

Dossier professionnel

Dans le cadre de la formation CAPAES

Pierre BETTENS

23 mai 2005

Résumé

Dossier écrit dans le cadre de la formation CAPAES définie par le décret du 17 juillet 2002. Ce document est une réflexion personnelle analysant tant mon parcours professionnel que ma pratique dans l'enseignement. Ce document se réfère à l'article 4 du décret, il constitue la troisième partie de la formation CAPAES.

Je décrirai, tout au long de ce document, comment je remplis mes fonctions. Je tâcherai de commenter et d'illustrer au mieux mes propos.

1 Qui suis-je ?

Je suis détenteur d'une licence en sciences mathématiques option algorithmique et programmation et d'un AESS¹ en sciences. Je travaille actuellement comme maître-assistant à la Haute Ecole de Bruxelles, département informatique (HEB / ESI).

La Haute Ecole de Bruxelles (HEB) est subdivisée en trois départements. Le département informatique sur le site de l'Ecole Supérieure d'Informatique (ESI) organise un bachelor et compte trois sections :

- informatique de gestion, il dépend alors de la catégorie économique,
- informatique industrielle et systèmes, finalité industrielle, il dépend de la catégorie technique, ou
- informatique industrielle et systèmes, finalité réseaux et télécommunication, qui dépend également de la catégorie technique.

Les deux autres départements sont ; le département pédagogique sur le site de DEFRE et le département "langues" sur le site de ISTI.

Pour rédiger ce document, je me suis basé sur une note de service de la Haute Ecole de Bruxelles, département informatique (HEB / ESI) expliquant brièvement le contenu du décret du 17 juillet 2002. Le candidat, moi-même en l'occurrence, doit démontrer une approche réflexive de sa pratique professionnelle en analysant son parcours, commentant, illustrant ses prestations. Je peux me baser sur les 14 compétences de l'article 3 du décret.

Dans cette première section, je me présente rapidement.

Dans la section 2, je présente mon parcours au sein de la Haute Ecole de Bruxelles, département informatique (HEB / ESI). J'explique le fonctionnement de l'école aux niveaux desquels j'interviens. Je présente rapidement les différents cours dont j'ai ou j'ai eu la charge. Je reste, à ce stade assez superficiel, cette section n'ayant pour but que de présenter mon environnement de travail.

Dans la section 3, j'illustre plusieurs compétences choisies parmi les 14 présentées à l'article 3 du décret. C'est dans cette section que je porte un jugement réflexif sur mes actions.

Enfin, je conclus dans la section 4.

2 Qu'ai je fait ? / Que fais-je ?

Cette section a pour objet de décrire mon parcours professionnel. Bien qu'ayant également travaillé dans une école secondaire pendant 5 ans, je ne développerai que mon travail au sein de la Haute Ecole de Bruxelles, département informatique (HEB / ESI).

J'ai travaillé à l'Institut Technique Don Bosco (IDB) de Woluwé-Saint-Pierre. Mon rôle dans cette école consistait à assumer les cours de Mathématiques pour les étudiants de 5^{ème}, 6^{ème} technique de qualification et professionnelle et 7^{ème} professionnelle dans les sections informatique, imprimerie et menuiserie.

Ma manière de travailler à changer du tout au tout (ou presque) lors de mon passage de l'enseignement secondaire supérieur à l'enseignement supérieur de type court. En effet,

¹Agrégation de l'Enseignement Secondaire Supérieure

à l'*Institut Technique Don Bosco (IDB)* je me concentrais assez peu sur la matière à enseigner pour forcer mon attention à la gestion du groupe. J'étais très attentif au contact avec les élèves afin de conserver une certaine discipline. Une grande partie de mon activité en classe était la gestion du groupe. L'essentiel, dans ces classes, est d'avoir un bon rapport avec les étudiants.

à la *Haute Ecole de Bruxelles, département informatique (HEB / ESI)*, je me concentre sur la matière que j'enseigne. J'essaie d'être au maximum compétent dans les divers cours que je donne. Je ne dois plus me focaliser sur la relation *eyes to eyes* avec les étudiants car il n'y a pas (ou très peu) de problèmes de discipline. Le contact avec les étudiants est facile, convivial et sympathique.

2.1 Au sujet des cours

Au cours des quelques années passées à la *Haute Ecole de Bruxelles, département informatique (HEB / ESI)*, j'ai assumé divers cours. Je vous présente, brièvement, l'organisation interne de l'école.

Les cours sont soit **théoriques**, soit **pratiques** soit sont des cours de **langage**². Les cours sont « chapeautés » par un **coordinateur**.

Les cours théoriques se donnent tantôt en groupe d'environ 120 étudiants - je parlerai alors de *grands groupes*, tantôt en groupe de 60 étudiants - je parlerai alors de *groupes intermédiaires* et tantôt en groupe d'environ 30 étudiants - je parlerai alors de *petits groupes*.

Les cours pratiques sont généralement des laboratoires. Ces laboratoires se donnent en groupes de 15 étudiants - je parlerai alors de *groupe de laboratoires*.

Les cours de langage se composent d'un cours théorique par groupe de 60 étudiants et d'un laboratoire par groupe de 15 étudiants. Afin d'offrir aux étudiants un meilleur suivi, le professeur³ en charge du cours théorique sera également en charge des laboratoires pour les quatre sous-groupes correspondant au cours théorique.

Le coordinateur. Étant donné le nombre d'étudiants, un même cours est donné par plusieurs professeurs. Le coordinateur s'assure de la cohérence entre les différents groupes.

- il est responsable du syllabus commun,
- il répartit les tâches entre les professeurs qui donnent le cours ou le laboratoire, il précise qui prépare les exercices, les examens, ...
- il s'assure du respect des échéances en « relancant » ses collègues,
- ...

²Je différencie ici les cours de langage car ils ont certaines particularités importantes.

