

Curriculum Vitæ

Bechir ZALILA — Ingénieur — Docteur

État civil	: Célibataire	D. de Naiss.	: 26 Mai 1981
Adresse	: BP N 70, bureau de poste de Chihia 3041, Sfax, Tunisie	Nationalité	: Tunisienne
Tél	: (+216) 74 848 210	Mail	: bechir.zalila@enis.rnu.tn
		Site web	: http://www.redcad.org/members/zalila

Fonctions

2008-présent **Assistant Universitaire.** SFAX/TUNISIE
Affiliation : enseignant-chercheur au **DGIMA** (Département de Génie Informatique et de Mathématiques Appliquées)
Lieu : **ENIS** (École Nationale d'Ingénieurs de Sfax)
▷ 1^{re} année : responsable du module "Programmation structurée".
▷ 2^e année : responsable du module "Génie logiciel distribué".
▷ 3^e année : responsable de l'axe (spécialité) "Ingénierie des Systèmes Distribués" (ISD) et des modules "Plates-formes de développement d'applications distribuées" et "Maintenance des logiciels".

Formation

2005-2008 1^{re}, 2^e et 3^e années de doctorat en informatique répartie temps-réel et em- PARIS/FRANCE
(3 ans) barquée.
Sujet de thèse : CONFIGURATION ET DÉPLOIEMENT D'APPLICATIONS TEMPS-RÉEL RÉPARTIES EMBARQUÉES À L'AIDE D'UN LANGAGE DE DESCRIPTION D'ARCHITECTURE. Thèse soutenue le 07 novembre 2008 et obtenue avec mention *Très Honorable*
Lieu : **ENST** (École Nationale Supérieure des Télécommunications)
▷ **Composition du jury.**
▷ *Président* : Jacques MALENFANT, Professeur, LiP6 - Université Pierre & Marie Curie.
▷ *Rapporteurs* : Yvon KERMARREC, Professeur, TELECOM Bretagne et Lionel SEINTURIER, Professeur, Université de Lille 1.
▷ *Examineurs* : Peter FEILER, Directeur de recherche, SEI - Carnegie Mellon University et Franco GASPERONI, Directeur général, Europe - AdaCore.
▷ *Directeurs de thèse* : Laurent PAUTET, Professeur, ENST et Jérôme HUGUES, Maître de conférences, ENST.

2004-2005 MASTER M2 Informatique. PARIS/FRANCE
(1 an) **Spécialité** SAR (Systèmes et Applications Répartis). **Parcours** SRETR (Systèmes Répartis Embarqués ou Temps-réel)
Lieu : **UPMC** (Université Pierre & Marie Curie (Paris VI))

2002-2005 Études d'ingénieur. PARIS/FRANCE
(3 ans) **Domaines d'étude** : architecture des systèmes, électronique et robotique
Lieu : **ENST**

2000-2002 Enseignement préparatoire aux concours d'entrée aux grandes écoles d'ingénieurs. LA MARSIA/TUNISIE
(2 ans) **Spécialité** : MP*
Lieu : **IPEST** (Institut Préparatoire aux Études Scientifiques et Techniques)

1993-2000 Enseignement secondaire. SFAX/TUNISIE
(7 ans) Baccalauréat-mathématique obtenu avec mention *Très Bien*
Lieu : Lycée Secondaire 9 Avril 1938

Stages

04/2005- Stage de fin d'études. PARIS/FRANCE
09/2005 **Sujet du stage** : *Optimisation, Déterminisme et Asynchronisme de Souches*
(6 mois) *et Squelettes CORBA pour Systèmes Répartis Temps-réel*
Lieu : ENST
▷ *Programmation temps-réel répartie en Ada.*
▷ *Théorie de la compilation.*

Activités de Recherche

Les standards de répartition (CORBA, DSA...) sont apparus pour rendre le développement des applications réparties plus facile : le problème de la complexité de développement des applications réparties est résolu par ces standards. Avec l'apparition de nombreux standards, une nouvelle forme de complexité et de difficulté est apparue : optimiser les composants fournis par l'intergiciel en fonction des propriétés de l'application. De plus, les applications réparties conçues selon ces standards sont généralement très mal-adaptées pour les systèmes temps-réel répartis embarqués. En effet, l'empreinte mémoire de telles applications est généralement grande et l'analyse d'ordonnancement est souvent très difficile même en utilisant des standards comme RT-CORBA.

