

Curriculum Vitæ Travaux et Titres

Julien DeAntoni

année 2007

1 Résumé	1
2 Informations générales	2
État civil	2
Formation	2
Expérience professionnelle	3
Informations complémentaires	3
3 Activités de recherche	4
Travaux de recherche durant le DEA	4
Travaux de recherche durant la thèse	5
Perspectives scientifiques	7
Travail en cours	8
4 Activités d'enseignements	9
Conclusion des enseignements	10
Autres interventions et encadrements	11
5 Autres activités	12
Réalisation de logiciels	12
Valorisation et transfert scientifique	12
Récompenses	13
Activités d'intérêt général	13
6 Publications et présentations	14
Chapitre d'ouvrage	14
Communications internationales	14
Communications nationales	14
Rapports de recherche	15
Principaux séminaires	15

1 Résumé

Julien DeAntoni

Post-Doctorat INRIA

(Laboratoire IRISA, projet Triskell – Rennes – depuis le 1^{er} Novembre 2007)

Doctorat d'Informatique de l'Institut National des Sciences Appliquées de Lyon

(Laboratoire CITI – Lyon – 12 Octobre 2007)

DEA

(Laboratoire CITI et Université Claude Bernard – Lyon – 2004)

Diplôme d'ingénieur maître

(IUP Micro-Informatique/Micro-Électronique – Le Mans – 2003)

Publications 1 chapitre d'ouvrage, 6 communications internationales, 1 communication nationale (en anglais)

Enseignements moniteur à l'université Claude Bernard

Année	Discipline	Public	TD	TP
2004-2005	Décharge	-	16	
	PCI	L1 sciences		48
	Programmation fonctionnelle	L1 Math-Info		48
	Programmation Impérative (initiation)	L2 Math-Info		24
2005-2006	Système d'exploitation	L3 MASS (formation continue)	24	
	Programmation fonctionnelle	L1 Math-Info	18	24
2006-2007	Programmation Impérative (initiation)	L2 Math-Info		24
	Programmation Impérative Algo et complexité	L3 Info		30
	Surveillance CISCO	L3 Info		9
Total	-	-	58	207

Activités/responsabilités participant au projet ARA REVE
Suivi de la réalisation de la plateforme de démonstration "Flashy"
Comité de programme et de sélection ETR07, session doctorant

2 Informations générales

État civil

Nom : Julien DeAntoni
Date de naissance : 26 août 1980 à Neufchâteau (Vosges), 27 ans
Nationalité : Française
Situation familiale : Célibataire

Adresse personnelle : 2, pré des bonnets rouges
35000 Rennes
France
Téléphone : (+33) 6 09 84 54 23

Adresse professionnelle :
IRISA Projet Triskell
campus de Beaulieu
35 042 Rennes Cedex
FRANCE
Téléphone : (+33) 2 99 84 72 81
Fax : (+33) 2 99 84 71 71
Mél : julien.deantoni@inria.fr

Formation

2004 – 2007 **Doctorat d’Informatique**, soutenance le 12 octobre 2007
Institut National des Sciences appliquées de Lyon (INSA-Lyon) – Axe systèmes embarqués, laboratoire CITI
Sujet : SAIA : un style architectural pour assurer l’indépendance vis-à-vis d’entrées / sorties soumises à des contraintes temporelles

2003 – 2004 **DEA Documents, Images, et Systèmes d’Information Communicants**
Université Claude Bernard (Lyon 1)
Mention Bien, 14.99 / 20
Sujet : Étude des limites et amélioration d’une approche architecturale dirigée par les modèles pour les systèmes embarqués.

2000 – 2003 **Diplôme d’ingénieur maître**
IUP Micro-Informatique/Micro-Électronique du Mans, France
Spécialité : Informatique Industrielle
Mention A.Bien

1998 – 200 **Brevet de Technicien Supérieur en Électronique**
Lycée Henri Loritz (Nancy), académie de Nancy-Metz
Mention Bien, 15.33 / 20

Juin 1998 **Baccalauréat génie électronique**
Lycée Henri Loritz (Nancy), académie de Nancy-Metz
Mention Bien, 15.33/20