³Chaque fois que j'emploie le terme *professeur*, il est bon de le traduire par le terme officiel maître-assistant

Pour chaque cours, je tâcherai de présenter, si nécessaire, un rapide descriptif du cours, ce qu'il implique en terme de préparation, la manière d'évaluer le cours, ce que représentent les corrections et/ou les particularités du cours en terme de pédagogie ou d'organisation. Je préciserai, également la ou les années où j'ai donné le cours. Lorsque je ne précise pas, les cours sont destinés aux étudiants de *bachelor* en informatique toutes sections confondues.

2.1.1 Cours de Mathématiques

Public: 1^{ère}

Année(s): 1998-1999, 1999-2000

Cours théorique en petits groupes.

C'est le seul cours qui repose sur des prérequis. Comme tel, il doit être géré différemment des autres cours, nous devons faire face à un public hétérogène (en terme de connaissances) et le faire avancer.

2.1.2 Cours de Logique

Public: 1^{ère}

Année(s): 1998-1999, 2000-2001

Cours théorique en petits groupes qui est en fait un cours d'algorithmique élémentaire où les étudiants peuvent se familiariser avec les structures conditionnelles, répétitives, les structures de données élémentaires, etc.

Pour illustrer des concepts algorithmiques, il existe deux écoles :

- la première utilise le **pseudo-code**, un formalisme suffisamment abstrait que pour atteindre un niveau de généralité acceptable et suffisamment concret pour être utilisable. L'avantage d'une telle approche est l'indépendance par rapport au langage.
- la seconde école prône l'**utilisation d'un langage** afin d'illustrer les concepts. Avec cette approche, l'étudiant est en mesure de tester ce qu'il produit. Dans ce cas, c'est le compilateur et non lui qui sanctionne ses algorithmes. L'étudiant est moins à même de *savoir* si un algorithme est correct ou pas, il attend l'avis du compilateur. Cette approche est aussi proche d'un langage. Ces deux inconvénients, mis en avant par les partisans de la première école, sont à mettre en parallèle avec l'opportunité qu'offre cette méthode de pouvoir tester ses algorithmes à un stade où l'étudiant est encore clairement néophyte en la matière.

C'est l'approche *pseudo-code* qui est choisie à la *Haute Ecole de Bruxelles, département informatique (HEB / ESI)*. Ce choix a déjà fait l'objet de beaucoup de discussions. Les arguments de chacun étant tous valables, je reste indécis quant au choix d'une des deux écoles. Je crois que c'est à chacun de sentir à quel moment il a besoin d'un compilateur pour valider son algorithme.

2.1.3 Cours d'introduction aux logiciels de base

Public: 1^{ère}

Année(s): 1998-1999

Ce laboratoire n'est actuellement plus dans notre grille horaire. Il consistait en l'apprentissage d'une suite bureautique, MS Word, MS Excel et MS Access.

2.1.4 Cours de Microprocesseurs et logiciels de contrôle et laboratoires

Public: 1^{ère}

Année(s): 1998-1999, 1999-2000, 2000-2001, 2001-2002, 2002-2003, 2003-2004

Cours théorique en grands groupes, accompagné de laboratoires.

Pendant ma première année de coordination de ce cours, j'ai remanié le syllabus [?] afin qu'il réponde mieux à mes attentes en tant qu'outil d'apprentissage pour l'étudiant. J'ai repris le travail de mes prédécesseurs en y amenant des améliorations. J'ai choisi de transformer le syllabus en *slides commentés*. Cette approche permet, à mon sens, d'avoir un lien fort avec l'exposé oral, permet facilement de compléter ses notes, et ne demande pas d'écrire trop pendant le cours étant donné que certains commentaires apparaissent déjà dans le syllabus.

2.1.5 Cours d'Assembleur et laboratoires

Public: 1^{ère}

Année(s): 1998-1999, 1999-2000, 2000-2001, 2001-2002, 2002-2003, 2003-2004

Cours théorique en grands groupes et laboratoires, s'articulent de la même manière que le cours de microprocesseurs. J'ai également remanié le syllabus [?], de manière plus profonde cette fois. J'ai conservé l'approche *slides commentés* du cours de microprocesseurs.

2.1.6 Initiation à la programmation

Public: 1^{ère} assistant ingénieur

Année(s): 1999-2000

Cours qui se donnait aux étudiants «assistant-ingénieur». La section n'existe plus, le cours non plus! Dans ce cours d'introduction, je voulais couvrir les différents niveaux de programmation. J'entend par là la programmation de bas niveau, la programmation dans des langages évolués et la programmation au niveau utilisateur.

Afin d'atteindre mes objectifs, je leur ai montré, très rapidement, un exemple de programme assembleur. Ensuite, j'ai donné un cours d'algorithmique élémentaire avec les exemples illustrés en PASCAL [13]. Ce cours étant leur seul contact avec l'informatique je trouvais bon qu'ils testent leurs algorithmes sur machine. La troisième partie du cours, que je n'ai pas eu le temps d'aborder, consistait en un langage de haut niveau, je voulais utiliser un tableur.

Les notes [?] sont une bonne introduction aux concepts de base de la programmation. On retrouve ici la « seconde école » (cfr 2.1.2)

2.1.7 Cours de langage C et laboratoires

Public: 1^{ère}

Année(s): 1999-2000, 2000-2001, 2001-2002

Cours théorique en petits groupes, accompagné de laboratoires.

Personnellement, j'ai beaucoup aimé [19] qui traite de la manière de commenter ses programmes. J'en ai fait l'éloge auprès des étudiants et y ai puisé des exemples afin de bien faire comprendre l'importance de **commenter** correctement son code.

2.1.8 Cours de langage Java et laboratoires

Public: 1^{ère}

Année(s): 2003-2004

Ce cours remplace le cours de langage C, il s'organise de la même manière si ce n'est que l'on observe des différences au niveau des laboratoires. Nous nous sommes chargés de la réalisation des slides en commun en utilisant un wiki [24].