L'objectif de ma thèse est de fournir des propositions exploitant les fonctionnalités fournies par les langages de description d'architecture pour produire pour chaque application répartie, un intergiciel qui lui soit dédié minimisant ainsi son empreinte mémoire et optimisant sa performance. Plus généralement, il s'agit de proposer et d'implémenter un processus automatique de développement qui, à partir d'une description architecturale d'une application répartie, effectue différentes analyses pour vérifier le bon fonctionnement de cette application (sémantique, ordonnancement...). Ensuite, une large partie des composants applicatifs ainsi que les composants intergiciels sont générés et configurés automatiquement. Finalement, les composants qui sont générés automatiquement et les composants fournis par l'utilisateur doivent être intégrés pour construire l'application proprement dite et la déployer.

On s'intéressera dans un premier temps au langage de description d'architecture AADL (Architecture Analysis and Design Language). Il faut identifier les composants de l'intergiciel qui seront produits automatiquement. Le reste des composants constitueront le cœur d'un intergiciel minimal qui sera commun à toutes les applications. Cette approche permettra d'obtenir un intergiciel réduit aux besoins spécifiques d'une application répartie donnée ce qui correspond aux demandes du projet Européen ASSERT (Projet animé par l'Agence Spatiale Européenne et qui compte plus de 30 partenaires dont l'ENST).

Ce travail demande une recherche dans le domaine des architectures intergicielles importantes. Plusieurs instances d'un intergiciel minimal (une pour chaque langage de programmation supporté) seront conçues ou prises sur des logiciels existants afin de mener à bien une phase de prototypage.

Dans un second temps, il faudra adopter une approche pour assembler automatiquement les composants applicatifs et intergiciels. Un premier axe portera naturellement sur les aspects fonctionnels. Un autre aspect plus délicat portera non pas sur la partie architecturale mais sur la partie comportementale de la modélisation des composants de l'intergiciel. En effet, il conviendra non seulement de pouvoir générer automatiquement l'assemblage des composants mais les parties comportementales de ces composants. Pour ce faire, une étude sera menée et prendra en compte les travaux effectués dans le cadre de PolyORB et dans le contexte du projet IST ASSERT afin de traiter des problématiques liées au temps réel.

Ocarina : Ocarina¹ est un outil écrit en Ada pour manipuler des modèles AADL. Ocarina permet d'effectuer l'analyse sémantique des modèles AADL, l'analyse d'ordonnancement (à travers l'outil Cheddar), et la vérification formelle (en générant des réseaux de Petri à partir des modèles AADL et en les analysant avec l'outil CPN-AMI). Ocarina permet de générer des applications distribuées à partir de leur modèles AADL.

Ma contribution dans Ocarina consiste à développer les générateurs de vers les intergiciels PolyORB et PolyORB-HI et à assurer l'intégration correcte des composants.

1. <http://aadl.enst.fr/ocarina>

PolyORB : PolyORB est un "intergiciel schizophrène". Je contribue à PolyORB notamment en développant son nouveau compilateur IDL : IAC. PolyORB est un logiciel libre disponible au téléchargement sur le site de Ada-Core²

PolyORB-HI : PolyORB-HI³ est un intergiciel dédié aux systèmes critiques (aviation, espace, etc). Une grande partie du code de cet intergiciel est générée automatiquement avec l'outil Ocarina.

Encadrement de Thèses

2009-présent

- ▷ Participation à l'encadrement de FATMA KRICHEN qui effectue une thèse intitulée "*Technique et mécanisme de support d'exécution pour la reconfiguration dynamique des architectures logicielles à composants pour les systèmes embarqués*" en cotutelle avec l'Université de Toulouse-2 Le Mirail.

Encadrement de Mastères de Recherche

2006-2007

- ▷ Participation à l'encadrement de JULIEN DELANGE qui a effectué un stage intitulé "*Génération de Code C pour Applications Critiques*" dans le cadre de son MASTER Recherche SAR à l'ENST, Paris. Mastère soutenu en septembre 2007 et obtenu avec mention *Bien*

2007-2008

- ▷ Participation à l'encadrement de GILLES LASNIER qui a effectué un stage intitulé "*Étude et Support du Standard AADLv2 dans Ocarina*" dans le cadre de son MASTER Recherche SAR à TELECOM ParisTech. Mastère soutenu en septembre 2008 et obtenu avec mention *Bien*

2009-2010

- ▷ Participation à l'encadrement de SIHEM LOUKIL qui a effectué un projet intitulé "*Extension d'un Langage de Description d'Architecture pour la Programmation Orientée Aspect*" dans le cadre de son MASTER Recherche NTSID à l'ENIS.