Expérience professionnelle

- 11/2007 –** Post-doctorant INRIA SPEEDS au sein du projet Triskell
IRISA Rennes, France
Sujet : Approche architecturale dirigée par les modèles pour le développement d'application temps réel indépendamment de leur plateforme de communication avec l'environnement
- 2004 – 2007** Doctorant au sein du laboratoire CITI de l'INSA de Lyon - Axe systèmes embarqués
Boursier MERT, moniteur de l'enseignement supérieur
Sujet : SAIA : un style architectural pour assurer l'indépendance vis-à-vis d'entrées / sorties soumises à des contraintes temporelles
- 2003 – 2004** Stagiaire dans le département informatique fondamental de l'INSA de Lyon
Projet de recherche : Approche architecturale dirigée par les modèles pour les systèmes embarqués.
- 03/2003 – 08/2003** Stagiaire dans le département informatique fondamental de l'INSA de Lyon
Sujet : Développement, dans le langage SDL, d'un robot explorateur d'après la méthode PROSEUS : "*A development method for PROotyping embedded SystEms by using UML and SDL (PROSEUS)*" Jean-Philippe Babau, Ahmad Alkhodre - ECOOP 2001 - Budapest, Hungary
- 05/2002 – 08/2002** Stagiaire chez Computer Way, SSII spécialisée dans le développement de site web - Le Mans, France
Sujet : Développement du moteur d'un site web pour la diffusion géographique d'information (base de données + script Php)
- 04/2000 – 08/2000** Stagiaire chez Alstom - Nancy, France
Sujet : Développement et mise en place d'un banc de test pour une carte électronique d'interface de puissance de contrôle moteur (assembleur + cartes analogiques d'entrées / sorties)
-

Informations complémentaires

Langues étrangères :

- Anglais : Lu, parlé, écrit
- Espagnol : Lu, parlé
- Italien : Quelques notions

Activités extra-professionnelles :

- Sports : judo (niveau national, arrêt suite à blessure)
- Centres d'intérêt : couture

3 Activités de recherche

Ma première approche au monde de la recherche a eu lieu en mars 2003 au cours d'un stage de maîtrise d'une durée de six mois au Centre d'Innovation en Télécommunication Intégration de service (CITI) de l'Institut National des Sciences Appliquées (INSA) de Lyon. Au cours de ce stage, j'ai pu découvrir le monde de la recherche académique et confirmer mon envie de continuer dans cette voie.

En septembre 2003, j'ai ainsi effectué un DEA d'informatique entre l'université Claude Bernard de Lyon et L'INSA de Lyon. Cette année m'a permis d'avoir une première formation à la recherche, par la recherche. En travaillant avec l'axe systèmes embarqués du laboratoire CITI de l'INSA de Lyon, j'ai proposé une structuration basée sur MDA (*Model Driven Architecture*) pour faciliter la réutilisation de système de contrôle commande.

Du 1^{er} octobre 2004 à la fin du mois d'octobre 2007, j'ai préparé une thèse à Lyon dans l'axe systèmes embarqués du CITI de l'INSA de Lyon. Mes travaux ont abouti à la proposition d'un style architectural pour assurer l'indépendance d'un système vis-à-vis de sa communication avec l'environnement, lorsque cette communication est soumise à des contraintes temporelles. J'ai soutenu ma thèse le 12 octobre 2007.

Depuis le 1^{er} novembre 2007, j'ai démarré un post-doctorat INRIA à Rennes au sein du projet Triskell de l'IRISA. Ce post-doctorat se place dans le cadre du projet européen SPEEDS, où, comme pendant ma thèse, je continue à servir d'interface entre deux domaines que sont les systèmes embarqués temps réel et l'ingénierie dirigée par les modèles.

Travaux de recherche durant le DEA

Laboratoire de recherche : CITI de l'INSA de Lyon
Équipe d'accueil : Axe systèmes embarqués
Responsable : Jean-Philippe Babau
Date de début : 1^{er} Mars 2004
Date de fin : 1 juillet 2004

Titre : Étude des limites et amélioration d'une approche architecturale dirigée par les modèles pour les systèmes embarqués.

Sur la base des travaux réalisés durant un stage précédent, mon stage de DEA a permis d'investiguer les pistes à suivre pour permettre d'utiliser l'ingénierie dirigée par les modèles pour le développement de systèmes temps réels. En particulier il a permis de définir les différentes étapes de modélisation ainsi que les différents modèles associés. De plus, mon stage de DEA s'étant déroulé au même instant que les prémisses de l'approche MDA (*Model Driven Architecture*) préconisée par l'OMG (*Object Management Group*), mes travaux ont donné lieu à un tour d'horizon des différentes approches de modélisation et de transformation de modèles ainsi que des outils associés.

C'est lors de ce stage que le premier prototype d'outil de modélisation a été réalisé grâce à l'outil de

méta-modélisation GME (*Generic Modeling environment*).

Mots-clés systèmes embarqués, Architecture logicielle, ingénierie dirigée par les modèles, MDA

Travaux de recherche durant la thèse

Laboratoire de recherche : CITI de l'INSA de Lyon

Équipe d'accueil : Axe systèmes embarqués

Directeur de thèse : Jean-Philippe Babau, Mcf à l'INSA de Lyon, habilité à diriger les recherches

Financement : MERT + monitorat CIES

Date de début : 1^{er} Octobre 2004

Date de soutenance : 12 Octobre 2007

Jury :

Président	Yvon Trinquet, Professeur à l'Université de Nantes - IRCCyN
Rapporteurs	Jean-Marc Jézéquel - Professeur à l'université de Rennes - IRISA Lionel Seinturier - Professeur à l'université de Lille 1 - LIFL
Examineurs	Sébastien Gérard - Chercheur au CEA-Saclay - LIST Stéphane Ubéda - Professeur à l'INSA-Lyon - CITI François Vernadat - Professeur à l'INSA-Toulouse - LAAS

Titre : SAIA : un style architectural pour assurer l'indépendance vis-à-vis d'entrées / sorties soumises à des contraintes temporelles

Du fait de leur complexité croissante, le développement des systèmes embarqués et temps réel nécessitent l'application de principes de génie logiciel. Cependant, les contraintes temporelles et la criticité de ces systèmes nécessitent également l'application de techniques formelles.