Nous avons choisi de proposer des *travaux dirigés* tout au long du premier semestre. Ces *travaux dirigés* ne sont pas cotés, ils sont libres. Au second semestre, les étudiants doivent réaliser deux projets cotés. Cette approche est très formative au premier semestre, la certification n'apparaissant qu'au second. Cela permet aux étudiants de "prendre calmement le langage en main".

2.1.9 Cours de système d'exploitation

Public: 1^{ère}

Année(s): 2000-2001, 2001-2002

Cours théorique, se donne en petits groupes. Nous regrettons qu'il ne soit pas possible d'y ajouter des laboratoires. J'ai réécrit le syllabus afin qu'il soit plus accessible aux étudiants.

2.1.10 Cours de système d'exploitation

Public: 2^{ème} industrielle

Année(s): 2000-2001, 2001-2002, 2002-2003

Un des cours pour lesquels un bon rafraichissement des connaissances s'imposait. C'est un cours passionnant. Je n'ai pas la chance de donner la partie théorique, assurée par Jean-Claude mais je m'occupe de quelques laboratoires qui illustrent les concepts développés théoriquement. Il existe pas mal de lecture pour ce cours, [12], [23], [17], [20] ...

2.1.11 Cours de système d'exploitation

Public: 3^{ème} industrielle

Année(s): 2001-2002, 2002-2003, 2003-2004

Dans ce cours, qui se déroule dans un laboratoire, nous approchons l'aspect *temps réel* des systèmes d'exploitation. Je consacre peu de temps à la théorie et me penche sur la compréhension et l'intégration de l'API RtLinux [25].

2.1.12 Cours de gestion et administration des réseaux, intégration Windows NT et Linux

Public: 2^{ème} industrielle

Année(s): 2001-2002, 2002-2003, 2003-2004

Cours qui se donne en parallèle avec les cours de *Gestion et administration des réseaux, aspect Windows* et *Gestion et administration des réseaux, aspect Unix*, il traite principalement de *Samba* [26] pour l'aspect pratique, mais d'autres solutions sont exposées sans être développées.

J'ai rédigé des notes [?] basées essentiellement sur [21], livre que je recommande aux étudiants et que je laisse à disposition au laboratoire.

2.1.13 Cours de méthodologie de l'informatique

Public: 1^{ère}

Année(s): 2000-2001, 2001-2002

Ce cours, sous-fréquenté, essaie de donner aux étudiants des *tips and triks* de réussite. Bien qu'ayant un coordinateur, chaque titulaire du cours l'organise comme il l'entend, seuls les contenus sont coordonnés. Ce cours et la manière d'encadrer les étudiants de première afin de favoriser leur réussite est actuellement en discussion et refonte complète (cfr 2.2)

C'est avec ce cours que j'ai commencé à mettre de l'information à disposition des étudiants via mon site web [27].

2.2 Tutorat et remédiations

Pendant quelques années, nous avons organisé, sur le temps de midi, des séances de **tutorat**. Des professeurs se tiennent à disposition des étudiants afin de répondre à leurs questions concernant les cours de Math, Logique et Langues principalement.

L'idée est bonne mais ne fonctionne pas bien. Les quelques rares étudiants qui se présentent ne sont pas ceux qui en ont le plus besoin. Les ressources engagées sont sous-utilisées. À ce jour, nous abandonnons cette manière de faire.

Un système de *coaching* des groupes d'étudiants est à l'étude pour les années futures.

L'idée est la suivante :

Un professeur est chargé d'encadrer un groupe d'une quinzaine d'étudiants tout au long de l'année. Ce professeur ne donne pas cours à ces étudiants.

Le *coach* propose des moments d'évaluation aux étudiants. Certains d'entre eux sont obligatoires. Pendant ces moments, il peut fournir des explications supplémentaires, rediriger vers un autre collègue, proposer une méthode de travail, "remettre en selle" un étudiant, ...

Je pense que cet encadrement peut être très bénéfique pour les étudiants, nous verrons ...

2.3 Suivi des étudiants en stage

Chaque enseignant de la *Haute Ecole de Bruxelles, département informatique (HEB / ESI)* est chargé de suivre, au minimum, un étudiant en stage. Ce suivi nous permet d'avoir un contact privilégié avec un étudiant de troisième. Il est aussi très bénéfique pour moi qui ne suis jamais sorti de l'environnement scolaire. Il me permet d'avoir un contact avec l'entreprise. Nous en profitons toujours pour se renseigner sur les profils recherchés dans l'entreprise. Ces moments de discussion nous permettent de nous remettre en question, ils nous donnent des pistes quant à la pertinence de nos choix.

2.4 Evaluation et délibérations

Les délibérations sont une période assez difficile pour moi. Elles marquent le moment de l'année où nous devons évaluer nos étudiants. Afin qu'elles se passent pour le mieux, nous nous sommes défini des règles internes de réussite et d'échec. Hors de ces règles, nous délibérons effectivement. Cette période est toujours une période de choix de conscience : "*Qu'est-ce qui est le mieux pour untel, en tant qu'étudiant et individu ?*". Cette période se passe mieux si l'évaluation a été bien faite.

L'évaluation concerne l'aspect certificatif de mon travail et c'est celui qui me motive le moins. Autant j'aime me former, enseigner, partager mon expérience, etc, autant je n'aime pas évaluer les étudiants. Je dois à la fois les coter ⁴ sur le travail qu'ils ont fait et estimer leur capacité à suivre les cours suivants, à s'intégrer dans le monde professionnel, etc.

Je dois pouvoir dire si les résultats obtenus dans un cours justifient que l'étudiant recommence son année ou passe ou encore lui donne une dispense du cours en cas de redoublement. Ces choix doivent toujours être pris en âme et conscience.

