



# L'immersion : un pont vers la vie professionnelle

Philippe Saliou, Vincent Ribaud

► **To cite this version:**

Philippe Saliou, Vincent Ribaud. L'immersion : un pont vers la vie professionnelle. Colloque LMD en informatique: Europe et emploi. Montpellier-France, May 2005, Montpellier, France. ACTES DU COLLOQUE LMD EN INFORMATIQUE 2005, 2005. <hal-01448446>

**HAL Id: hal-01448446**

**<http://hal.univ-brest.fr/hal-01448446>**

Submitted on 27 Jan 2017

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

# L'immersion : un pont vers la vie professionnelle

Philippe Saliou, Vincent Ribaud

EA3883, LISyC, Université de Bretagne Occidentale, C.S. 93837 29238 Brest Cedex 3, France

E-mail: {Philippe.Saliou, Vincent.Ribaud } @univ-brest.fr

## Résumé

Depuis Septembre 2002, l'Université de Bretagne Occidentale propose une cinquième année destinée aux étudiant(e)s diplômé(e)s d'un IUP Informatique. L'historique de cette création est décrit dans la section 1.

La section 2 présente le dispositif pédagogique de cette formation d'Ingénierie du Logiciel par Immersion, qui reproduit les réalités professionnelles afin que l'apprentissage du monde du travail soit importé dans les murs de l'université.

Le plan de formation est construit sur un projet de 6 mois en équipe, soutenu par un processus d'apprentissage spécifique, piloté et tutoré par un enseignant-chercheur issu du monde professionnel. Le processus d'apprentissage s'appuie sur les processus de production réellement mis en œuvre chez notre partenaire, la société de services Thales-IS. En dehors des cours de langues et de communication, aucun cours traditionnel n'est dispensé.

L'année de formation est structurée en trois périodes : une période d'apprentissage tutoré de 4 mois, une période de mise en pratique accompagnée de 2 mois, une période de stage en entreprise de 4 à 6 mois.

Pendant les deux premières périodes, les étudiant(e)s, par équipe de 5, sont immergé(e)s dans un véritable projet qui se déroule exactement comme en entreprise, dans une salle simulant l'environnement standard des entreprises de développement logiciel.

Dans la section 3, on met en regard les artefacts d'une société de services, leur translation dans le paradigme d'immersion et les apports que cette translation apporte au dispositif pédagogique. Les points abordés sont les suivants : -1- Activités (ou compétences) d'un ingénieur du logiciel : de l'intérêt d'avoir une formation basée sur les compétences. - 2 - Une organisation industrielle issue de Thales-IS : l'apprentissage de l'autonomie par la reproduction d'un système opérant. - 3 - Un référentiel d'apprentissage mais aussi de production : de la multiple nature des savoirs (faire mais aussi apprendre). - 4 - Une unité de lieu, de temps et de moyens : la garantie d'une situation authentique d'apprentissage. - 5 - Les outils de management au service de l'apprentissage : la gestion et l'amélioration de la performance appliquée au développement des compétences. - 6 - Une plate-forme d'analyse, de conception, de mise en œuvre, de tests et de déploiement en vraie grandeur : la maîtrise du savoir-faire technologique.

En perspective, on s'interroge sur la possibilité d'un chemin inverse : de l'apprentissage par immersion vers l'entreprise et de l'enrichissement que cela pourrait apporter au monde professionnel.

## 1 Historique

L'Université de Bretagne Occidentale (UBO) dispose d'un cursus DEUG-licence-maitrise-DESS en Informatique depuis 1991. En 1998, un IUP GMI Ingénierie Informatique a été créé.

### 1.1 Le DESS d'Informatique

Ce DESS comprenait 542 heures d'enseignement et un projet de 8 semaines (272 h CM, 38 h TD, 232 h TP) dont 447 heures d'informatique et un stage (de 4 à 6 mois).

### 1.2 La maîtrise IUP GMI Ingénierie Informatique

Cette maîtrise comprend 4 UE d'informatique de 120 heures chacune (Systèmes distribués, Programmation-objet et modélisation, Génie logiciel et gestion de projet, Temps réel), deux langues, de l'économie et un stage (de 4 à 6 mois).

### 1.3 Poursuite d'études après l'IUP

La première promotion d'ingénieur-maître (ou de maître) de l'IUP est sortie en juin 2001. L'idée généralement admise était qu'il(elle)s allaient s'insérer dans la vie professionnelle. Cependant, un jury de poursuite d'études délivre des avis.

A l'issue de l'année 2000-2001, 47 étudiant(e)s (sur 58) ont eu leur maîtrise et 25 ont reçu un avis favorable de poursuite d'études. En 2001-2002, il s'est avéré que plus de 30 ont poursuivi des études en DESS (1 seul à Brest en DESS).

### 1.4 Expérimentation d'une adaptation du DESS pour les ex-IUP

Il s'est donc avéré utile de proposer une année supplémentaire après le cursus IUP.

Il existait un très fort recouvrement entre la maîtrise IUP et le DESS Informatique, environ 80 % des cours de DESS ayant été abordés en IUP3, généralement par les mêmes intervenants. Il a donc été possible d'axer les apprentissages de cette cinquième année sur des nouvelles pratiques.

Le débouché principal du DESS est le travail d'ingénieur logiciel dans une société de services en informatique. L'adaptation du DESS pour les étudiant(e)s venant d'IUP visait à faciliter la transition dans le monde du travail en reproduisant, à l'intérieur de l'université, les réalités de la vie professionnelle : cette nouvelle pédagogie a été baptisée Ingénierie du Logiciel par Immersion.

Le conseil d'administration de l'UBO a donné le feu vert pour une expérimentation de deux ans avant le passage au LMD. Les étudiant(e)s venant d'IUP ont été dispensés des matières informatiques (soit 447 heures sur 552).

A l'issue de ces deux années, le principe de l'immersion a été retenu comme une deuxième année de Master Pro IUP.

### 1.5 Coût

Lors de l'expérimentation, l'arrivée de 10 étudiant(e)s a permis d'ouvrir un groupe de TP en plus (232 h TP soit environ 150 ETD pour financer tout le nouveau cursus de 25 semaines) en attendant un financement à la hauteur des autres DESS.

La particularité de l'immersion ne permet pas de raisonner en CM-TD-TP, mais en coût global ETD par étudiant(e). Au 1<sup>er</sup> semestre, les UE d'apprentissage tutoré sont rémunérées selon la formule 1 ECTS – 1 étudiant(e) – 1 ETD, soit 120 ETD pour une compagnie de 5. Au 2<sup>ème</sup> semestre, on a la formule 3 ECTS – 1 étudiant(e) – 1 ETD, soit 60 ETD par compagnie.

### 1.6 Flux

Année universitaire	Inscrit(e)s en maîtrise IUP	Admis(e)s	Avis favorable	Inscrit(e)s en immersion
---------------------	-----------------------------	-----------	----------------	--------------------------

DESS 2002-2003	59	47	31	10
DESS 2203-2004	52	42	24	8 (2 "IUP" extérieurs)
Master 2 Pro IUP 2004-2005	53	43	28	10

## **2 Description du dispositif d'apprentissage par immersion**

Les objectifs du Master par Immersion sont la maîtrise des activités d'ingénierie logiciel, le travail coopératif en équipe, l'adaptation aux changements, l'apprentissage des nouvelles technologies, le développement des Systèmes d'Information.