Dans ce cadre, le travail développé pendant ma thèse propose un style architectural pour le développement d'une architecture logicielle dédiée aux systèmes de contrôle de processus. Les principaux objectifs de l'approche proposée sont de permettre le développement, via des modèles, d'une application de contrôle indépendamment d'une plateforme de communication et de s'assurer ensuite de la correction du déploiement de cette application de contrôle sur une plateforme de communication spécifique. Par plateforme de communication on considère les capteurs, les actionneurs et les pilotes associés.

L'étude des architectures logicielles a permis d'identifier certains principes clés devant être mis en œuvre. Ces études ont mis en avant la nécessité de définir des styles architecturaux permettant d'une part d'appliquer certains principes de génie logiciel et d'autre part de restreindre les configurations réalisées afin qu'elles restent analysables. La première contribution de mes travaux consistent donc à proposer une architecture spécifique permettant d'obtenir la qualité logicielle recherchée : c'est-à-dire l'indépendance vis-à-vis de la plateforme.

On remarque également aux vues de l'études des architectures logicielles que la description de la QoS (*Quality of Service*) requise et fournie par un composant permet d'utiliser celui-ci en respect de ses exigences extra-fonctionnelles. Ainsi, lors de la liaison entre deux composants, un contrat de QoS est établi si les deux composants peuvent être connectés tout en respectant les contraintes de QoS. Sur ce point, ma thèse a étendu les travaux existants où les contrats de QoS traités se focalisent sur les propriétés simples et décrites à l'aide de valeurs discrètes, contraignant ainsi l'expressivité. De plus, il est classique lors de la connexion entre deux interfaces, d'utiliser un connecteur possédant un comportement (appelé "glue") pour pallier l'hétérogénéité entre les interfaces. Du constat de cette étude nous notons un point

important, il est nécessaire de quantifier l'impact du connecteur sur la QoS lors de l'établissement du contrat, ce point n'est pas traité dans la littérature dès lors que la "glue" n'est pas un choix réalisé parmi une liste finie de comportements.

3.0.1 La structure adoptée pour SAIA

Puisque l'objectif est de permettre le développement d'une application de contrôle indépendamment d'une plateforme de communication spécifique, une étude a été réalisée sur les différents styles architecturaux permettant de s'abstraire d'une plateforme, au sens large du terme. Ceux-ci mettent clairement en avant l'utilisation de différentes couches logicielles reflétant différents niveaux d'abstraction vis-à-vis de la plateforme. On note aussi qu'il est possible de décrire une plateforme abstraite afin de spécifier les besoins d'une application et ainsi définir un ensemble de plateformes potentiellement utilisables. Ces structurations sont rarement utilisées dans le domaine des systèmes visés car la correction de l'application est fortement liée aux performances de la plateforme.

Le style architectural proposé dans cette thèse propose d'utiliser une structuration en trois couches distinguant :

1. une application de contrôle et sa plateforme abstraite ;
2. une plateforme réelle ;
3. le connecteur complexe permettant de lier la plateforme abstraite et la plateforme réelle.

D'une part, l'utilisation de l'ingénierie des modèles nous permet d'exprimer les différents concepts relatifs à chacune des couches et d'exprimer également les contraintes d'associations entre ces différentes couches. D'autre part, il est possible, par transformation de modèle, de pallier les inconvénients de performance des structurations en couches en destructurant le système lors de son implémentation.

3.0.2 La gestion de la QoS dans SAIA

Puisque la correction d'une application de contrôle de processus est dépendante des performances de la plateforme, il faut restreindre l'ensemble des plateformes réelles utilisables par une description de la QoS sur la plateforme abstraite. On définit ainsi un ensemble de plateformes réelles qui, sous réserve de satisfaire la QoS et les services décrits dans la plateforme abstraite, peuvent être utilisées par l'application de contrôle concernée. On obtient ainsi une application non dépendante d'une plateforme réelle spécifique, cependant on est tout de même dépendant d'un ensemble de plateformes réelles, pas forcément connu a priori. Plus cet ensemble est grand, plus le niveau d'indépendance est élevé.