En début de carrière, c'était simple ! J'évaluais des mathématiques. Dans ce cas de figure, l'évaluation ne peut pas être ambiguë ; $1 + 1 = 2$ ⁵. La réponse est correcte ou ne l'est pas.

Par contre, dans le domaine informatique, beaucoup de paramètres entrent en ligne de compte. A la lecture d'un code source, je dois évaluer s'il se compile, s'il tourne mais encore évaluer sa robustesse, sa réutilisabilité, sa lisibilité, s'il est correctement et suffisamment commenté, sa justesse, si les étudiants utilisent bien les concepts vus au cours, ...

2.5 Conseil de département

Je suis membre du conseil de département technique de la *Haute Ecole de Bruxelles, département informatique (HEB / ESI)* depuis l'année scolaire 2001-2002. Je me suis porté candidat car je m'intéresse au fonctionnement de l'école. J'estime que ces heures consacrées bénévolement sont bénéfiques personnellement et professionnellement et je pense qu'elles le sont aussi pour l'école.

⁴Noter en français

⁵Quoique en base 2 ...

3 Quelles compétences ai-je ?

Sous ce titre bien pompeux, je vais essayer de faire un état des lieux des compétences que je développe dans mon travail. Ce genre d'exercice est toujours périlleux, il impose de s'évaluer correctement sans se sous-estimer ni se sur-estimer.

Par facilité, j'ai repris quelques compétences définies dans le décret CAPAES (cfr article 3, décret du 17 juillet 2002) que je vais tâcher d'illustrer.

3.1 Promouvoir la réussite des étudiants notamment par le prise en compte de la diversité des parcours

Nos étudiants viennent soit de l'enseignement général toutes disciplines confondues soit de l'enseignement technique. Hormis pour le cours de Math qui se base sur certains acquis, les cours ne supposent aucun prérequis. Tenir un tel discours est sans doute hypocrite, nous supposons toujours qu'un étudiant a réussi l'enseignement secondaire et nous basons sur des connaissances communes.

Pour faciliter la *remise à niveau*, je propose (via mon site web [31]) des liens et des références de livres. Je signale également aux étudiants que je suis disponible pour répondre à leurs questions, soit oralement, soit par mail ou encore via un forum que j'ai mis en place.

3.2 Faire face aux devoirs et aux dilemmes éthiques de la profession

Je ne peux voir le terme *éthique* sans parler du choix des logiciels que nous proposons à nos étudiants et par là, des **logiciels libres**.

Nos étudiants doivent, et personne ne peut le nier, travailler chez eux afin d'acquérir les diverses compétences que nous exigeons. Ils ont, pour ce faire, besoin de divers logiciels allant des compilateurs à certains IDE ⁶ spécifiques en passant par des logiciels de bureautiques tels tableur, traitement de texte et générateur de slides.

J'estime qu'il est de mon devoir d'éclairer les étudiants sur :

- les dangers du monopole d'une société,
- les problèmes qu'occasionnent des formats fermés,
- les implications du piratage,

J'essaie de les ouvrir aux logiciels libres. Je pointe leurs mauvaises habitudes et explique pourquoi il est bon de lutter contre ces pratiques.

3.3 Travailler en équipes pluridisciplinaires en partageant la responsabilité collective de la formation

Le fonctionnement interne de la *Haute Ecole de Bruxelles, département informatique (HEB / ESI)* est propice au développement de cette compétence si ce n'est que le travail n'est pas spécialement *pluridisciplinaire*.

Pour chaque cours, un coordinateur est désigné. Il est responsable de l'homogénéité du cours. Nous sommes généralement plusieurs maîtres-assistants à assurer la charge d'un même cours pour des groupes différents de la même année d'étude.

⁶*Integrated Development Environment*

Dans ce cas, il est primordial, tant pour l'étudiant que pour le corps professoral, que les contenus et les niveaux soient les plus proches possibles.

Pour ce faire, les notes de cours sont communes à l'ensemble des titulaires du cours. L'équipe en charge d'un cours conçoit les questions d'examen et les énoncés d'exercices en concertation.

Exemple du cours de langage Java

Cette année est la première année où nous donnons *Java* comme premier langage de programmation. Nous avons donc du, mes collègues et moi-même concevoir le cours, décider de l'organisation des laboratoires, etc. Nous nous sommes d'abord rencontrés afin de fixer la manière de travailler. Nous avons décidé d'utiliser un *wiki* [24] pour communiquer sans devoir nous réunir trop fréquemment. Il n'empêche que nous nous rencontrons ponctuellement, dans les couloirs ou lors de réunions organisées à la demande.

Dans cette préparation, deux points se détachent :

1. Confection du cours

En nous basant sur la documentation Java et les livres associés [28], nous avons, via le wiki, confectionné le cours. Un collègue est chargé du rassemblement des informations, un autre de la confection des slides.

2. Organisation des laboratoires

Suite à quelques réunions fixant la manière de travailler, nous nous sommes réparti les diverses préparations ; les *travaux dirigés*, les projets du second semestre et les questions d'examens.

L'organisation des autres cours dans lesquels j'interviens est sensiblement la même. Lorsque je suis, moi-même, coordinateur d'un cours, je fonctionne toujours de cette manière.

3.4 Construire avec les étudiants un contexte propice à l'apprentissage

Afin de construire un tel contexte, je pense qu'il faut que les *règles du jeu* soient bien définies dès le départ. Pour ce faire, je pose clairement mon contrat didactique au début du cours. Je fixe

- le nombre de séances et leur caractère obligatoire ou pas (cfr 3.5)
- mes attentes quant à leur travail : "*Que devez-vous faire ?*",
- la manière d'évaluer le laboratoire ou le cours ; *quoi et comment ?*.

Dans le cas d'un laboratoire, je définis, dans mon exposé oral et dans les notes que je remets, le travail attendu. Ceci peut prendre diverses formes : rapport de projet, présentation orale, rapport "format texte", manipulation personnelle, etc (cfr 3.5). Dans le cas d'un cours théorique, je précise simplement la méthode d'évaluation (cfr 3.6).