L'idée principale de cette formation innovante, baptisée « Ingénierie Logiciel par Immersion », est de faire en sorte que les réalités du monde professionnel soit importées dans les murs de l'université. Le dispositif pédagogique reproduit autant que possible les phénomènes que les futurs ingénieurs seront amenés à rencontrer : un environnement de travail professionnel, la relation client-fournisseur, le respect d'un référentiel de développement, l'emploi de méthodes et des outils associés, la coopération entre membres d'une équipe, etc. En dehors des cours de langues et de communication, aucun cours traditionnel n'est dispensé. Le plan de formation est construit sur un projet de 6 mois en équipe, piloté et tutoré par un professionnel expérimenté en ingénierie du logiciel.

L'année de formation est structurée en trois périodes : une période d'apprentissage tutoré de 4 mois, une période de mise en pratique accompagnée de 2 mois, une période de stage en entreprise de 4 à 6 mois.

Les apprentissages d'ingénierie du logiciel portent sur l'ensemble des connaissances, des procédés et des acquis scientifiques et techniques pour la conception et la fabrication d'un Système d'Information de sa commande jusqu'à sa mise en exploitation.

Pendant les deux premières périodes (6 mois), les étudiant(e)s sont immergé(e)s dans un véritable projet qui se déroule exactement comme en entreprise, avec les spécificités suivantes :

- Les étudiant(e)s sont réparti(e)s en équipe de 5 appelée compagnie. Les compagnies sont indépendantes et les étudiant(e)s doivent jouer plusieurs rôles (organisationnel, qualité, technique, soutien, etc.) dans le contexte de collaboration et de coopération lié au travail en équipe.
- Le projet se déroule au sein de l'université, dans une salle dédiée spécialement réaménagée, simulant l'environnement standard des entreprises de développement logiciel. Cette salle comprend un bureau « paysagé » par compagnie, une salle de réunion commune et un espace « Internet ». Chaque étudiant(e) dispose d'un poste de travail personnel.
- Le chef de projet qui pilote et tuteur le projet est un enseignant-chercheur issu du monde professionnel, appelé tuteur de compagnie. Il s'appuie pour cela sur un référentiel d'apprentissage spécifique, construit en partie, à partir du référentiel « métier » de la filiale informatique du groupe Thales, Thales Information Systems.
- Une plate-forme technologique professionnelle (IBM, Oracle, Rational, Microsoft, Open Source) est mise à disposition de chaque compagnie ; et ce en raison de l'importance capitale que revêt l'architecture technique dans la construction des systèmes informatiques actuels.
- Un processus d'apprentissage est intégré dans l'ensemble des processus de production sur lesquels s'appuie la fabrication d'un logiciel : Gestion de Projet, Ingénierie de Développement Logiciel et Soutien au Développement.
- Chaque compagnie construit un Système d'Information différent à partir d'un cahier des charges, avec des processus similaires mais en utilisant des méthodes, des outils et des technologies différentes.
- La troisième période est un stage de pré-embauche (4 à 6 mois). Il s'agit d'un stage de production dont l'objectif est la mise en application des compétences d'ingénierie du logiciel en entreprise. L'étudiant(e) se voit alors confier un petit projet de « A à Z » selon le processus de développement en vigueur dans l'entreprise ou selon le processus de sa formation. Une alternative consiste à effectuer un stage permettant d'approfondir une (voire plusieurs) des activités essentielles d'ingénierie logiciel : gestion de configuration du logiciel, capture des besoins, analyse, conception, etc.

## **3 L'immersion comme sas vers la vie professionnelle**

Dans cette section, on mettra en regard les artefacts d'une société de services (*en italique*) et leur déclinaison dans le paradigme d'immersion.

### **3.1 Activités (ou compétences) d'un ingénieur du logiciel**

#### *3.1.1 Modèle des activités d'une société de services en informatique*

*Les activités d'ingénierie du logiciel généralement admises dans une société de services sont les suivantes :*

- 1. Soutien à l'ingénierie du logiciel : mettre en place les Ateliers de Génie Logiciel (AGL) et soutenir l'utilisation des méthodes et outils*
- 2. Analyse des exigences et conception du système : identifier les exigences du système et concevoir l'architecture du système*
- 3. Analyse des exigences sur le logiciel : établir ce que le logiciel doit faire*
- 4. Conception préliminaire : concevoir l'architecture du logiciel et de ses interfaces*
- 5. Conception détaillée : concevoir les données et les algorithmes du logiciel*
- 6. Codage : traduire la conception détaillée en langage de programmation*
- 7. Test unitaire : définir et exécuter les tests unitaires*
- 8. Intégration et test d'intégration : définir les tests d'intégration, intégrer le logiciel et exécuter les tests*
- 9. Test de validation/qualification du logiciel : établir les prévisions de test de validation/qualification officielle, définir les jeux d'essai et procédures de tests du logiciel et préparer l'environnement de test, installer les logiciels dans les environnements de test, exécuter les tests et évaluer les résultats*
- 10. Documentation de soutien logistique : préparer la documentation*

### 3.1.2 Structure de la formation

Cette vision linéaire des activités est souvent reproduite dans les cours d'ingénierie du logiciel (le modèle de cycle de vie utilisé en V, en cascade ou en spirale n'y changeant rien). Ceci amène les élèves à les percevoir comme une suite de « matières » qui seraient relativement indépendantes les unes des autres et qu'on pourrait aimer ou non (et donc s'en dispenser ou non ...).

Dans le monde du travail, ces activités sont d'abord mises au service de la réalisation de l'affaire<sup>1</sup>. Une affaire est décomposée en lots de travaux tenant compte des produits à livrer, des activités à réaliser et de l'organisation à mettre en place. Une activité se décompose en tâches et la capacité à effectuer une tâche peut s'appeler une compétence.

Dans un organisme, les activités sont regroupées en familles, qui forment souvent la structure de l'organisation : management, études, soutien, exploitation, ...

Notre structure d'activités est donc organisée comme suit :

- les activités se répartissent en 3 domaines principaux (les familles) qui peuvent être assimilés à des sous-processus de fabrication<sup>2</sup> et donc à des spécialisations du métier d'ingénieur logiciel (cf. figure 1) ;
- les activités et leurs tâches (donc les compétences utilisées) sont situées dans un projet réel, fourni en deux lots.

Il est à noter que cette organisation est celle du curriculum informatique du 1<sup>er</sup> semestre : les UE correspondent aux domaines et les EC aux activités. Le 2<sup>ème</sup> semestre ne comprend que 2 UE : Mise en Pratique Accompagnée (MPA) et stage. MPA correspond au deuxième lot. L'organisation des activités du deuxième lot est déléguée aux élèves : à elles et à eux d'adapter la structure du 1<sup>er</sup> lot (établi lors du 1<sup>er</sup> semestre) aux contraintes de production du 2<sup>ème</sup> lot, donc de mobiliser leurs compétences à bon escient et au bon moment.