Expérience d'enseignement

2005-2006

(42 heures)

- ▷ Travaux pratiques d'informatique répartie et temps-réel (Ada et JAVA). Module MACS. **ENST** (21 heures)
- ▷ Travaux pratiques d'informatique temps-réel (Ada). Module TRAM. **ENST** (3 heures)
- ▷ Travaux pratiques d'informatique distribuées (Ada/DSA) et d'algorithmique distribuée (Ada). Module ISAR. **ENST** (6 heures)
- ▷ Travaux pratiques de programmation concurrentes et temps-réel (Ada). Module FSET. **UPMC** (6 heures)
- ▷ Travaux pratiques d'informatique temps réel et répartie (Ada). Module ETER. **UPMC** (6 heures)

2006-2007

(87 heures)

- ▷ Chargé de cours langage AADL pour les élèves du module INF-342. **ENST** (3 heures)
- ▷ Travaux dirigés "Shell" pour les élèves admis sur titre. **ENST** (1h30)
- ▷ Conception et encadrement de mini projets d'informatique temps-réel/répartie pour les élèves du module INF-223. **ENST** (6 heures)
- ▷ Projet d'équipe pour les élèves de première année. **ENST** (9 heures)
- ▷ Travaux pratiques de langage C pour les élèves de première année. **ENST** (15 heures)

2. <http://libre.adacore.com/polyorb>

3. <http://aadl.enst.fr/polyorb-hi>

- ▷ Travaux pratiques de langage ESTEREL pour les élèves du module INF-222. **ENST** (12 heures)
- ▷ Travaux pratiques de langage Ada/Temps-réel pour les élèves du module FSET du Master SAR. **UPMC** (4 heures)
- ▷ Travaux pratiques de langage Ada/Temps-réel pour les élèves du module INF-223. **ENST** (12 heures)
- ▷ Travaux pratiques de AADL et intergiciels pour les élèves du module ETER du Master SAR. **UPMC** (4 heures)
- ▷ Travaux pratiques “Protocoles” pour les élèves du module INF-352. **ENST** (6 heures)
- ▷ Travaux pratiques Ada/DSA et communication de groupes pour les élèves du module INF-346. **ENST** (6 heures)
- ▷ Travaux pratiques AADL et informatique Temps-réel répartie pour les élèves du module INF-342. **ENST** (9 heures)

2007-2008

(49h30)

- ▷ Chargé de cours langage AADL pour les élèves du module RAAR du Master SAR. **UPMC** (2 heures)
- ▷ Chargé de cours langage AADL pour les élèves du module ETER du Master SAR. **UPMC** (2 heures)
- ▷ Chargé de cours langage AADL pour les élèves du module INF-342. **ENST** (3 heures)
- ▷ Travaux dirigés sur la théorie de l’ordonnancement temps réel pour les élèves du module INF-342. **ENST** (1h30)
- ▷ Travaux pratiques de langage de programmation système (en langage C) pour les élèves de première année. **ENST** (7h30)
- ▷ Travaux pratiques langage AADL pour les élèves du module RAAR du Master SAR. **UPMC** (2 heures)
- ▷ Travaux pratiques de langage Ada/Temps-réel pour les élèves du module FSET du Master SAR. **UPMC** (4 heures)
- ▷ Travaux pratiques de langage Ada/Ravenscar pour les élèves du module FSET du Master SAR. **UPMC** (2 heures)
- ▷ Travaux pratiques de programmation POSIX (langage C) pour les élèves du module FSET du Master SAR. **UPMC** (4 heures)
- ▷ Travaux pratiques langage AADL pour les élèves du module ETER du Master SAR. **UPMC** (4 heures)
- ▷ Travaux pratiques RT-CORBA (Ada) pour les élèves du module ETER du Master SAR. **UPMC** (2 heures)
- ▷ Travaux pratiques communication de groupes pour les élèves du module INF-346. **ENST** (3 heures)
- ▷ Travaux pratiques de programmation POSIX (langage C) pour les élèves du module INF-223. **ENST** (1h30)
- ▷ Travaux pratiques de programmation Ada pour les élèves du module INF-223. **ENST** (6 heures)
- ▷ Travaux pratiques de CORBA (Ada) pour les élèves du module INF-223. **ENST** (3 heures)
- ▷ Travaux pratiques de socket Java pour les élèves du module INF-223. **ENST** (1h30)
- ▷ Travaux pratiques AADL et informatique Temps-réel répartie pour les élèves du module INF-342. **ENST** (4h30)