D'un point de vue plus humain, j'adapte mon attitude en fonction du public et du type de cours. Je vouvoie généralement mes étudiants afin de garder une certaine distance. Je les tutoie lorsque je veux accentuer le contact de manière plus personnelle, lorsque je donne une explication individuelle par exemple.

Dans les cours théoriques, je suis relativement distant pour éviter les problèmes de discipline - sans doute un reliquat de mon passage

dans le secondaire - et je compense par une bonne dose d'humour. Je pense que cette approche est bien perçue.

Dans les laboratoires de première, je garde cette distance en début d'année lorsque les étudiants sont nombreux et n'ont pas encore l'habitude du travail au laboratoire. Par la suite, je cherche plus le contact individuel et m'autorise le tutoiement ponctuel.

Dans les laboratoires de seconde et troisième, je vouvoie les étudiants dès que je parle face au groupe mais les tutoie systématiquement dans les contacts individuels. J'essaie de créer une bonne ambiance dans le laboratoire où chacun trouve sa place et n'hésite pas à poser ses questions.

3.5 Planifier le cours et construire des dispositifs d'enseignement appropriés aux adultes

Je distinguerai dans cette section les cours théoriques des laboratoires.

3.5.1 Cours théoriques

Les cours théoriques consistent en des exposés magistraux. Cette manière de faire nous est imposée par le nombre. Elle n'est pas forcément négative. Elle permet à l'étudiant d'avoir un autre type de relation avec la matière ...et ne l'empêche pas de participer. Au fil des années, l'expérience aidant, je m'autorise une pédagogie interrogative. Même composé de 40 à 100 étudiants, j'envoie parfois un étudiant au tableau afin qu'il résolve un exercice ou réexplique une notion à ses condisciples.

Lors du premier cours, je précise :

- quelles sont les notes et comment se les procurer,
- où se trouve l'information (valves, site internet [31], ...),
- la bibliographie que je commente,
- le nombre de séances, l'importance du cours en terme de pondération finale,
- l'organisation du cours, existence ou non d'un laboratoire associé,
- la méthode d'évaluation

Cette manière de faire responsabilise les étudiants. Je leur fournis une série d'outils et leur précise comment les utiliser . Je ne vérifie pas s'ils les utilisent, convenablement ou non.

3.5.2 Laboratoires

Les laboratoires s'organisent de manière très diverses suivant les équipes qui en ont la charge. On distingue ; les **travaux dirigés**, les **projets** et les **manipulations au laboratoire**.

Les travaux dirigés Ils se déroulent donc au laboratoire. Les étudiants reçoivent, par séance, une série d'exercices illustrant tel ou tel concept. Le professeur en charge du labo se tient à la disposition des étudiants afin de répondre aux questions, orienter, commenter les solutions, ...

Je ne vais pas vers l'étudiant, j'attends qu'il vienne vers moi. Je précise, et rappelle en début de chaque séance, que je suis disponible pour répondre aux questions. J'attends que les étudiants fassent un premier pas vers moi. Je suis convaincu que leur laisser cette initiative les aide à grandir. Cette approche fonctionne assez bien.

Les exercices proposés ne sont pas obligatoires et les étudiants avancent à leur propre rythme. Nous leur signalons s'ils sont en concordance avec nos attentes. Ces séances ne sont pas cotées.

Les projets Dans ce cadre, les étudiants reçoivent un énoncé qui les tiendra en haleine plusieurs semaines. La présence des étudiants est libre. Ils travaillent soit chez eux, soit au laboratoire où je me tiens à leur disposition. Nous ne nous inquiétons - au sens vérification - pas de la présence d'un étudiant, il choisit la méthode de travail qui lui convient le mieux.

Nous proposons un échéancier aux étudiants présents et nous les suivons s'ils le désirent. Le seul point sur lequel nous sommes stricts est la date de remise du rapport, nous fixons une *date butoir* après laquelle il est trop tard pour remettre son rapport. Nous nous arrangeons pour que l'étudiant moyen puisse remettre son travail lors des laboratoires qui précèdent ladite date butoir.

Avec les étudiants présents, je peux alors instaurer un climat convivial dans lequel nous apprenons à nous connaître et je tente de les guider vers leur solution du problème.

Manipulation au laboratoire Je propose ici un sorte de *how to* sur un sujet. L'étudiant est amené à tester les divers concepts un par un. C'est un type d'apprentissage plus dirigé.

Je laisse les dernières manipulations très "ouvertes" afin de voir jusqu'où va l'étudiant, comment il cherche, quel est son degré de curiosité. Ce *degré de curiosité* est très important pour moi car j'estime qu'un informaticien doit être curieux, il doit tester les choses, essayer différents cas, ...

3.6 Maitriser et utiliser les outils d'évaluation des apprentissages adaptés à l'enseignement dispensé et pouvoir répondre de ses choix

3.6.1 Évaluation des cours théoriques

Je donne rarement des cours à proprement parler théorique en ce sens qu'ils demandent de pouvoir restituer un savoir et de faire preuve d'une compréhension des concepts. Nous enseignons à de futurs *bachelor* en informatique, qui devront faire preuve de connaissances pratiques plus que théoriques. Dans cette optique, nous évaluons peu la connaissance des concepts théoriques en tant que tels mais plutôt la capacité à les mettre en oeuvre.

Dans le cas d'un cours de langage, par exemple. L'examen du cours théorique se déroule "sur machine". Les étudiants sont amenés à générer des bouts de code et à justifier leurs choix d'implémentation, nous testons la capacité de l'étudiant à écrire du code, savoir le *debugger*, s'y retrouver dans la documentation, ... Pour ce faire, nos questions sont graduelles et plus théoriques au fur et à mesure de l'avancement de l'examen. Nous commençons par une question du type "*Hello world*"

pour terminer par des questions portant sur une classe inconnue du langage ou sur la gestion des exceptions, ...