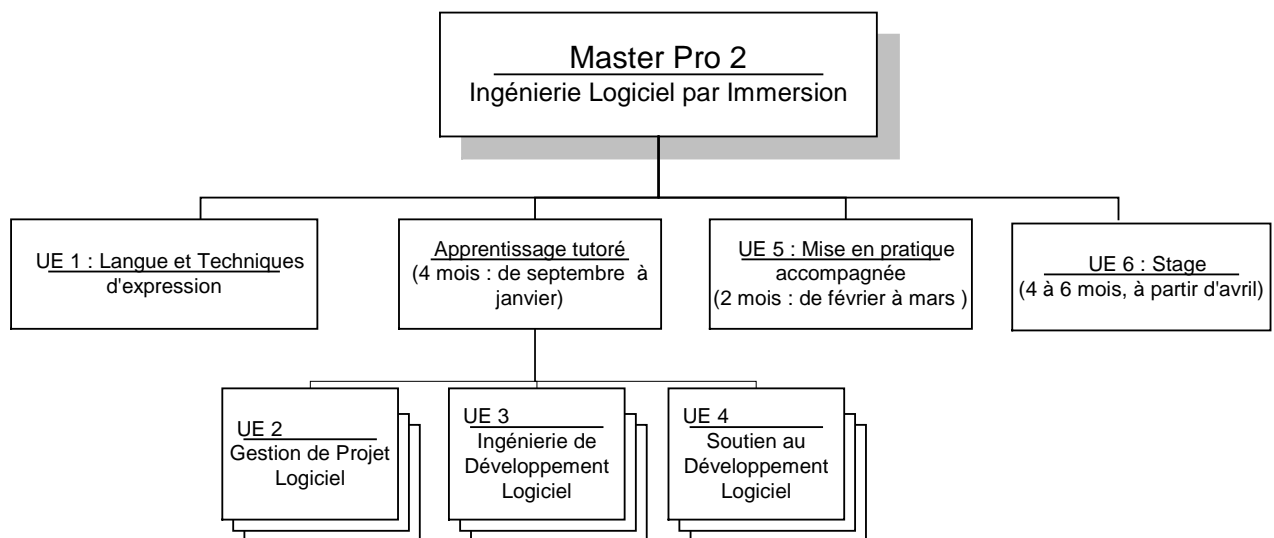


Figure 1 : Structure du curriculum

Le tableau ci-dessous donne la décomposition des EC (activités) dans les UE (domaines).

UE	Désignation de l'EC	ECTS
Langue et Techniques d'Expression (6 ECTS)	Anglais	4
	Techniques d'Expression	2
Gestion de Projet Logiciel (5 ECTS)	Conduite De Projet Logiciel	2
	Assurance Qualité Logiciel	1
	Gestion de Configuration du Logiciel	2
Ingénierie de Développement Logiciel (13 ECTS)	Capture Des Besoins	2
	Architecture Technique	3
	Analyse	2
	Conception du Logiciel	2
	Codage - Tests Unitaires	2
	Intégration – Qualification	2
Soutien au Développement Logiciel (6 ECTS)	Support Technique	1
	Support Méthodes et Outils	2
	Documentation Utilisateur	2
	Installation - Déploiement	1
Mise en Pratique Accompagnée	Mise en Pratique Accompagnée	10
Stage en entreprise	Stage en entreprise	20

<sup>1</sup> Une affaire est l'ensemble des fournitures et des prestations (travaux ou services) définies par un contrat ou une commande.

<sup>2</sup> Il existe un quatrième domaine ou sous-processus qui est diffus dans toutes les activités : la relation client-fournisseur.

## **3.2 Une organisation industrielle issue de Thales-IS**

### **3.2.1 Rôles et responsabilités d'un service informatique**

*Un modèle simplifié de l'organisation des rôles et des responsabilités de la société de services Thales Information System est donné dans les paragraphes suivants.*

#### **3.2.1.1 Le Chef de Service (CS)**

*Le Chef de Service (CS) adopte les engagements contractuels de la société et nomme, en accord avec le RCA, les Responsables de Lot Logiciel (RLL) auxquels il délègue la réalisation des travaux spécifiés par le RCA. Il s'assure que les conditions initiales de lancement des travaux sont remplies et il fournit au RLL les moyens matériels et les ressources humaines nécessaires à l'accomplissement de sa tâche. Il veille tout au long du déroulement de l'affaire au respect des engagements de chacun et arbitre les conflits éventuels entre le RCA et les RLL.*

#### **3.2.1.2 L'Ingénieur Commercial (ICL)**

*L'Ingénieur Commercial (ICL) est responsable de la prospection commerciale et de la mise en œuvre des accords commerciaux. Il assure la préparation des propositions commerciales, la négociation des contrats en respectant les objectifs de qualité et de rentabilité requis ainsi que le suivi des affaires en cours jusqu'à leur terminaison.*

#### **3.2.1.3 Le Responsable Chargé d'Affaire (RCA)**

*Le Responsable Chargé d'Affaire (RCA) conduit la réalisation de l'affaire, depuis sa notification jusqu'à sa clôture, en garantissant la tenue des engagements contractuels (performances, délais) et le résultat financier prévu.*

#### **3.2.1.4 Le Responsable de Lot Logiciel (RLL)**

*Le Responsable de Lot Logiciel (RLL) conduit et maîtrise les activités nécessaires à la réalisation du lot qui lui est confié. Il :*

- *est responsable de la qualité du logiciel et planifie l'ensemble des activités y ayant trait,*
- *est responsable du délai, des charges allouées et de l'élaboration des rapports d'avancement du lot,*
- *est responsable de la conformité des produits et des processus de production de son lot,*
- *est responsable de l'engagement et du suivi des actions de formation nécessaires pour les membres de l'équipe logiciel,*
- *est responsable de la bonne utilisation des outils et des méthodes choisis pour le développement du logiciel,*
- *oriente les choix techniques qu'il prend au nom du métier et les fait valider par le RCA et/ou le CS,*
- *pilote l'équipe logiciel et en cas de conflit avec le RCA, demande l'arbitrage du CS.*

#### **3.2.1.5 L'équipe logiciel**

*Le RLL est assisté d'une équipe logiciel qu'il dirige opérationnellement. Il peut déléguer certaines de ses prérogatives comme par exemple les fonctions de :*

- *Responsable de la Gestion des Configurations Logiciel, désigné pour le lot et chargé de la définition et la mise en œuvre du Plan de Gestion des Configurations du Logiciel ;*
- *Responsable de l'intégration et des tests, désigné pour le lot et chargé de suivre les composants élémentaires, leur intégration en composants logiciels et la qualification des composants logiciels au niveau système.*

*Chaque réalisateur de l'équipe :*

- *a la responsabilité de composants du logiciel et s'engage sur le respect des exigences qui sont allouées à ces composants,*
- *participe par le biais des lectures croisées, aux évaluations des produits planifiées dans les plans du lot,*
- *connaît et respecte les principes fondamentaux et les règles de gestion des configurations mises en œuvre pour le lot,*
- *connaît les objectifs de charges, de délais et de qualité assignés par le RLL et s'engage à les respecter,*
- *assure le rapport de l'avancement de ses travaux vers le RLL.*

#### **3.2.1.6 L'Ingénieur Qualité (IQ)**

*La mission de l'Ingénieur Qualité (IQ) consiste à s'assurer de la conformité (par rapport aux plans du lot) des produits et des processus de production du logiciel.*

#### **3.2.1.7 Les groupes support**

*Il s'agit des groupes qui peuvent intervenir sur une affaire à des titres divers. On trouve généralement :*

- *l'équipe-système pour la fourniture et l'installation des matériels et des licences des logiciel de base,*
- *le support méthodes et outils pour le support à la mise en œuvre des procédures, des guides et des outils.*

### **3.2.2 Rôles et responsabilités dans l'apprentissage par immersion**

*Une description détaillée des rôles des enseignants de notre dispositif est donnée en [RS04]. A défaut de trouver les termes adéquats, on distinguera rôles et responsabilités : le rôle se rapporte au personnage tel qu'il est perçu par l'institution et les élèves alors que les responsabilités sont les missions nécessaires au bon fonctionnement du dispositif. Un rôle assume le plus souvent de multiples responsabilités et certaines responsabilités sont réparties entre plusieurs rôles.*

### 3.2.2.1 Rôles

Le dispositif par immersion distingue trois rôles :

- le « **Tuteur-Coordinateur** » est le responsable de la formation vis-à-vis de l'université, de l'équipe pédagogique et des élèves. Comme tout responsable de formation, son travail s'effectue tout au long de l'année (organisation, gestion et suivi de la formation). Cependant, dans notre dispositif, il existe des spécificités de son travail avant le début de la formation (donner le cadre des projets, préparer la logistique, mettre à jour et mettre à disposition le référentiel d'apprentissage), pendant la première et la deuxième période (ajuster et maintenir le référentiel, gérer les ressources, conduire et tutorer l'équipe de tuteurs), pendant la troisième période (suivi des stages certes mais surtout ambassadeur du dispositif auprès des entreprises) et à la fin de la formation (bilan, planification d'ajustement et préparation de la rentrée suivante).
- le « **Tuteur de compagnie** » est l'authentique chef de projet de la compagnie. Son travail s'effectue avant le début de la formation (rédiger le cahier des charges ainsi que la proposition technique et financière), pendant la première période (coordonner la compagnie, réguler les apprentissages, développer les compétences), pendant la deuxième période (surveiller le bon déroulement, arbitrer et résoudre les situations de blocage) et tout au long de la formation en ce qui concerne l'évaluation.
- le « **Tuteur-Support d'activité** » accompagne une activité d'ingénierie. Son travail s'effectue principalement pendant la première période (formuler les objectifs de l'activité, assister la production). C'est une personne-ressource qui fournit un support à l'activité, relativement indépendamment du dispositif d'apprentissage.