2008-2009 (S1)

(18 heures)

- ▷ Chargé de cours langage AADL pour les élèves du module RAAR du Master SAR. **UPMC** (2 heures)
- ▷ Chargé de cours langage AADL pour les élèves du module ETER du Master SAR. **UPMC** (2 heures)
- ▷ Travaux pratiques Rhapsody C++ (UML + Statecharts) pour les élèves du module INF-222. **ENST** (3 heures)
- ▷ Travaux pratiques langage AADL pour les élèves du module RAAR du Master SAR. **UPMC** (2 heures)
- ▷ Travaux pratiques de programmation Ada pour les élèves du module INF-223. **ENST** (3 heures)
- ▷ Travaux pratiques de langage Ada/Temps-réel pour les élèves du module FSET du Master SAR. **UPMC** (4 heures)
- ▷ Travaux pratiques langage LUSTRE pour les élèves du module LS du Master SAR. **UPMC** (4 heures)

2008-2009 (S2)

(180 heures)

- ▷ Chargé de cours “programmation structurée” pour les élèves de 1^{re} année, génie informatique. **ENIS** (15 heures, 4 sections)
- ▷ Chargé de cours “génie logiciel distribué” pour les élèves de 2^e année, génie informatique. **ENIS** (30 heures, 4 sections)

2009-2010

(240 heures)

- ▷ Chargé de cours “programmation structurée” pour les élèves de 1^{re} année, génie informatique. **ENIS** (15 heures, 4 sections)
- ▷ Chargé de cours “génie logiciel distribué” pour les élèves de 2^e année, génie informatique. **ENIS** (30 heures, 4 sections)
- ▷ Chargé de cours “plates-formes de développement d’applications distribuées” pour les élèves de 3^e année, génie informatique, axe ISD (Ingénierie des Systèmes Distribués). **ENIS** (30 heures, 1 section)

- ▷ Chargé de cours “maintenance des logiciels” pour les élèves de 3^e année, génie informatique, axe GL (Génie Logiciel). ENIS (30 heures, 1 section)

Connaissances

Informatique

- ▷ Bases du développement logiciels en plusieurs langages dont Ada, C, C++, JAVA. . .
- ▷ Programmation temps-réel
- ▷ Programmation embarquée
- ▷ Programmation répartie (CORBA, Ada/DSA)
- ▷ Architectures des systèmes informatiques
- ▷ Théorie des langages
- ▷ Théorie de la compilation
- ▷ Maintenance des logiciels
- ▷ Langages synchrones (LUSTRE, ESTEREL)
- ▷ Algorithmique répartie avancée
- ▷ Modélisation formelle des systèmes répartis (automates, réseaux de Petri)
- ▷ Vérification des modèles (Model Checking)
- ▷ Bases de la sécurité informatiques
- ▷ Robotique : notions de base

Réseaux

- ▷ Protocoles et Services
- ▷ Modélisation de protocoles et services (SDL)

Systèmes

- ▷ GNU/LINUX
- ▷ UNIX (Solaris)
- ▷ MACOS X
- ▷ MS Windows

Publications

Articles de Revues

- [1] Hugues, J., Zalila, B., Pautet, L., Kordon, F. : From the Prototype to the Final Embedded System Using the Ocarina AADL Tool Suite. ACM Transactions in Embedded Computing Systems (TECS) 7(4) (jul 2008) 1–25
- [2] Hamid, I., Zalila, B., Najm, E., Hugues, J. : Automatic framework generation for hard real-time applications. Innovations in Systems and Software Engineering : A NASA Journal (mar 2008)
- [3] Zalila, B., Hugues, J., Pautet, L. : An Improved IDL Compiler for Optimizing CORBA Applications. ACM SIGAda Ada Letters XXVI(3) (dec 2006) 21 – 27 **Best student paper award.**

Articles de Colloques avec Actes

- [1] Loukil, S., Kallel, S., Zalila, B., Jmaiel, M. : Toward an Aspect Oriented ADL for Embedded Systems. In : the 4th European Conference on Software Architecture - ECSA'2010, Research Challenge Paper, Copenhagen - Denmark (August 2010)
- [2] Loukil, S., Kallel, S., Zalila, B., Jmaiel, M. : AO4AADL : an Aspect Oriented ADL for Embedded Systems. In : the 10th Annual International Conference on New Technologies of Distributed Systems - Notere 2010, a Demonstration Paper, Tozeur - Tunisia (June 2010)