3.6.2 Évaluation des laboratoires

Je vais détailler ici les différentes formes d'évaluation que je mets en place dans les laboratoires. Pour rappel (voir 3.4), je distingue : rapport de projet, présentation orale et rapport "format texte".

Rapport de projet Les étudiants doivent me remettre un rapport complet. Celui-ci précise la démarche qu'ils ont mis en oeuvre afin de résoudre le projet, présente toutes les variables utilisées dans le programme et contient une copie du code source commenté.

Dans la pratique, les étudiants ont beaucoup de difficultés pour présenter leur manière de travailler. Je constate souvent une mauvaise maîtrise du français tant en grammaire qu'en orthographe. Ils donnent également l'impression de ne pas avoir analysé leur problème.

Présentation orale Cette présentation a lieu lorsque les étudiants doivent faire des recherches sur un sujet. Elle permet à l'étudiant de s'exprimer devant le groupe. Elle l'habitue à faire des démonstrations devant un «public».

Je trouve important de développer cette capacité ainsi que préparer une activité pour un groupe. Habituellement, les étudiants qui présentent nous⁷ demandent de réaliser certaines manipulations pour tester les logiciels et/ou concepts présentés.

Cette approche est souvent enrichissante et appréciée par les étudiants.

Rapport "format texte" Je demande ce type de rapport lorsque l'étudiant réalise des manipulations au laboratoire. Ce texte me permet de savoir, étape par étape, ce qu'ils ont fait comme manipulations.

⁷Les autres étudiants et moi-même

3.7 Porter un regard réflexif et interdisciplinaire sur ses connaissances scientifiques et son enseignement

Plus j'en sais et plus je me rends compte de l'étendue de ce que je ne connais pas

Un inconnu

Cette phrase résume un peu comment je perçois mes connaissances scientifiques. Chaque année (pour ne pas dire chaque jour) passée à la *Haute Ecole de Bruxelles, département informatique (HEB / ESI)*, j'apprends. Que ce soit en langage de programmation, en système d'exploitation, dans les domaines plutôt du ressort de mes collègues ou encore dans ma manière de donner cours ou dans la manière dont je suis perçu des étudiants, chaque jour apporte son lot de découvertes.

Je me rends compte au fil des années que je dois plutôt me spécialiser qu'essayer d'être généraliste. Pour moi *l'informatique* est un domaine aussi vaste que les *sciences* ... et on ne peut pas être à la fois mathématicien et biologiste.

Mes choix de "spécialisation" se portent sur **les langages de programmation, les systèmes d'exploitation**, l'aspect *temps réel* et en moindre mesure la **sécurité**.

3.7.1 Connaissances scientifiques

La connaissance souffre d'une certaine dualité.

D'une part, elle s'accroît chaque jour. En ce qui me concerne, je parle de lectures, de discussions avec les collègues et de contacts avec les étudiants. J'apprends car je suis curieux et lis beaucoup de choses. J'apprend car je me renseigne sur les cours de mes collègues ainsi que sur leurs centres d'intérêt et nous en discutons. J'apprend car dans les laboratoires les étudiants travaillent, cherchent et me font part de leurs résultats.

D'autre part, la connaissance est volatile. Ce qui n'est pas entretenu, utilisé, a une fâcheuse tendance à s'oublier. Pour pallier à cela, j'essaie d'éviter de répéter des cours. J'entends par là que je préfère avoir un cours de Java et un cours de système plutôt que deux fois le cours de Java. Cela m'oblige à faire un travail de rappel sur un ensemble de connaissances afin qu'elles ne sombrent pas dans l'oubli.

3.7.2 Enseignement

Lorsque j'ai débuté - stress de début de carrière aidant - j'étais plutôt adepte d'une pédagogie transmissive. Je me suis rapidement rendu compte que ce n'était pas la solution en particulier avec un public de l'enseignement secondaire technique. Je suis assez vite passé à une pédagogie plus active que je mets en pratique aujourd'hui encore. Je tâche, par exemple, de ne pas présenter une série de slides de manière soporifique. J'interroge l'auditoire ou un étudiant en particulier, je plaisante, ...

En parallèle à la pédagogie, je pense que *l'enseignement est une histoire de contacts*. Lors de la première rencontre, j'essaie d'avoir un contact *eyes to eyes* (rapide) avec chaque étudiant. Je montre que je suis présent.

3.8 Actualiser ses connaissances et ses pratiques

La formation, surtout dans ce domaine, est primordiale. Je passe beaucoup de temps à me former. La difficulté est que le domaine est tellement vaste que j'ai l'impression que la tâche est impossible. Pour arriver à mes fins, j'essaie :

- de lire beaucoup,
- de participer à des formations,
- de me tenir informé via le net par exemple

3.8.1 La lecture

Si le lecteur se penche sur la bibliographie technique, il pourra prendre connaissance de quelques-uns de mes centres d'intérêt. Je lis chaque mois les magazines *Login* et *Linux magazine* ... parfois *Misc*.

Dans l'optique d'intégrer la formation à Perl dans un cours d'administration système, je me suis procuré [16], [10] et [15]. Actuellement, je m'intéresse au langage Ada [14] afin d'éventuellement l'intégrer au cours de système d'exploitation de troisième, développant les concepts "temps réel" en complément à RTLinux.

3.8.2 Les formations

Chaque année, j'essaie de participer à l'une ou l'autre formation. Actuellement, c'est le centre *TechnoFutur3* [29] qui retient notre attention. Voici la liste, j'espère exhaustive, des formations que j'ai été suivre depuis que je travaille à la *Haute Ecole de Bruxelles, département informatique (HEB / ESI)*. Vous ne trouverez pas de justificatif en annexe pour la simple raison que je n'ai pas juger opportun de les conserver.