Afin d'obtenir un niveau d'indépendance élevé, il est important que la plateforme abstraite décrive les besoins de l'application de manière indépendante d'une technologie existante. Il est également important que la QoS ne restreigne pas drastiquement cet ensemble. Elle doit donc être exprimée d'une manière riche. Plutôt que de se baser sur des valeurs constantes ou des intervalles, nous décrivons un langage de la QoS décrit à l'aide d' ω -expression régulière. Enfin, dans le but de vérifier qu'une plateforme réelle appartient à l'ensemble des plateformes utilisables, des contrats de QoS sont établis entre chaque composant de la plateforme abstraite et de la plateforme réelle. Les contrats sont établis si la QoS fournie satisfait la QoS fournie. Les travaux menés durant ma thèse ont alors défini formellement chacun des concepts relatifs à la QoS ainsi que la relation de satisfaction permettant l'établissement d'un contrat de QoS.

Évaluation de l'approche

L'évaluation de SAIA a été réalisée au travers de divers aspects : l'outillage, les méthodes et plusieurs mises en œuvre. En particulier un outil pour la modélisation de systèmes conformes au style architectural défini pendant ma thèse a été développé. D'une part cet outil permet d'assurer que la description d'un système respecte les types de composants et les contraintes structurelles définies dans le style architectural.

D'autre part, il permet, par transformation de modèles d'utiliser les outils d'analyse formelle permettant de vérifier que les différents contrats de QoS peuvent être établis.

Plusieurs mises en œuvre ont également été réalisées. En particulier, j'ai participé à deux concours internationaux de robotique durant lesquels mon approche a été utilisée. Ceci a permis de montrer la faisabilité de l'approche, mais également de montrer, en ayant des performances tout à fait respectables (2^{ème} au premier concours et 4^{ème} au deuxième), que l'utilisation de modèle n'est pas incompatible avec la notion de performance.

Ainsi, il a été possible d'extraire plusieurs bonnes propriétés de l'approche proposée :

1. Au niveau fonctionnel, il a été possible de réutiliser un application de contrôle sur plusieurs plateformes réelles grâce à la spécification de la plateforme abstraite. De plus, lors d'une modification dans la plateforme réelle ou dans la plateforme abstraite, les changements interviennent localement au sein du ou des sous connecteur(s) concerné(s). Le fait de concentrer les modifications à réaliser dans un composant est une aide précieuse pour le développeur. La séparation du connecteur complexe en sous connecteurs permet également au développeur de se concentrer sur un composant spécifique du système. Puisque la plateforme abstraite agit comme une interface entre la plateforme réelle et l'application, il est également possible de réaliser des changements dans l'application de contrôle sans pour autant que ceux-ci soient propagés dans tout le système. Il est également possible de développer l'application de contrôle et la connexion à une plateforme réelle en parallèle.
2. Au niveau extra-fonctionnel, l'utilisation d'un langage de QoS permet d'être plus expressif, cependant, il est difficile de savoir intuitivement si un contrat peut être établi ou non. SAIA offre donc une chaîne d'outil permettant d'automatiser l'établissement de contrat de QoS. De plus, les contrats sont établis sous connecteur par sous connecteur. Ainsi, il est possible de se focaliser sur le paramétrage d'un sous connecteur jusque l'établissement d'un contrat de QoS. Ce paramétrage est facilité par la possibilité de réaliser un ensemble d'analyses de manière automatique.

Perspectives scientifiques

Les travaux menés durant ma thèse sont à la croisée de plusieurs domaines. Aussi, ils m'ont ouverts beaucoup de perspectives de recherche. Dans cette section, je vais aborder quatre des points qu'il me paraît intéressant d'investiguer.

- L'application d'un style architectural permet d'atteindre certains objectifs de qualité logiciel tels que la réutilisation, la facilité de maintenance, etc. Il me paraît intéressant d'étendre les travaux existants de deux points de vue. Premièrement les styles architecturaux existants sont relativement informels et reste très généraux de sorte qu'il est souvent difficile de savoir quand les appliquer de même qu'il est souvent difficile d'être certain de les appliquer correctement. L'ingénierie des modèles permet quant à elle de simplifier l'application de patrons de conception et permet également l'application et la vérification de contraintes structurelles et comportementales. Ainsi, il semble intéressant d'investiguer dans quelle mesure il est possible de réaliser, grâce à l'ingénierie dirigée par les modèles, un outil permettant d'une part la définition d'un style architectural et d'autre part de permettre la modélisation de systèmes conformes à ce style.
- Dans SAIA, le style architectural développé pendant ma thèse, la spécification de la QoS dans la plateforme abstraite est réalisée pour chaque *entrée* et pour chaque *sortie*. Cependant, cette description est statique quelque soit le mode de fonctionnement (normal, dégradé, etc.) de l'application de contrôle. Certaines *entrées* et/ou *sorties* devraient pouvoir spécifier une qualité de service différente en fonction du cycle de vie de l'application de contrôle. Il serait alors nécessaire de décrire le cycle de vie de l'application de contrôle ainsi que les liens qui lient ce cycle de vie à la QoS de la plateforme abstraite. Enfin, il faudrait investiguer d'une part l'impact du cycle de vie sur le connecteur complexe et sur l'établissement des contrats de QoS et d'autre part spécifier formellement comment s'effectue le passage d'un mode à un autre.
- Il existe plusieurs niveaux de description d'une architecture : logique, technique et opérationnelle. SAIA permet de décrire une architecture logique, ce qui permet de réaliser des analyses avant de