### 3.2.2.2 Responsabilités

Sous un autre angle, on peut distinguer plusieurs types de responsabilités (ou d'activités pour les enseignants) spécifiques à l'immersion :

Responsabilités	Description
Responsable du référentiel	Crée, consolide, adapte et enrichit le référentiel d'apprentissage : dispositif (principes, règles, séquençement, etc.), fiches d'apprentissage avec les produits et ressources pédagogiques, logiciels de modélisation et de développement (Oracle, IBM, Rational et Open Source), ouvrages, documentation, etc.
Responsable logistique	Achat et suivi du matériel. Suivi et renouvellement des contrats avec les éditeurs. Relation avec Thales-IS. Approvisionnement des outils et des licences, ouvrages, etc.
Tuteur de tuteur	Accompagnement et assistance aux tuteurs de compagnie et tuteurs-support d'activité : régulateur, modérateur et/ou médiateur.
Tuteur-Organisateur	Définit le cadre projet spécifique à sa compagnie : besoins matériels et logiciels, choix techniques et fonctionnels, cahier des charges, proposition technique et financière.
Tuteur-Chef de projet (Tuteur-RLL)	Organise chaque séquence d'apprentissage (ajustement, élaboration et régulations des parcours et des apprentissages individuels et collectifs). Soutient et assiste au quotidien individuellement chaque étudiant(e). Organise la réunion d'avancement hebdomadaire.
Tuteur-Client	Maîtrise d'ouvrage au travers d'un jeu de rôles pour les activités de capture des exigences, d'analyse, d'installation-déploiement, de recette, de maintenance (bogues, demande d'évolution), etc.
Tuteur-Modérateur <sup>3</sup>	Consolidation des produits de qualité moyenne (voire médiocre).
Tuteur-Responsable chargé d'affaires	Responsable hiérarchique qui arbitre, régule, voire impose (respect des coûts et délais, arbitrage « hiérarchique » entre élèves).
Tuteur d'apprentissage	Elabore et fournit le contexte et les ressources pédagogiques en lien avec le métier. Accompagne et entraîne chaque étudiant(e) (Coaching). Conseille, réoriente, aide, régule voire arbitre en cas de blocage. Evalue les apprentissages effectués (ou les compétences développées).

<sup>3</sup> « Le modérateur est un expert extérieur à l'affaire capable d'apporter un regard impartial sur les aspects fonctionnels et techniques du projet. Il intervient essentiellement comme un pair sur lequel le chef de projet ou les membres de l'équipe peuvent se reposer pour conforter leurs choix [TIS02]. »

### 3.2.3 Rôles perçus par les élèves

Un des aspects les plus troublants de ce dispositif est que, malgré sa complexité, il semble naturel aux élèves. Comme il reproduit une organisation professionnelle, les élèves perçoivent et intègrent essentiellement les rôles professionnels décrits au § 3.2.1, ainsi que le rôle du client puisqu'il est présent tout au long des deux périodes. Quand aux rôles et responsabilités spécifiques au dispositif (cf. § 3.2.2), les élèves n'en perçoivent que la partie émergée (le logisticien, le chef de projet, le client, etc.) c'est-à-dire ceux qui sont mis en scène dans la simulation qui soutient le processus d'apprentissage.

## 3.3 Un référentiel d'apprentissage mais aussi de production

### 3.3.1 TEMPO, le référentiel de Thales-IS France

TEMPO est un ensemble cohérent de composants de type documents, formations et outils qui permet à Thales-IS France de définir d'une part ses engagements de gestion et de prestations, d'autre part de fournir à son personnel une description des moyens à mettre en œuvre pour obtenir la satisfaction de ces engagements.

Les documents du référentiel TEMPO sont organisés en niveaux. Au sommet se trouve le Manuel Assurance Qualité (MAQ), puis le niveau procédures, et enfin le niveau instructions, documents techniques détaillant la mise en œuvre des procédures.

#### 3.3.1.1 Le Manuel d'Assurance Qualité (MAQ)

Il est structuré selon les items de l'ISO 9001. Il propose une vision synthétique globale des actions qualité à Thales-IS France. Pour les détails des actions à conduire, il s'appuie sur les procédures du référentiel.

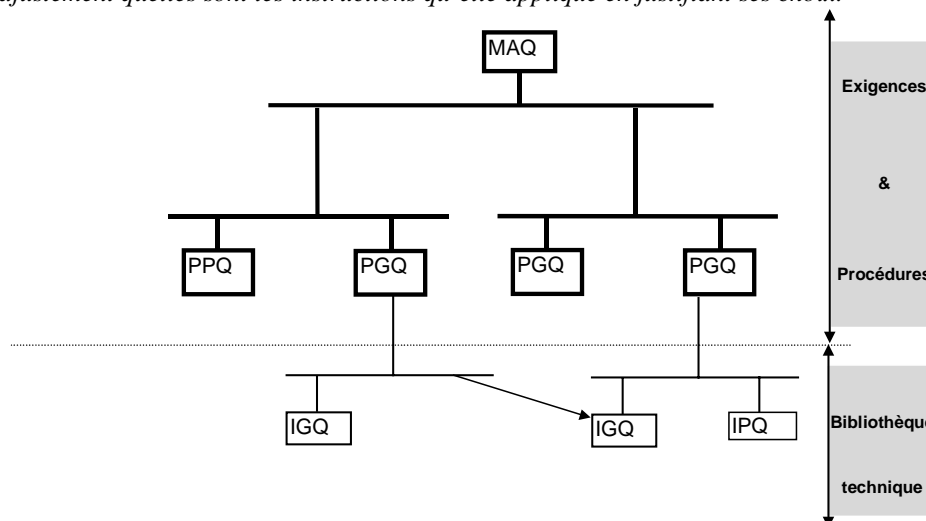
#### 3.3.1.2 Les Procédures Générales (et Particulières) Qualité (PGQ ET PPQ)

Elles sont la base de la description d'un processus et des activités associées. Afin d'avoir la plus grande stabilité possible, elles abordent le QUOI, fait QUAND et PAR QUI. La procédure fixe les exigences sur le processus. Les procédures sont applicables, mais afin de répondre aux spécificités des affaires, chaque procédure donne des éléments d'ajustement du processus. Pour une affaire donnée, la procédure une fois ajustée, décrit les engagements que prend l'affaire sur la mise en œuvre du processus concerné.

#### 3.3.1.3 Les Instructions Générales (et Particulières) Qualité (IGQ ET IPQ)

Le plus souvent appelées " Guide ", ce sont des recommandations de mise en œuvre qui traitent du COMMENT et du AVEC QUOI. En règle générale, chaque instruction est rattachée à au moins une procédure. Une procédure peut avoir plusieurs instructions. Afin de répondre au mieux à la diversité des affaires, les instructions, à l'image des procédures, peuvent être ajustées à chaque affaire.

