aussi important de ne pas penser la machine uniquement comme une contrainte, le fait de pouvoir tester effectivement un programme peut amener l'élève à gérer des erreurs qu'il n'aurait pas nécessairement détecté sur papier, car dans l'impossibilité de tester.

Des attentes formalisées sur les commentaires dans le code permettent à la fois de développer de bonnes pratiques et de fournir une matière de compréhension, et d'évaluation, pour l'enseignant.

Si les évaluations ne peuvent pas se faire sur machine, il devient d'autant plus important que les élèves travaillent sur des activités débranchées afin de respecter l'alignement pédagogique entre les activités réalisées en classe et les évaluations proposées.

Pour les devoirs à la maison, ceux-ci peuvent viser des objectifs (Bloom et al., 1984) du niveau compréhension (en proposant des exercices de lecture de programmes et de traçage de codes) ou évaluation (en inspectant un programme pour y déceler des erreurs).

5. Conclusion

Dans cet article nous avons, en nous appuyant sur le concept de transposition didactique, apporté des éléments de réponse à notre double objectif de formation : 1) donner aux étudiants des moyens d'analyse et de compréhension des effets liés à la transposition didactique externe, sur lesquels ils ne peuvent agir directement ; 2) dans ce contexte, leur donner des moyens pour agir lors du processus de transposition didactique interne qu'ils vont réaliser dans leurs classes. Pour chacune, nous évoquons ponctuellement à quelles occasions mobiliser différents apports de la didactique.

Ainsi, sur la base d'un retour d'expérience, nous avons poursuivi la réflexion engagée par Declercq (2021), sur les outils et concepts de la didactique que l'on peut mobiliser en formation des enseignants d'informatique. Par rapport aux quatre axes qui structurent le texte de Declercq ; compétences de la pensée informatique, apports de la psychologie de la programmation, cadres issus de la didactique des mathématiques et approche instrumentale ; nos propositions s'inscrivent essentiellement dans les deux derniers. Approfondir les deux autres est un prolongement possible de ce travail.

Au-delà des réponses présentées dans cet article, fondées essentiellement sur une approche empirique, il nous semble que l'on peut dégager quelques grands enjeux à adresser à la fois aux formateurs d'enseignants d'informatique et aux chercheurs en didactique. Le premier est le développement de communautés et de liens entre celles-ci, par exemple la SSIE en Suisse, la communauté d'enseignants de SNT/NSI en France ou la communauté CAI (Bachy et al., 2022). Pour le moment les liens entre ces communautés sont quasi inexistantes, ce qui limite la mutualisation et la co-construction de connaissances et de ressources, aussi bien dans le contexte de la formation des enseignants que dans l'enseignement même de l'informatique.

Parallèlement, le développement de travaux de recherche en didactique de l'informatique, domaine encore largement en construction, est un enjeu important pour ces communautés. Cela participera à la réponse à plusieurs besoins : création de liens entre les communautés au travers de travaux interinstitutionnels ; étayer et compléter les apports pouvant être réalisés en formation ; donner une visibilité et une légitimité en tant que domaine de recherche à part entière. Nous terminons en soulignant une question parmi d'autres que pourraient interroger ces communautés : *quels rapports entre l'informatique et les autres disciplines scolaires : nécessité ou liaison dangereuse ?* Si cela peut inciter à une certaine prudence il ne faut pas non plus occulter que le travail en informatique avec des travaux dans d'autres disciplines permet, notamment, de faire émerger et vivre des raisons d'être de certains savoirs.

Remerciements

Les auteurs remercient Micha Hersch et Biljana Petreska von Ritter (HEP Vaud) pour les échanges qui ont précédé la rédaction de cet article.

Références

- Bachy, S., Corieri, P., Goletti, O. O., Hoarau, S. S., Komis, V. K., Massart, T., Mens, K., Parriaux, G., Rafalska, M., & Romero, M. (2022). *Communauté d'Apprentissage de l'Informatique*. Didapro 9, Le Mans.
- Bell, T., Witten, I. H., & Fellows, M. (2002). Computer science unplugged. *Off-line activities and games for all ages (draft)*.
- Bloom, B. S., Engelhart, M. D., Furst, E. J., Hill, W. H., & Krathwohl, D. R. (1984). *Taxonomy of educational objectives : Handbook 1 : Cognitive domain*. Longman Publishing Group.