traiter les détails de l'implémentation. La plateforme d'exécution et l'effet de l'ordonnancement sont abstraits par des budgets temporels. Afin de réaliser l'implémentation exécutable d'un modèle spécifiant une architecture logique, il est nécessaire de préciser le langage de programmation, l'utilisation ou non d'un intergiciel ou encore le modèle de tâches. Il serait alors intéressant d'investiguer le raffinement de ce modèle logique afin de produire une architecture opérationnelle. Dans la même idée que ACCORD¹, une solution serait de caractériser une plateforme réelle d'exécution multitâche (du type d'un système d'exploitation temps réel) et une machine virtuelle d'exécution multitâche. Alors, dans la même idée que SAIA, la machine virtuelle serait vue comme une plateforme d'exécution abstraite. Ensuite le travail consisterait à : formaliser les caractéristiques de QoS pertinentes, formaliser la description de la QoS requise et fournie et formaliser la relation de satisfaction pour chaque caractéristique de QoS. Ainsi, on devrait pouvoir caractériser la connexion entre la plateforme réelle et la plateforme abstraite et, si nécessaire, expliciter les contraintes et méthodes permettant d'envisager l'établissement d'un contrat de QoS.

- Dans SAIA, les interprétations nécessaires entre les informations de la plateforme abstraite et celles de la plateforme réelle sont réalisées statiquement pendant la phase de modélisation. Dans les systèmes où les contraintes temporelles sont moindres et les systèmes ouverts (PDA, téléphones, ...) l'arrivée et le déploiement d'une application devrait se faire de manière transparente sur plusieurs plateformes. L'utilisation d'une plateforme abstraite semble être une bonne solution pour la spécification des besoins sans pour autant spécifier une technologie particulière. Dans le but de connecter la plateforme abstraite à la plateforme réelle au moment du déploiement, il serait alors nécessaire de définir une base ontologique regroupant la sémantique de la plateforme abstraite et de la plateforme réelle ainsi que des interprétations préétablies permettant de connecter les deux plateformes. Il serait également nécessaire d'utiliser un intergiciel spécifique permettant de gérer, à l'exécution, la description ontologique des plateformes ainsi que le choix de l'interprétation pour l'établissement de la connexion entre les plateformes.

Travail en cours

Depuis le 1^{er} Novembre 2007, j'ai démarré un post-doctorat à Rennes au sein du projet Triskell de l'IRISA sous la direction du Pr Jean-Marc Jézéquel. Des travaux émergent par rapport aux problématiques du projet Triskell et à mon projet de recherche qui s'inscrit dans la continuité de ma thèse.

Ces travaux visent à travailler dans le cadre du projet SPEEDS (Speculative and Exploratory Design in Systems Engineering). Le projet SPEEDS est un projet européen FP6 destiné à permettre la description et la validation formelle d'une architecture à composant pour les domaines embarqués critiques. Il regroupe de nombreux partenaires académiques (INRIA France - Verimag France - OFFIS Allemagne - PARADES Italie), industriels (Airbus - Bosch - Israël Aerospace Industry - SAAB - ...) et divers vendeurs d'outils (Telelogic - Esterel Technology - TNI-software - ...). Ces composants peuvent idéalement être logiciel / matériel à temps discret ou continu. Les travaux réalisés durant ma thèse vont me permettre d'intervenir sur le niveau de description de l'architecture et en particulier sur le passage entre la description d'une architecture logique et la description d'une architecture opérationnelle. De plus, mes compétences en modélisation, en systèmes embarqués et en électronique vont me permettre d'intervenir auprès des nombreux partenaires d'horizon divers.

De plus, SPEEDS se place comme un point central entre plusieurs outils de modélisation et/ou de validation. Pendant ma thèse, la manipulation conjointe d'outils de modélisation, de méthodes formelles et d'outils de validation, me permet de d'aborder plus sereinement les enjeux liés à cette problématique.

¹Agnès Lanusse and Sébastien Gérard and François Terrier – Real-Time Modeling with UML : The ACCORD Approach, Lecture Notes in Computer Science - 1999

4 Activités d'enseignements

En tant que moniteur à l'université Lyon 1, j'ai effectué 196 heures d'enseignements (équivalent TD) sur trois années. Parmi ces 196 heures, j'ai effectué 58 heures de TD et 207 heures de TP. Pendant cette période, j'ai essayé de varier mes enseignements afin de pouvoir m'adapter plus facilement à un large spectre d'enseignement. Le tableau suivant fournit un résumé des heures d'enseignements effectuées (Ce tableau récapitulatif ne tient pas compte des heures de service non rémunérées pour l'encadrement de stages).