Certaines instructions sont applicables, certaines ne sont que d'utilisation conseillée : dans ce cas, chaque affaire précise dans son dossier d'ajustement quelles sont les instructions qu'elle applique en justifiant ses choix.



La " partie haute " de l'arborescence, formée par le MAQ et les PGQ (PPQ), traite des exigences et engagements, elle constitue la partie " Exigences et procédures " du référentiel. La " partie basse ", qui propose des recommandations de mise en œuvre, constitue la bibliothèque technique du référentiel.

### 3.3.2 Référentiel d'apprentissage

Notre objectif a été de définir un cadre de référence devant répondre aux attentes de tous les « co-apprenants » que sont les élèves et les enseignants mais aussi de tous les autres intervenants impliqués dans la formation.

Durant les 2 premières années où la formation était expérimentale, nous avons élaboré une première version d'un référentiel d'apprentissage du métier d'ingénieur logiciel en nous appuyant sur 2 choix pédagogiques essentiels :

- L'emploi de procédés de fabrication en phase avec les environnements actuels de développement logiciel : UML 1.3 [BRJ98] et le Processus Unifié [JBR99]. Cette notion de procédés recouvre tous les aspects de production d'un logiciel tant sur les aspects méthodiques, techniques que sur les outils utilisés.

- L'utilisation du référentiel société mis à notre disposition par la filiale informatique du groupe Thales [TIS02]. Parmi la somme de matériaux disponibles, nous avons surtout retenu l'aspect « guide des bonnes pratiques » de l'ingénieur logiciel mais aussi quelques documents d'organisation comme les règles de présentation et de gestion des documents ou le formulaire d'entretien de performances et de fixation des objectifs.

Nous avons fusionné l'ensemble de ces éléments pour constituer notre référentiel d'apprentissage. Ce référentiel d'apprentissage soutient d'une part le découpage temporel de l'année en trois périodes, qualifiées d'itérations et d'autre part le processus d'apprentissage mis en œuvre lors des deux premières périodes.

Le référentiel d'apprentissage comprend à ce jour :

- Un référentiel métier TEMPO-ILI comprenant :
  - Les éléments issus de l'ajustement du référentiel Thales-IS.
  - Des références à d'autres référentiels de type ISO 9000:2000.
  - Des exemples de documents produits sur des projets réels
- Des supports d'utilisation des plates-formes techniques.
  - Guides d'installation et d'utilisation
  - White Papers
  - Red Books
- Des ouvrages ainsi que des supports d'auto-formation.
- La description du processus d'apprentissage constitué principalement :
  - Des fiches d'apprentissage définissant étape par étape le travail à réaliser.
  - Des fournitures pédagogiques mises à disposition au fur et à mesure de l'avancement du projet.

Chaque activité d'ingénierie se décompose en tâches. L'apprentissage d'une tâche est décrit dans une fiche d'apprentissage. Elle décrit le travail à réaliser, les différentes fournitures pédagogiques sur lesquelles doit s'appuyer l'apprentissage ainsi que les produits attendus. Un exemple est donné dans la figure 2.

<b>Numéro</b> : 8	<b>Date</b> : 21/11/2004	<b>Origine</b> : Ph. SALIOU	<b>Action</b>		
<b>Affaire</b> : ILI-Lot1	<b>FICHE D'APPRENTISSAGE</b>		Ingénieur Analyste	<b>V. Turtle</b> <b>W. Harrow</b>	
<b>Domaine / UE</b> : Ingénierie du Développement du Logiciel (IDL) <b>Activité / EC</b> : Capture des besoins (CB)			<b>Nom</b> : Consolidation des exigences		
<b>DESCRIPTION DU TRAVAIL</b>					
<p><b>Ce travail</b> a pour objet de consolider dans un document unique l'ensemble des exigences à satisfaire par le logiciel. Ce travail répond à plusieurs objectifs :</p> <p>Etre conforme au référentiel TEMPO-ILI malgré un processus 2TUP (2-Track Unified Process). Rapprocher les 2 points de vue : besoins fonctionnels et besoins techniques.</p> <p>...</p> <p><b>Le résultat attendu</b> se matérialisera sous la forme d'une Spécification des Exigences Logiciel (SEL). Cette spécification constitue un document de référence pour la conception du logiciel.</p> <p>...</p> <p><b>Les ressources pédagogiques</b> suivantes peuvent vous venir en aide à la rédaction de la SEL: Guide de rédaction simplifié pour la spécification des exigences sur le logiciel (TEMPO-IGQ347). Exemples de SEL</p> <p>...</p>					
<b>Produits</b>			<b>Version</b>	<b>Dates clés</b>	
Spécification des Exigences du Logiciel (SEL)			A	03/12/2004	
<b>ETAT D'AVANCEMENT</b>					
<b>Estimé</b>			<b>Réel</b>		
<b>Date de début</b>	<b>Date de fin</b>	<b>Charge</b>	<b>Date de début</b>	<b>Date de fin</b>	<b>Consommation</b>
25/11/2004	03/12/2004	3,5 j   3,5j			
<b>Date</b>	<b>Consommation</b>	<b>Livraisons-observations</b>			

Figure 2 : Fiche d'apprentissage



Sa structure est standardisée. Les éléments principaux sont :

- la tâche (ici la consolidation des exigences) relative à l'activité (et au domaine),
- le rôle à jouer (ici analyste) avec le noms des élèves concernés,
- la description du travail (ici une tâche de consolidation),
- les fournitures pédagogiques (ici un guide de rédaction et des exemples réels),
- les produits de sortie à réaliser (ici une Spécification des Exigences du Logiciel),
- des éléments de charge et de planning .

L'élève perçoit donc la fiche d'apprentissage comme un élément essentiel d'organisation de ses compétences. Par exemple, au sein du domaine d'ingénierie du développement, l'activité « capture des besoins » se décompose en capture des besoins fonctionnels, capture des besoins techniques et consolidation des exigences. Pour chaque tâche, l'élève peut associer les produits amonts issus d'activités précédentes, les guides de bonnes pratiques ayant soutenu la tâche, sa place dans le processus de production, les produits de sortie de la tâche, etc.

Lors de la deuxième itération, les étudiant(e)s vont s'appuyer sur le référentiel d'apprentissage de la première itération. Ce référentiel d'apprentissage va alors être utilisé comme un référentiel de développement logiciel.

### 3.4 Une unité de lieu, de temps et de moyens

#### 3.4.1 Exemple d'une affaire d'infogérance

*Il n'existe aucune organisation-type qui s'appliquerait quelque soit l'affaire à réaliser. Au mieux, on trouve une classification des affaires par type comme l'infogérance, l'intégration de systèmes, l'ingénierie des systèmes d'information de gestion (à l'image des modèles du CMMI : Systems Engineering, Software Engineering, Integrated Product and Process Development, Supplier Sourcing). A titre d'exemple, on montre quelques éléments d'une procédure de conception et lancement d'une affaire d'infogérance.*

*La conception et la prise en charge d'une affaire d'infogérance d'administration et d'exploitation se décompose en trois étapes principales :*