- Bosch, M. (2020). *Didactic transposition : A tool for analysis*. ICMI Awardees Multimedia Online Resources. <https://youtu.be/QQIK4PN6xpM>
- Brousseau, G. (1986). Fondements et méthodes de la didactique des mathématiques. *Recherches en didactique des mathématiques*, 7(2), 33-115.
- CDIP. (2017). *Plan d'études cadre pour les écoles de maturité : Informatique*. https://edudoc.ch/record/131918/files/rlp_inf_2017_f.pdf
- Centre Hubertine Auclert. (2022). *Les freins à l'accès des filles aux filières informatiques et numériques*. <https://m.centre-hubertine-auclert.fr/article/publication-de-l-etude-les-freins-a-l-acces-des-filles-aux-filieres-informatiques>
- Chevallard, Y. (1985). *La transposition didactique* (Vol. 95). Grenoble : La pensée sauvage.
- Chevallard, Y. (1999). L'analyse des pratiques enseignantes en théorie anthropologique du didactique. *Recherches en Didactique des Mathématiques*, 19(2), 221-265.
- Chevallard, Y. (2002). Organiser l'étude. 3. Ecologie & régulation. In J.-L. Dorier, M. Artaud, M. Artigue, R. Berthelot, & R. Floris (Éds.), *Actes de la XIe école d'été de didactique des mathématiques* (p. 41-56). La Pensée Sauvage, Grenoble. http://yves.chevallard.free.fr/spip/spip/IMG/pdf/Organiser_1_etude_3.pdf
- Chevallard, Y. (2003). Approche anthropologique du rapport au savoir et didactique des mathématiques. In S. Maury & M. Caillot (Éds.), *Rapport au savoir et didactiques* (p. 81-104). Editions Fabert.
- CIIP. (2018). *Plan d'études romand*. <https://www.plandetudes.ch/per>
- Coulanges, L., & Robert, A. (2015). Les mathématiques dans les activités du professeur—Conséquences pour la formation. In L. Theis (Éd.), *Pluralités culturelles et universalité des mathématiques : Enjeux et perspectives pour leur enseignement et leur apprentissage —Actes du colloque EMF 2015—GT1* (p. 81-94)
- Dechaux, E., & Jolivet, S. (2024). *Apprentissage de la programmation, contexte numérique des enseignants d'informatique genevois*. Didapro 10 - DidaSTIC, Louvain-la-Neuve.
- Declercq, C. (2021). *Didactique de l'informatique : Une formation nécessaire*. 28(3). <https://doi.org/10.23709/STICEF.28.3.8>
- DGEP. (2022). *Ecole de maturité. Plan d'études et liste des examens*. (p. 195-201) [Programmes de la filière Gymnasiale]. https://www.vd.ch/fileadmin/user_upload/organisation/dfj/dgep/dgep_fichiers_pdf/DGEP_brochure_EM_web.pdf
- DGES II. (2021). *Programmes des disciplines enseignées dans la filière gymnasiale au Collège de Genève valable pour les élèves scolarisés dès 2021* (p. 160-165) [Programmes de la filière Gymnasiale]. <https://www.ge.ch/document/programmes-disciplines-enseignees-dans-filiere-gymnasiale-au-college-geneve-valable-eleves-scolarises-2021>
- Douady, R. (1986). Jeux de cadres et dialectique outil-objet. *Recherches en Didactique des Mathématiques*, 7(2).
- 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*, 21-29.
- Drijvers, P., Tacoma, S., Besamusca, A., Doorman, M., & Boon, P. (2013). Digital resources inviting changes in mid-adopting teachers' practices and orchestrations. *ZDM*, 45(7),

987-1001.

- Fluckiger, C. (2019). *Une approche didactique de l'informatique scolaire*. Presses universitaires de Rennes.
- Gueudet, G., & Trouche, L. (2010). *Des ressources aux documents, travail du professeur et genres documentaires*. Presses Universitaires de Rennes et INRP.
- Jolivet, S., Wang, P., & Dechaux, E. (2022). Formation didactique des enseignants d'informatique : Quels concepts utiliser ? *Actes du colloque DidaPro9 - DidaSTIC*. DidaPro9 - DidaSTIC, Le Mans.
- Kaspar, D. (2020). *La noosphère, un lieu de tension pour le curriculum : Étude didactique de la mise en place d'un système d'évaluation de manuels scolaires sur l'étude du champ additif à l'école primaire* [PhD Thesis, Université Grenoble Alpes ; Universidade federal de Mato Grosso].
- Lermigeaux-Sarrade, I. (2018). *Rôle de l'organisation de l'espace de travail sur les activités effectives et empêchées des enseignants : Rôle de la configuration de la salle de sciences dans l'apprentissage de la compétence d'argumentation* [PhD Thesis]. Université Grenoble Alpes.
- Petreska von Ritter, B., Edelmann, R., Silva, D. D., Farenc, N., Haussauer, V., Holzer, R., Pellet, J.-P., & Hersch, M. (2023). Modulo, des moyens d'enseignement de l'informatique à visée participative. *Petit x*, 119, 7-19.
- Sigayret, K., Tricot, A., & Blanc, N. (2022). Unplugged or plugged-in programming learning : A comparative experimental study. *Computers & Education*, 184, 104505. <https://doi.org/10.1016/j.compedu.2022.104505>
- SVIA, SSIE, & SSII (Éds.). (2018). *Interface*. https://svia-ssie-ssii.ch/wp-content/uploads/2019/06/SVIA_Interface_2018.pdf
- Vincent, J.-M. (2017). *L'informatique débranchée*. Editions POLE.
- Wing, J. M. (2006). Computational thinking. *Communications of the ACM*, 49(3), 33-35.