1999-2000	Siemens Siemens	Linux fondamentals Linux administration
2000-2001	TechnoFutur3 Cours du soir Formation interne	Intégration Windows NT / Linux Formation Delphi UML
2002-2003	TechnoFutur3 TechnoFutur3 Formation interne	Introduction Java Java Avancé Base de données
2003-2004	TechnoFutur3 TechnoFutur3	Java Enterprise Java Beans Design Pattern

3.8.3 Information via le net

Ma principale source d'information est le site *Linuxfr.org* [30], portail francophone sur linux et les logiciels libres. Je ne vais pas présenter dans le détail les avantages d'un tel site. En bref, il fournit beaucoup d'informations récentes sur des sujets très divers.

3.9 Etre un partenaire actif dans l'organisation et le développement de son institution

Il est primordial que chaque enseignant s'implique dans son école. Dès mon entrée en service au sein de la haute école, je me suis engagé dans le **conseil de**

département (voir 2.5). Hormi cet engagement de cinq ans, chaque année est ponctuée de plusieurs rendez-vous.

3.9.1 Le "triumvirat"

Il y a quelques années, nous avons changé de locaux. Je passe sur le détail du déménagement pour lequel beaucoup de collègues ont redressés leurs manches et j'enchaîne sur la réorganisation des laboratoires et de l'intranet de l'école. La mise en place d'une nouvelle infrastructure demandant un travail assez conséquent et difficilement supportable par l'unique technicien de l'époque, une équipe appelée *triumvirat*⁸ a servi d'interface entre la technique et les professeurs.

Ce triumvirat est encore actif. Il propose des idées ou met en oeuvre ses idées ou celles proposées par les collègues afin que l'outil informatique soit pratique et pleinement utilisable.

3.9.2 Rénovation des grilles

Nous avons entrepris la **rénovation des grilles de cours** pour toutes les actions. Dans ce cadre, je participe activement aux réunions. Ces réunions m'apportent beaucoup et m'offre une vue plus vaste de notre école. Elles me permettent aussi de dialoguer avec mes collègues et d'avoir de riches échanges d'idées.

3.9.3 Responsabilité d'un laboratoire

Les laboratoires de l'école sont répartis en deux catégories, les *laboratoires stables* et *instables*.

Les laboratoires stables sont gérés par les techniciens. Les machines de ces laboratoires ont une configuration fixe proposant une liste de logiciels fixées à l'avance.

Les laboratoires instables sont destinés aux manipulations demandant les droits d'administrateur/root. Ces machines sont donc destinées à des tests et nécessitent d'être parfois réinstallées. Pour que tout se passe bien, les divers collègues utilisant un tel labo doivent se coordonner. Un prof est responsable de chaque laboratoire. Je suis responsable d'un *laboratoire instable*.

3.9.4 Site internet

Je propose un **site internet** aux étudiants. Ils y trouvent :

- les notes de cours,
- les énoncés d'exercices,
- certaines solutions,
- des outils (comilateurs, ...),
- une faq,
- un forum,
- un wiki,
- un formulaire d'évaluation (pour mes cours),
- ...

⁸Nous étions trois ...

4 Conclusion

J'ai commencé la rédaction de ce *dossier professionnel* avec quelques craintes. En effet, je ne me rendais pas bien compte de la tâche à accomplir et ne cernais pas bien les attentes du lecteur. Je pense maintenant que le principal objectif de ce travail est la remise en question de chacun. Ce n'est pas sans raison que le décret du 17 juillet 2002 nous demande d'avoir une approche réflexive sur notre enseignement.

Ce travail est maintenant terminé. Je peux dire qu'il a été tout aussi stressant que bénéfique ! Stressant simplement car il est certificatif. Bénéfique car il oblige à se remettre en question. Bien que me posant, chaque année, des questions quant à mon enseignement, mes méthodes, mon contact avec les étudiants, ... le fait de coucher ses réflexions par écrit est positif. Ce travail m'a obligé à structurer mes idées et à y réfléchir plus profondément. Cela me renforce dans l'idée qu'il faut s'évaluer régulièrement. C'est dans cette optique que je propose un formulaire d'évaluation aux étudiants sur mon site. Aujourd'hui plus qu'hier, j'essaie d'avoir un retour des étudiants dont j'ai la charge.

Références bibliographiques pédagogiques

- [1] L. PAQUAY et coll. *Former des enseignants professionnels. Quelles stratégies ? Quelles compétences ?* De Boeck, BRuxelles, 2001.
- [2] P. PARMENTIER L. PAQUAY et coll. En quoi les situations d'enseignement / apprentissage favorisent-elles la construction de compétences ? développement d'un outil d'analyse : le comp.a.s. 2002.
- [3] L. PAQUAY et coll J. BECKERS. Comment décrire et analyser un dispositif réputé professionnalisant ? proposition d'un outil descriptif et interprétatif.
- [4] L. PAQUAY et P. PERRENOUD M. ALTET. *Formateurs d'enseignants, quelle professionnalisation ?* De Boeck, Bruxelles, 2002.
- [5] P. MEIRIEU. *Apprendre ... oui, mais comment ?* ESF, Paris, 1987.