- 1) *l'étape d'analyse des exigences, de spécification et de planification*
  - prise de connaissance du système d'information du client et de son environnement,
  - spécification détaillée du service à fournir au client et de ses critères d'acceptation,
  - découpage de l'affaire en lots opérationnels,
  - planification de l'ensemble de la phase de conception (spécification, réalisation, vérification, validation et acceptation ),
  - spécification de la prestation de service lot par lot,
  - spécification du contrôle qualité,
    - a. définition des moyens et méthodes de mesure des indicateurs de performance,
    - b. définition des moyens et méthodes de mesure des indicateurs de niveaux de service ;
  - vérification des spécifications ( conformité aux exigences).
- 2) *l'étape de réalisation avec éventuellement une étape de transferts des moyens*
  - transferts,
    - a. transfert des licences, droits d'utilisation et des contrats des logiciels et progiciels,
    - b. transfert des contrats liés aux matériels,
    - c. transfert de personnels,
    - d. transfert administratif,
    - e. transfert des équipements, des documents et des données ;
  - développement et mise en place de l'ensemble des moyens, méthodes et instructions à mettre en œuvre pour assurer la prestation de service,
  - développement et mise en place de l'ensemble des moyens, méthodes et instructions à mettre en œuvre pour s'assurer de la qualité de la prestation et du service,
  - vérification de la reprise des compétences, des installations et des fonctionnalités par domaines ou parties du système d'information et échelonnée dans le temps,
  - tests et validation du service ( en interne).
- 3) *l'étape de qualification , de validation , approbation du service et recette*
  - établissement du cahier de recette,
  - rédaction du Plan d'Assurance Qualité,
  - approbation par le client du cahier de recette et signature du Procès Verbal de recette (validation d'aptitude - VA )
  - approbation par le client des indicateurs définis dans la Convention de Services suite aux campagnes de mesures effectuées pendant un temps d'observation défini d'un commun accord (validation de service régulier - VSR).

*La phase de prise en charge débute à la signature du contrat et se termine à l'issue de l'acceptation de la recette et de la validation du service régulier par le Client.*

### 3.4.2 Processus d'apprentissage et de production

Le fil conducteur de l'organisation est le projet<sup>4</sup>. L'exemple du paragraphe précédent décrit une procédure organisationnelle (ou ? : chez le client/le fournisseur, quand ? : les étapes, qui ? : le client/le fournisseur, quoi ? : les activités de spécification, de transfert, etc.). On peut aussi citer la définition de la norme ISO 10006 (version 2003) reprise par l'AFNOR (norme X50-105) : « Le projet est un processus unique qui consiste en un ensemble d'activités coordonnées et maîtrisées, comportant des dates de début et de fin, entrepris dans le but d'atteindre un objectif conforme à des exigences spécifiques, incluant des contraintes de délais, de coûts et de ressources [X50-105]. »

Un dispositif axé sur la transférabilité des compétences impose une révision systématique des pratiques pédagogiques et évaluatives. Pour Jacques Tardif, l'organisation scolaire dresse des barrières difficilement franchissables, quelles soient curriculaires ou liées à l'organisation du travail. La division disciplinaire nuit considérablement à la transférabilité des connaissances et des compétences développées par les élèves en milieu scolaire et elle limite énormément les enseignants dans la création d'environnements axés sur le transfert des apprentissages. De plus, l'organisation du travail en milieu scolaire divise les tâches et les horaires des enseignant(e)s et des élèves comme s'il s'agissait de travailleurs spécialisés sur une chaîne de montage en fixant des lieux, des durées et actions ponctuelles pour chaque discipline [Tar99].

En réunifiant le lieu, le temps et les moyens (à l'image du projet/affaire du monde professionnel), le système par immersion fournit un processus d'apprentissage. Ce processus est guidé par le cycle de développement logiciel retenu et possède les caractéristiques suivantes :

- Il se matérialise sous la forme d'un enchaînement de séquences d'apprentissage avec des objectifs clairs et contrôlés à chaque séquence.
- A chaque séquence correspond un certain nombre de tâches décrites par des fiches d'apprentissage établissant les apprentissages visés et/ou les compétences à mobiliser.
- Chaque fiche d'apprentissage est affectée à un ou plusieurs étudiant(e)s qui doivent alors assumer le métier (ou rôle) inhérent à l'activité concernée.
- Chaque séquence met à disposition des fournitures pédagogiques que les étudiant(e)s doivent s'approprier.
- Pour chaque fiche d'apprentissage, les étudiant(e)s peuvent s'appuyer au quotidien sur l'assistance de tuteurs d'apprentissage (qui est, la plupart du temps, le tuteur de compagnie).
- Chaque fiche d'apprentissage décrit la forme et le contenu des produits attendus. Elle constitue le support principal de l'évaluation.

On peut représenter ce processus dans un tableau à deux dimensions comme l'illustre la figure 3.

Les 3 colonnes représentent les 3 domaines principaux d'activités couverts par la formation. Ces 3 colonnes peuvent également être perçus comme 3 sous-processus d'apprentissage complémentaires et fortement imbriqués.

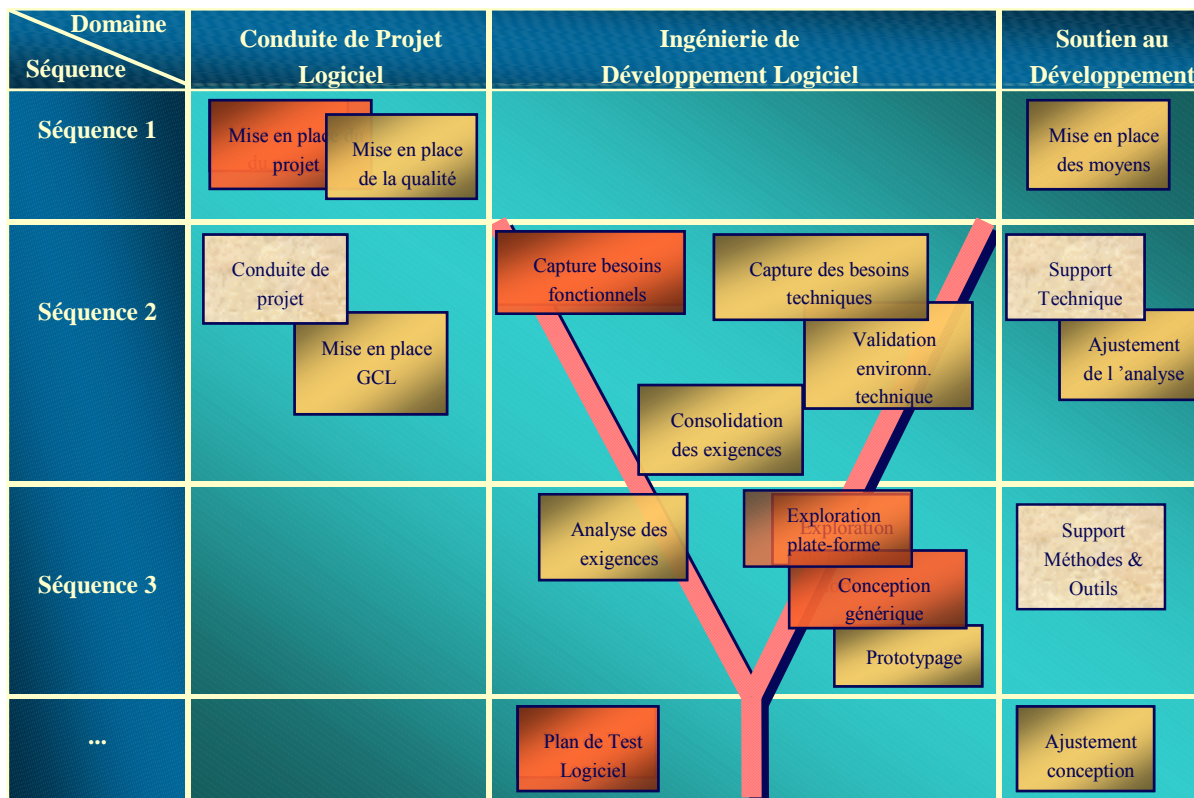


Figure 3 : Processus d'apprentissage

<sup>4</sup> L'affaire est le terme employé dans le paragraphe précédent, mais il signifie un projet défini dans le cadre d'un contrat.