- [3] Krichen, F., Hamid, B., Zalila, B., Coulette, B. : Designing Dynamic Reconfiguration for Distributed Real Time Embedded Systems. In : the 10th Annual International Conference on New Technologies of Distributed Systems - Notere 2010, Tozeur - Tunisia (june 2010)
- [4] Lasnier, G., Zalila, B., Pautet, L., Hugues, J. : OCARINA : An Environment for AADL Models Analysis and Automatic Code Generation for High Integrity Applications. In : Reliable Software Technologies'09 - Ada Europe, Brest, France (jun 2009)
- [5] Zalila, B., Pautet, L., Hugues, J. : Towards Automatic Middleware Generation. In : 11th IEEE International Symposium on Object-oriented Real-time distributed Computing (ISORC'08), Orlando, Florida, USA (may 2008) 221–228
- [6] Hugues, J., Pautet, L., Zalila, B. : From MDD to Full Industrial Process : Building Distributed Real-Time Embedded Systems for the High-Integrity Domain. In : Monterey Workshop 2006. Volume LNCS., Paris, France (jan 2008) 35–52
- [7] Delange, J., Hugues, J., Pautet, L., Zalila, B. : Code Generation Strategies from AADL Architectural Descriptions Targeting the High Integrity Domain. In : 4th European Congress ERTS, Toulouse, France (jan 2008)
- [8] Hugues, J., Pautet, L., Zalila, B., Dissaux, P., Perrotin, M. : Using AADL to build critical real-time systems :Experiments in the IST-ASSERT project. In : 4th European Congress ERTS, Toulouse, France (jan 2008)
- [9] Zalila, B., Hamid, I., Hugues, J., Pautet, L. : Generating Distributed High Integrity Applications from their Architectural Description. In : 12th International Conference on Reliable Software Technologies - Ada-Europe 2007. Volume LNCS., Geneva, Switzerland (jun 2007) 155–167
- [10] Hugues, J., Zalila, B., Pautet, L. : Combining Model processing and Middleware Configuration for Building Distributed High-Integrity Systems. In : 10th IEEE International Symposium on Object-oriented Real-time distributed Computing (ISORC'07), Santorini Island, Greece (may 2007) 307–312
- [11] Hugues, J., Zalila, B., Pautet, L., Kordon, F. : Rapid Prototyping of Distributed Real-Time Embedded Systems Using the AADL and Ocarina. In : 18th IEEE/IFIP International Workshop on Rapid System Prototyping (RSP'07), Porto Allegre Brésil (may 2007) 106–112
- [12] Hamid, I., Zalila, B., Najm, E., Hugues, J. : A generative approach to building a framework for hard real-time applications. In : 31st Annual NASA Goddard Software Engineering Workshop, Baltimore, USA (mar 2007) 269–278
- [13] Hugues, J., Zalila, B., L.Pautet : Middleware and Tool suite for High Integrity Systems. In : Work-in-Progress session of the Real-Time Systems Symposium (RTSS' -WiP06), Rio de Janeiro, Brésil (dec 2006) 1–4

Documents diplômants

- [1] Zalila, B. : *Optimisation, Déterminisme et Asynchronisme de Souches et Squelettes CORBA pour Systèmes Répartis Temps-réel*. Master's thesis, Université Pierre & Marie Curie, Paris VI (sep 2005)
- [2] Zalila, B. : Configuration et déploiement d'applications temps-réel réparties embarquées à l'aide d'un langage de description d'architecture. PhD thesis, École Nationale Supérieure des Télécommunications (nov 2008)

Rapports Techniques

- [1] Hugues, J., Zalila, B. : PolyORB High Integrity User's Guide. Technical report, École Nationale Supérieure des Télécommunications (jan 2007)
- [2] Vergnaud, T., Zalila, B., Hugues, J. : Ocarina : a Compiler for the AADL. Technical report, École Nationale Supérieure des Télécommunications (jun 2006)

Séminaires

- [1] Zalila, B. : Configuration and Deployment of Distributed Real-time and Embedded Applications Using an Architecture Description Language. In : Méthodes Formelles pour les Systèmes Logiciels et Matériels. (apr 2008)

Langues

Arabe Langue maternelle

Français Bilingue

Anglais Avancé (4/5). Obtention du TOEFL (*Test Of English as a Foreign Language*) en juin 2004 avec un score de 617 (la barre d'admission étant 550)

Divers

Loisirs

- ▷ Musique classique et orientale
- ▷ Romans policiers et science-fiction
- ▷ Cinéma

Autres

- ▷ Permis de conduire catégorie B