Année	Discipline	Public	TD	TP
2004-2005	Décharge	-	16	
	PCI	L1 sciences		48
	Programmation fonctionnelle	L1 Math-Info		48
	Programmation Impérative (initiation)	L2 Math-Info		24
2005-2006	Système d'exploitation	L3 MASS (formation continue)	24	
	Programmation fonctionnelle	L1 Math-Info	18	24
2006-2007	Programmation Impérative (initiation)	L2 Math-Info		24
	Programmation Impérative Algo et complexité	L3 Info		30
	Surveillance CISCO	L3 Info		9
Total	-	-	58	207

Les paragraphes suivant détaillent la liste des unités d'enseignements dans lesquelles je suis intervenu.

PCI (Permis de Conduire Informatique) : année 04-05

Étudiants : 1^{er} semestre de Licence scientifique

Objectifs : Introduction à l'informatique en tant qu'outil

Contenu : plusieurs axes sont abordés tels que le maniement d'un système d'exploitation (Windows), la bureautique ou encore l'Internet et les courriels. Ce fut ma première intervention en tant que chargé de TP. Elle m'a permis de m'habituer à la gestion et l'animation d'un groupe de manière indépendante des difficultés techniques.

Volume horaire : 48 heures de TP

Algorithmique et programmation fonctionnelle et récursive : années 04-05 et 05-06

Étudiants : 2^{ème} semestre de Licence à l'IUP Math-Info

Objectifs : Apprentissage de l'algorithmique par la programmation fonctionnelle et récursive

Contenu : La formation s'appuie sur le langage scheme (<http://www.schemers.org/>). La première année, cette formation m'a permis de mettre en œuvre l'expérience acquise en PCI dans un contexte où l'informatique n'est plus enseigné comme un outil mais comme une discipline à part entière. J'ai pu suivre la deuxième année un même groupe en TD puis en TP. Ceci permet de mieux comprendre où interviennent les difficultés et d'en tirer les enseignements pédagogiques associés. Dans cette unité d'enseignement, j'ai été amené à rédiger des sujets de TPs notés.

Volume horaire : 72 heures de TP et 18 heures de TD

Algorithmique et programmation impérative initiation : années 04-05 et 06-07

Étudiants : 3^{ème} semestre de Licence à l'IUP Math-Info

Objectifs : Introduction à la programmation impérative par la résolution de problèmes simples et la présentation d'un langage algorithmique

Contenu : La formation se base sur le langage C. Lors de la première année, cette formation m'a amené à proposer un ensemble de projets sur plusieurs séances et de suivre leur réalisation.

Volume horaire : 48 heures de TP

Algorithmique Programmation et Complexité : années 06-07

Étudiants : 6^{ème} semestre de Licence à l'IUP Informatique

Objectifs : Introduction des concepts de Type de Données Abstraites (TDA), de complexité des algorithmes ainsi que de certaines méthodes de conceptions d'algorithmes

Contenu : Les TPs étaient articulés autour de projet sur plusieurs séances. En plus, de la correction des projets, je me suis investi sur la préparation des enseignements.

Volume horaire : 30 heures de TP

Système d'exploitation et programmation parallèle : année 05-06

Étudiants : 5^{ème} semestre de Licence MASS en formation continue

Objectifs : Bases de la programmation dans un environnement linux (script shell) et introduction à la programmation multiprocessus

Contenu : Cette formation est en adéquation avec ma thèse où je manipule ces notions quotidiennement. J'ai proposé quelques modifications dans cette unité d'enseignement dans le but d'aider les étudiants (saliés) à se familiariser rapidement avec les notions nouvelles auxquelles je suis habitué.

Volume horaire : 24 heures de TD

Conclusion des enseignements

Mon statut de moniteur m'a permis de faire mes premières armes en tant qu'enseignant. Il m'a motivé à continuer à enseigner dans l'enseignement supérieur.

Je me suis fortement investi dans les formations « Algorithmique Programmation et Complexité » et « Algorithmique et programmation fonctionnelle et récursive ». Dans ces formations, j'ai proposé / modifié différents sujets de TD/TP en fournissant aux autres intervenants une mise en œuvre de référence et une correction détaillée. D'une manière générale, j'ai systématiquement pris part à l'évaluation au sein de mes enseignements en contribuant à l'élaboration des sujets d'examens et en corrigeant les différents projets des étudiants.

De plus, en essayant d'enseigner dans des formations ne faisant pas directement intervenir les compétences acquises en recherche, j'ai essayé d'ouvrir un maximum le spectre d'enseignements dans lesquels je pourrai être un éventuel intervenant futur tout en restant disponible et ouvert aux enseignements que l'on me proposait. À l'issue de ces trois ans, je désire continuer sur cette voie en postulant à la qualification au titre de maître de conférence.