L'axe vertical de ce processus représente le temps. Il permet de matérialiser l'enchaînement des séquences et des jalons essentiels à l'apprentissage de l'ingénierie du logiciel. La durée moyenne d'une séquence est de 2 semaines en fonction des travaux mis en œuvre dans la séquence.

Le parcours de formation comprend deux cycles (appelés itérations) du processus.

La première itération est fortement tutorée ; on pourrait aussi la qualifier de « mentorée » ou « entraînée » (les termes anglais « mentoring » et « coaching » seraient plus signifiants). Elle est avant tout destinée à l'apprentissage et les élèves la perçoivent très vite comme une occasion unique où il(elle)s ont droit à l'erreur (l'évaluation est avant tout formative).

Une organisation fixe du travail est mise en place pour la deuxième itération. L'objectif de la deuxième itération est la production et la mise en exploitation du système d'information complet tel que demandé dans le cahier des charges. Les élèves vont s'appuyer sur les fiches d'apprentissage de la première itération pour mettre en œuvre (de manière autonome) un processus de développement logiciel en phase avec les choix d'architecture technique, les outils de développements utilisés ainsi que les contraintes de délais imposés par le client.

Ainsi, la formation est construite autour d'étapes d'apprentissage et de réalisation, mobilisées dans les deux itérations. Elle permet :

- de prendre en compte le rôle structurant du temps et son intensité dans le monde du travail. L'apprentissage de la capacité à hiérarchiser les séquences, à maîtriser un processus global de travail sera mobilisée au-delà de cette activité précise.
- d'établir des relations coopératives entre apprenants ce qui conduit à alterner entre des temps individualisés de travail en équipe et des temps de véritable travail d'équipe.

### **3.5 Les outils de management au service de l'apprentissage**

Une organisation est dotée d'un ensemble d'outils de management en rapport avec la gestion des ressources humaines (GRH). Le dispositif d'immersion ne cherche pas à reproduire tous les processus de GRH, mais ceux liés à la gestion et au transfert des compétences sont progressivement intégrés dans le dispositif. Nous illustrons ce propos avec deux expérimentations réalisées cette année :

- la revue de pairs comme support au transfert de compétences inter-étudiants,
- l'entretien individuel de performances mis au service du développement de compétences.

#### **3.5.1 Revues de pair**

##### **3.5.1.1 Objectifs**

*Le but des revues de pairs est de détecter et d'éliminer, précocement et efficacement, les défauts des produits au cours de leur développement. En effet, tout défaut non détecté au cours d'une phase induira sa propagation dans les autres produits du cycle de développement et un travail de reprise ultérieure.*

*D'autre part, les revues de pairs sont un moyen de favoriser les transferts de connaissance et d'expertise entre les participants.*

*Les activités de revues de pairs s'insèrent dans l'assurance de la qualité.*

*Les revues de pairs collectent plusieurs types d'informations :*

- les données sur les produits revus,
- les données sur les défauts,
- les données sur le processus de revue de pairs.

*Ces données vont permettre, d'une part, de corriger les défauts dans les produits du développement et, d'autre part, de constituer les données de capitalisation.*

##### **3.5.1.2 Déroulement**

*La réalisation d'une revue de pairs sur un produit comprend :*

- la préparation individuelle,
- la concertation entre les pairs et l'auteur,
- la collecte des défauts,
- le suivi et la clôture de la revue.

##### **3.5.1.3 Supports**

###### **3.5.1.3.1 Fiche de relevé des défauts**

*La fiche de relevé des défauts est utilisée pour :*

- la rédaction des défauts (en préparation individuelle et en réunion de revue),
- la classification des défauts,
- l'accord des pairs sur le défaut (à corriger, non retenu, décision différée),
- la correction et le suivi des défauts.

###### **3.5.1.3.2 Fiche de synthèse de revue de pairs**

*La fiche de synthèse de contient les informations suivantes :*

- l'identification de la revue : caractéristiques de l'affaire et du produit,
- le déroulement de la revue : les participants, les objectifs des pairs en préparation individuelle (points de vue, périmètre, critères de lecture), les activités (dates et temps passés),
- la synthèse des défauts majeurs (nombre et classification),
- toutes informations utiles sur le bilan de la revue,
- les signatures de clôture de la revue.

### 3.5.2 Revue de pairs « Transfert de compétences »

Une description détaillée de cette expérience est donnée en [SR05].

Nous étendons la revue de pairs telle que pratiquée dans l'industrie à une revue de pairs destinée au transfert de compétences. Habituellement, les informations collectées lors d'une revue de pairs portent essentiellement sur les défauts détectés. Dans notre système, ces informations sont étendues aux suggestions, propositions d'amélioration et surtout aux interrogations ou compléments d'apprentissage que les étudiant(e)s « pairs » souhaiteraient avoir face aux productions de leurs camarades.

Pour chaque revue de pairs, les activités minimales et obligatoires pour réaliser une revue de pairs sur un produit sont :

- La préparation et l'organisation de la revue par le tuteur de compagnie.
- La collecte par les étudiant(e)s-pairs des interrogations, remarques, propositions d'amélioration voire défauts potentiels.
- La réunion de revue qui a pour finalité la concertation entre les pairs et le(s) auteur(s). Cette réunion sera systématiquement pilotée par le tuteur de compagnie et donnera lieu à la rédaction d'un bilan de revue par les pairs.

La revue de pairs « Transfert de compétences » telle que déclinée dans notre système d'immersion a la particularité d'être à la fois :

- un outil de contextualisation du transfert de compétences entre étudiant(e)s ;
- un outil issu du monde « réel », perçu et accepté comme tel par les étudiant(e)s ;
- un outil d'auto-évaluation coopérative pouvant se substituer en partie à l'évaluation formative.

Elle a également favorisé l'implication individuelle des étudiant(e)s, en renforçant leurs désirs de savoirs et de faire savoir, tout en augmentant leur coopération.

### 3.5.3 Entretiens individuels de performance

#### 3.5.3.1 Définition

*Fixer des objectifs et apprécier les résultats de ses collaborateurs font partie intégrante du rôle du manager chez Thales-IS. Les principales caractéristiques du processus d'entretien de performances et de fixation des objectifs sont :*

- *La définition d'objectifs possibles cohérents avec les objectifs d'équipe.*
- *L'engagement d'éventuelles actions pour favoriser le développement professionnel du collaborateur.*
- *Le système de notation associé à cet entretien permet une appréciation qualitative des résultats.*

#### 3.5.3.2 Déroulement de l'entretien

*Les objectifs fixés au cours de cet entretien doivent être limités en nombre (max. 5). Ils doivent être précis, mesurables et donner un calendrier de réalisation ainsi que les moyens nécessaires à leur mise en œuvre.*

*L'appréciation des performances doit être basée sur les objectifs fixés lors de l'entretien précédent : on évalue des résultats, pas des individus.*

### 3.5.4 Entretiens individuels « Développement de compétences »

La deuxième itération étant axée sur la production du logiciel, l'évaluation porte sur les produits réalisés. Ces produits résultant d'un travail d'équipe, nous avons imaginé utiliser la technique d'entretien pour apprécier la performance individuelle. Deux entretiens (d'environ une heure et demie) ont donc été réalisés : un au début de la première itération pour fixer les objectifs individuels pour la deuxième itération, définir les moyens nécessaires pour atteindre ces objectifs et définir les indicateurs de mesure de performances ; le second ayant pour but d'évaluer la performance basée sur les objectifs fixés lors du premier entretien.