Références bibliographiques techniques

- [6] David N. BLANK-EDELMAN. *Perl for System Administration*. O'Reilly, Morris street, 101, Sebastopol, CA, 95472, July 2000. ISBN : 1 56592 609 9.
- [7] Francis BORCEUX. *TEX, la perfection dans le traitement de texte*. ARTEL, Bruxelles, ciaco edition, 1989. ISBN : 2 87085 194 4.
- [8] Marco CESATI Daniel P. BOVET. *Le noyau linux*. O'Reilly, Rue Séguier, 16, 75006 Paris, 2001. ISBN : 2 84177 141 5.
- [9] Peter GROGONO. *La programmation en PASCAL*. interEditions, Avenue du Maine, 87, 75014 Paris, 1986. ISBN : 2 7296 0128 7.
- [10] Inconnu. *Ada*. Inconnu, Inconnu. ISBN : .
- [11] Tom Christiansen Larry WALL and Jon ORWANT. *Programmation en Perl*. O'Reilly, Rue Séguier, 16, 75006 Paris, 3^{ème} edition, 2002. ISBN : 2 84177 140 7.
- [12] Randal L.SCHWARTZ and Tom PHOENIX. *Introduction à Perl*. O'Reilly, Rue Séguier, 16, 75006 Paris, 3^{ème} edition, 2002. ISBN : 2 84177 201 2.
- [13] Lar KAUFMAN Matt WELSH, Kalle DALHEIMER. *Le système Linux*. O'Reilly, Rue Séguier, 16, 75006 Paris, 3^{ème} edition, 2000. ISBN : 2 84177 086 9.
- [14] Steve OUALLINE. *Vi IMproved - Vim*. New Riders Publishing, 2001. ISBN : 0 7357 1001 5.
- [15] Steve OUALLINE. *Practical C programming*. O'Reilly, Morris street, 101, Sebastopol, CA, 95472, 3^{ème} edition, Août 1997. ISBN : 1 56592 306 5.
- [16] Yves LIGIER Paolo ZANELLA. *Architectures et technologies des ordinateurs. Cours et exercices résolus*. Dunod, 3^{ème} edition, 1999. ISBN : 2 10 003801 X.
- [17] Peter KELLY Robert ECKSTEIN, David COLLIER-BROWN. *Samba. installation et mise en oeuvre*. O'REILLY, 2000. ISBN : 2-84177-090-7.
- [18] Dominique RODRIGUEZ. *L'essentiel de TEXet GNU-Emacs, Manuel de réalisation de documents scientifiques*. Dunod, dunod edition, 2000. ISBN : 2 10 004814 7.
- [19] Andrew TANENBAUM. *Systèmes d'exploitation. Systèmes centralisés. Suystèmes distribués*. Dunod / Prentice Hall, 3^{ème} edition, 1994. ISBN : 2 10 004554 7.

Références internet

- [20] Pierre BETTENS. Wiki du site pit-it.com, 2003. <http://wiki.pit-it.com>.
- [21] Inconnu. Rtlinux. <http://rtlinux.org>.
- [22] Samba team. Samba.org. <http://samba.org>.
- [23] Pierre BETTENS. Site perso à destination des étudiants de l'esi, 2000. <http://esi.pit-it.com>.
- [24] Sun ©. Site officiel java. <http://java.sun.com>.
- [25] TechnoFutur. Technofutur3, centre de formation. <http://technofutur3.be>.
- [26] Fabien PENSO. Linux fr, da linux french page. <http://linuxfr.org>.
- [27] Pierre BETTENS. Site perso, 2000. <http://pit-it.com>.
- [28] Inconnu. Site officiel de la haute ecole de bruxelles, 2002. <http://www.heb.be> Redirige vers les sites des départements, ESI, ISTI, DEFRE.
- [29] Laurent BEECKMANS et Michel WILLEMSE. Site officiel du département *ecole supérieure d'informatique* de la haute ecole de bruxelles, 2002. <http://www.heb.be/esi>.
- [30] Joerg MUELLER. Freemind, 2000-2001. <http://freemind.sourceforge.net> Taking the concept-mapping approach of human-computer interface design.

Table des matières

1	Qui suis-je ?	2
2	Qu'ai je fait ? / Que fais-je ?	2
2.1	Au sujet des cours	3
2.1.1	Cours de Mathématiques	4
2.1.2	Cours de Logique	4
2.1.3	Cours d'introduction aux logiciels de base	4
2.1.4	Cours de Microprocesseurs et logiciels de contrôle et laboratoires	5
2.1.5	Cours d'Assembleur et laboratoires	5
2.1.6	Initiation à la programmation	5
2.1.7	Cours de langage C et laboratoires	6
2.1.8	Cours de langage Java et laboratoires	6
2.1.9	Cours de système d'exploitation	6
2.1.10	Cours de système d'exploitation	6
2.1.11	Cours de système d'exploitation	6
2.1.12	Cours de gestion et administration des réseaux, intégration Windows NT et Linux	7
2.1.13	Cours de méthodologie de l'informatique	7
2.2	Tutorat et remédiations	7
2.3	Suivi des étudiants en stage	8
2.4	Evaluation et délibérations	8
2.5	Conseil de département	8
3	Quelles compétences ai-je ?	9
3.1	Promouvoir la réussite des étudiants notamment par le prise en compte de la diversité des parcours	9
3.2	Faire face aux devoirs et aux dilemmes éthiques de la profession . . .	9
3.3	Travailler en équipes pluridisciplinaires en partageant la responsabilité collective de la formation	9
3.4	Construire avec les étudiants un contexte propice à l'apprentissage .	10
3.5	Planifier le cours et construire des dispositifs d'enseignement appropriés aux adultes	11
3.5.1	Cours théoriques	11
3.5.2	Laboratoires	11
3.6	Maîtriser et utiliser les outils d'évaluation des apprentissages adaptés à l'enseignement dispensé et pouvoir répondre de ses choix . . .	12
3.6.1	Évaluation des cours théoriques	12
3.6.2	Évaluation des laboratoires	13
3.7	Porter un regard réflexif et interdisciplinaire sur ses connaissances scientifiques et son enseignement	14
3.7.1	Connaissances scientifiques	14
3.7.2	Enseignement	14
3.8	Actualiser ses connaissances et ses pratiques	15
3.8.1	La lecture	15
3.8.2	Les formations	15
3.8.3	Information via le net	15

3.9	Etre un partenaire actif dans l'organisation et le développement de son institution	15
3.9.1	Le "triumvirat"	16
3.9.2	Rénovation des grilles	16
3.9.3	Responsabilité d'un laboratoire	16
3.9.4	Site internet	16
4	Conclusion	17