Autres interventions et encadrements

En 2005, j'ai encadré un stage d'IUT effectué par Sylvain Roche dont le sujet portait sur l'implémentation et l'instrumentation temporelle d'un pilote logiciel sous linux. Mon travail a, dans un premier temps été la rédaction d'un sujet de stage. J'ai ensuite été amené à suivre l'étudiant et à répondre à ces diverses interrogations. Enfin, le moment venu je l'ai guidé dans la rédaction de son mémoire et dans la réalisation de sa présentation. Finalement j'ai assisté à sa soutenance devant ces enseignants.

En 2006, j'ai encadré un second stage d'IUT, effectué par Aurélien Courant, le sujet proposait la mise en place d'une architecture temps réel et multiprocessus ainsi que son instrumentation temporelle. J'ai du faire preuve d'un degré d'abstraction adapté à l'étudiant pour lui permettre une compréhension aisée du sujet. Comme pour l'année précédente, outre la rédaction du sujet, j'ai accompagné l'étudiant tout au long de son stage jusqu'à sa soutenance.

Enfin, en 2007, j'ai suivi le projet technique de deux étudiants : Adrien Bailly, en deuxième année d'école d'ingénieur CPE et Samarth Agrawal, en dernière année d'école informatique en Inde. Le projet consistait à réaliser une maquette, nommée "Flashy" servant de plateforme de démonstration pour l'équipe. Cette maquette commande un moteur en position à l'aide d'une commande de type PID afin de suivre le déplacement d'un chenillard de diodes électroluminescentes piloté par le port parallèle d'un ordinateur.

5 Autres activités

Réalisation de logiciels

Réalisation de l'interface graphique de pilotage de la plateforme de démonstration "Flashy", réalisée pour l'axe systèmes embarqués du laboratoire CITI. Cette interface a été réalisée à l'aide du langage Qt.

Un des résultats des travaux de ma thèse est le développement d'un environnement de description d'architecture logicielle conforme au style architectural SAIA. Cet outil propose trois fonctionnalités principales :

- la construction d'une architecture permettant exclusivement l'utilisation de composants décrits par le style (contraintes axiomatiques),
- la vérification de la cohérence de la configuration résultante vis-à-vis du style architectural (contraintes de types et structurelles),
- la transformation de la configuration obtenue vers des automates temporisés permettant la réalisation des analyses décrites dans ma thèse.

L'éditeur graphique a été développé grâce à l'outil GME, proposé par l'université de Vanderbilt. Cette outil permet de spécifier un éditeur graphique d'après la description des méta-modèles d'un langage (dans notre cas du style architectural), La transformation vers le langage IF a été réalisé dans le langage java par l'analyse d'un fichier xml conforme à la description de la configuration.

Valorisation et transfert scientifique

Pendant ma thèse, j'ai participé aux différentes réunions du projets ARA REVE (safe Reuse of Embedded components in heterogeneous enVironmEnts)². REVE est un projet financé par l'ANR dans le cadre des actions de recherche amont (ARA) du programme SSIA. Le projet REVE a été labellisé en 2005 pour une durée de 3 ans (2006-08).

Le but du projet REVE est de construire un modèle de composant, un support d'exécution et un système de typage pour les applications embarquées du domaine du temps-réel dans lequel des politiques de prise en compte des changements de contexte peuvent être spécifiées, programmées et vérifiées. Plus précisément, le projet cible deux types de changements de contexte :

- les changements de mode d'exécution,
- les changements de mode de communication.

Au sein du CITI, ce projet s'inscrit dans la poursuite des travaux menés autour de Qinna pour les modes d'exécution et de SAIA pour les modes de communication.

Au niveau des réunions, j'ai participé régulièrement aux rencontres du groupe (4 par an) et j'ai animé un séminaire sur SAIA. Au niveau des livrables, j'ai participé à l'état de l'art sur l'introduction des modes d'exécution et sur la prise en compte du contexte dans un système.

Je suis également intervenu lors des discussions sur la mise en place d'un modèle à composant adapté aux contraintes de l'embarqué.

²<http://reve.futurs.inria.fr/> - <http://www.ara-reve.org/>

Récompenses

Deuxième prix du concours international de robotique maRTian Task organisé en conjonction avec RTSS 2005 à Miami. <http://www.rtss.org/rtss2005/martiantaskwinners.htm>

Quatrième prix du concours international de robotique CiberMouse organisé en conjonction avec RTSS 2006 à Rio de Janeiro. http://www.ieeta.pt/~lau/web_ciberRTSS/results.htm

Activités d'intérêt général

Pendant mes années de thèse, j'ai de plus pris part à différentes activités d'intérêt général au niveau de l'équipe de recherche et de l'UFR d'enseignement.