Il nous est vite apparu que l'aspect « évaluation individuelle des performances » pouvait mettre en danger la dynamique de groupe sous-jacente à la formation. Pratiquement aucune évaluation sommative n'est effectuée dans notre dispositif et de manière naturelle, l'entretien individuel de performances (le deuxième) est devenu plutôt un entretien de développement de carrière<sup>5</sup> : il doit permettre soit de clarifier une ou plusieurs pistes d'évolution possibles, soit d'identifier ou construire un projet professionnel.

Cependant, le premier entretien a permis de définir des objectifs individuels et collectifs dans une logique de progrès et de s'accorder sur des actions de développement qui pourront être utilisés dans les années à venir.

Il reste beaucoup de travail pour en faire un outil réellement utile au dispositif, il a cependant été bien apprécié par les élèves qui l'ont perçu comme un entraînement aux techniques d'évaluation en vigueur dans le monde professionnel.

## 3.6 Une plate-forme d'analyse, de conception, de mise en œuvre, de tests et de déploiement en vraie grandeur

Le dernier point, mais pas le moindre, de cet article traite de l'environnement de travail de l'immersion.

L'architecture technique revêt une importance capitale dans la construction des systèmes informatiques actuels. Chaque compagnie dispose donc d'une plate-forme professionnelle, que les étudiant(e)s ne maîtrisent que partiellement en début de projet.

Le choix de ces plate-formes s'est fait en fonction de plusieurs objectifs pédagogiques :

- Mettre à disposition des outils et des frameworks techniques en phase avec les attentes du marché, notamment le développement de systèmes d'information dans une architecture « N-tier ».
- Confronter les étudiant(e)s à la complexité des infrastructures de déploiement et de développement actuelles.

---

<sup>5</sup> Il a cependant été utilisé pour faire prendre conscience à un unique étudiant que son engagement était en dessous de celui requis par le dispositif et le préparer à un impact sur l'évaluation finale du diplôme.

- Disposer d'outils amonts de modélisation, d'outils de développement de haut niveau s'appuyant sur des frameworks techniques permettant d'avoir un développement essentiellement piloté par les modèles.
- Disposer également d'outils bureautiques ainsi que d'outils professionnels de support au développement logiciel tels que des outils de gestion de configuration logiciel, des outils de planification, des outils de gestion des exigences, etc.

A titre d'exemple, la plate-forme utilisée par une des deux compagnies (nommée Breizh-IS) est décrite ci-après. L'autre plate-forme est bâtie sur la suite des outils IBM et Rational.

### 3.6.1 Architecture matérielle

L'architecture matérielle du futur système comprend :

- Les postes clients standards connectés sur le réseau enseignement de l'UBO. Il s'agit de postes banalisés disposant d'un navigateur Internet. Le système d'exploitation de ces postes clients peut être Windows ou Unix.
- Un serveur d'applications Open Source sous Windows 2000 ou Linux.
- Un serveur de base de données Oracle 10g sous Windows 2000.

### 3.6.2 Architecture logicielle

La solution proposée par Breizh-IS est basée sur une architecture 3-Tier s'appuyant sur l'offre Oracle 10g pour le serveur de données et sur les outils Open Source pour le serveur d'applications.

Il s'agit d'une architecture hybride « Open Source – Propriétaire » dont l'objet est d'essayer de tirer le meilleur parti des uns et des autres avec un objectif de productivité, d'efficacité, de réactivité corrélé à une maîtrise des coûts et si possible du moindre coût.

### 3.6.3 Les outils de développement

#### 3.6.3.1 La suite Oracle 10g

Oracle10g Designer permet de couvrir le cycle de développement complet d'un Système d'Information côté « données » : analyse, conception, génération, déploiement et maintenance d'une base de données Oracle10g. Les autres outils de Oracle 10g Developer Suite ne seront utilisés qu'en cas de besoin.

#### 3.6.3.2 Les outils Open Source

Les outils Open Source utilisés se répartissent en 3 sous-ensembles :

- Les outils d'analyse/conception :
  - Poseidon for UML 1.6 (url : <http://www.gentleware.com/products/poseidonCE.php3>)
- Les outils de développement :
  - ECLIPSE (url : <http://www.eclipse.org/>)
  - Plug-in Tomcat (url : <http://www.sysdeo.com/eclipse/tomcatPluginFR.html>)
  - Plug-in Easy-Struts (url : <http://easystruts.sourceforge.net/>)
  - Plug-in JUnit for Struts (url : <http://strutstestcase.sourceforge.net/>)
- Les frameworks :
  - Tomcat 4.1 (url : <http://jakarta.apache.org/tomcat>)
  - Apache 2.0 (url : <http://www.apache.org>)
  - Hibernate 2.0 (url : <http://www.hibernate.org>)
  - STRUTS (url : <http://jakarta.apache.org/struts>)
  - JUnit (url : <http://www.junit.org>)

## 4 Conclusion et perspectives

Cet article, après avoir exposé brièvement les principes de l'apprentissage par immersion, a mis en regard les artefacts d'une société de services, leur translation dans le paradigme d'immersion et les apports que cette translation apporte au dispositif pédagogique.

Les points abordés sont les suivants :

- Activités (ou compétences) d'un ingénieur du logiciel : de l'intérêt d'avoir une formation basée sur les compétences.
- Une organisation industrielle issue de Thales-IS : l'apprentissage de l'autonomie par la reproduction d'un système opérant.
- Un référentiel d'apprentissage mais aussi de production : de la multiple nature des savoirs (faire mais aussi apprendre).
- Une unité de lieu, de temps et de moyens : la garantie d'une situation authentique d'apprentissage.
- Les outils de management au service de l'apprentissage : la gestion et l'amélioration de la performance appliquée au développement des compétences.
- Une plate-forme d'analyse, de conception, de mise en œuvre, de tests et de déploiement en vraie grandeur : la maîtrise du savoir-faire technologique.

Le dispositif recueille un grand succès auprès des élèves et des employeurs. Cependant, le monde professionnel reste « consommateur » des diplômés par immersion. Pourtant, on pourrait envisager un chemin inverse : de l'apprentissage par immersion vers l'entreprise et de l'enrichissement que cela pourrait apporter au monde professionnel.

## 5 **Bibliographie**

- [BRJ98] Grady Booch, James Rumbaugh, Ivar Jacobson, The Unified Modeling language User Guide, Addison-Wesley Longman, 1998
- [CMMI02] CMMI for Systems Engineering/Software Engineering/Integrated Product and Process Development/Supplier Sourcing, (CMMI-SE/SW/PPD/SS, V1.1), <http://www.sei.cmu.edu/publications/documents/02.reports/02tr011.html>
- [JBR99] Ivar Jacobson, Grady Booch, James Rumbaugh, The Unified Software Development Process, Addison-Wesley Longman, 1999
- [RS03] Vincent Ribaud, Philippe Saliou, Software Engineering Apprenticeship by Immersion, International Workshop on Patterns in Teaching Software Development, ECOOP 2003, University of Darmstadt, Germany, 2003.
- [RS04] Vincent Ribaud, Philippe Saliou, Roles Transformation within a Software Engineering Master by Immersion, International Conference on Interdisciplinary Information Management, IDIMT 2004, Budejovice, Tcheque Republic, 2004.
- [SR05] Philippe Saliou, Vincent Ribaud, La revue de pairs, un support au transfert de compétences inter-étudiants dans un paradigme d'apprentissage par projet, Questions de pédagogie dans le supérieur, Lille, France, 2005.
- [TIS02] TEMPO, la maîtrise du développement de systèmes informatiques, Thales Information System, 2002.
- [Tar99] Jacques Tardif, Le transfert des apprentissages Les éditions logiques, 1999.
- [X50-105] Norme Afnor X50-105, Le management de projet, 1991.