Activités annexes au niveau de l'UFR

- Durant mes trois années de thèse, j'ai participé de manière bénévole à des séances de soutien aux étudiants. En particulier j'ai organisé des demi journées Questions/Réponses où j'étais de permanence pour les étudiants. Ces journées intervenaient soit pendant la réalisation d'un projet sur plusieurs séances afin de servir de conseiller, soit avant les examens pour dissiper les doutes des étudiants lors de leur révision finale.
- Devant le manque de littérature sur la manière de déboguer (ou déverminer) un programme, j'ai réalisé un guide pratique retraçant les différentes étapes et technique de débogage d'un programme informatique. Si ce guide se veut un maximum général, il est illustré au travers du langage C et du logiciel Microsoft Visual, utilisé en première année à l'université Claude Bernard. Il a été distribué aux étudiants de première année informatique.

Activités annexes au niveau de l'équipe

- En charge de l'accueil des stagiaires dans les locaux (préparation des machines, des bureaux et des comptes informatiques).
- L'équipe ne possédant pas de système de partage de calendrier géré par l'établissement, j'ai proposé la mise en place de partage de calendrier. J'ai pour cela instauré l'utilisation des agendas gérés par google.
- J'ai participé à la réhabilitation et la réorganisation des locaux externes aux locaux principaux du laboratoire

Participation à l'organisation de conférences

- Comité de programme de la session développement d'applications temps réel de l'école d'été temps réel 2007, session doctorants³ : sélection et relecture des articles + animation de la session "développement d'applications temps réel"

³<http://www.etr2007.org/>

6 Publications et présentations

Chapitre d'ouvrage

- [1] Jean-Philippe Babau, Julien DeAntoni. Architectures logicielles pour les systèmes embarqués temps réel. In “*École d’été temps réel*”, pages 75–94, 2007.

Communications internationales

(Avec édition d’actes et comité de sélection)

- [2] Julien DeAntoni, Jean-Philippe Babau. “*A MDA approach for systems dedicated to process control*”. Eleventh IEEE International Embedded and Real Time Computing Systems and Applications (RTCSA’05), 2005.
- [3] Julien DeAntoni, Jean-Philippe Babau. “*A MDA-based approach for real time embedded systems simulation*”. Nineth IEEE International Symposium on Distributed Simulation and Real Time Applications (DS-RT’05), 2005.
- [4] Julien DeAntoni, Jean-Philippe Babau. “*SAIA : Sensors/Actuators Independent Architecture - a showcase through martian task specification*”. Proceedings of the ERTSI 2005 - Embedded Real Time Systems Implementation Workshop, held in conjunction with 26th IEEE International Real Time Systems Symposium (RTSS’05), pages 43–50, 2005. <http://www.cs.york.ac.uk/ftpdir/reports/YCS-2005-397.pdf>.
- [5] Julien DeAntoni, Jean-Philippe Babau. “*Cibermouse design : A case study for SAIA model reuse*”. Proceedings of the Cibermouse Team reports, a satellite event of the 27th IEEE International Real Time Systems Symposium (RTSS’06), pages 9–12, 2006.
- [6] Julien DeAntoni, Jean-Philippe Babau. “*SAIA : safe deployment of sensors based real time application*” ; Workshop on Models and Analysis for Automotive Systems held in conjunction with the 27th IEEE International Real-Time Systems Symposium (RTSS’06) ; pages 45–48. 2006

Communications nationales

(Avec édition d’actes et comité de sélection)

- [7] Julien DeAntoni and Jean-Philippe Babau. “*Model driven engineering method for SAIA architecture design*”. Ingénierie Dirigée par les modèles (IDM’06), pages 109–122, 2006.

Rapports de recherche

- [8] Julien DeAntoni, Jumel Fabrice, Jean-Philippe Babau. “*Tech-report20070130 : A simple simulink robot simulator*”. Technical report, INSA-Lyon, 2007.

Principaux séminaires

Outre les exposés réalisés pour chacun des papiers publiés, j’ai réalisé plusieurs séminaires scientifiques parmi lesquels :

- SAIA : Sensor / Actuator Independent Architecture. Séminaire dans le cadre du séminaire d’équipe Triskell, 18 Novembre 2007
 - SAIA : Sensor / Actuator Independent Architecture. Séminaire dans le cadre des journées composants du cluster EmSoC recherche, 1 juin 2007, Grenoble.
 - SAIA : Sensor / Actuator Independent Architecture. Séminaire dans le cadre du projet REVE : safe Reuse of Embedded components in heterogeneous environments. 25 Avril 2007, Lyon
 - SAIA : a Sensor / Actuator Independent Architecture. Présentation poster à MDE 2006 (Model Driven Embedded), septembre 2006, Aber Wrac’h
 - SAIA : a Sensor / Actuator Independent Architecture. Présentation poster aux journées EmSoC recherche, 8-9 juin 2006, Villars de Lens
 - Une approche MDA pour les systèmes temps réel. Séminaire thésard au laboratoire CITI. avril 2005, Lyon
 - Démonstration de la modélisation et implémentation réalisées dans le cadre du concours Josefil. septembre 2004, Aber Wrac’h
-