

Programme de recherche & développement coopératif
Nano 2017

UGA-LIG/Corse

Rapport Annuel 2016
(1^{er} Janvier au 31 décembre 2016)

CONTACT POUR L'EXECUTION DE LA COLLABORATION

Partenaire

Technique:

M.	Prénom : Jean-François	Nom : Méhaut
Titre	Responsable scientifique	
Téléphone	04 38 78 60 65 04 57 42 46 11	
Adresse mail	Jean-Francois.Mehaut@imag.fr	

Administratif

Mme.	Prénom :Stéphanie	Nom : Bellier
Titre	Responsable administrative	
Téléphone	04 76 51 46 45	
Adresse mail	Stephanie.bellier@imag.fr	

ST Technique:

M.	Prénom :	Nom :
Titre		
Téléphone		
Adresse mail		

ST Administratif :

Contact ST	Prénom : François	Nom : Finck
Titre	Coordinateur du programme Nano 2012 Crolles	
Téléphone	04 38 92 30 72	
Adresse mail	francois.finck@st.com	

Statut du document : version 0.1	UGA-LIG/Corse	Date: Page 1 /8
-------------------------------------	---------------	--------------------

Sous Projet 1: Debugging Interactif

Description résumée des travaux effectués et des résultats obtenus:

Le sous-projet 1 DEMA s'intéresse à la mise au point (*debugging*) interactive des programmes parallèles pour des architectures multi-coeurs à mémoire partagée. Le travail se découpe en deux parties :

- Améliorer le debugging fonctionnel des applications en se basant sur une approche dite centrée sur le modèle de programmation (OpenMP),
- Combiner l'utilisation d'un debugger interactif avec le travail d'analyse et d'optimisation de performances.

Durant la première année du sous-projet 1, nous avons travaillé sur le debugging fonctionnel, en particulier pour des applications utilisant le modèle *fork-join* ou les tâches avec dépendances de données, sur l'environnement de programmation OpenMP. OpenMP est le standard de programmation des processeurs multi-coeurs, à la fois dans le contexte du calcul haute performance, mais aussi des processeurs embarqués multi-coeurs (ARM,...)

L'année 2016 a donc été consacrée au debugging de performance. Notre objectif était de permettre aux développeurs de pouvoir contrôler finement l'exécution et l'analyse du comportement de leurs applications, en particulier en pouvant au moyen de commandes start/stop de piloter le profiling de sections de leurs applications. Le programmeur a ainsi la possibilité de pouvoir analyser certaines sections de son code (en utilisant la syntaxe des points d'arrêt) ainsi que les régions contrôlées manuellement par l'utilisateur. Au niveau des informations de profiling, nous avons choisi de sélectionner celles fournies par le noyau du système d'exploitation (répertoire */proc*), les compteurs gérés par l'environnement de profiling *perf*.

Les travaux du sous-projet 1 ont été présentés en Décembre 2016 à la conférence OpenMPCon à Nara (Japon). Il s'agit d'une conférence organisée par la communauté OpenMP avec la participation de nombreux partenaires industriels et académiques.

Le logiciel Mcgdb développé dans le sous-projet 1 a fait l'objet d'un dépôt logiciel à l'Agence de Protection des Programmes. ST Microelectronics et l'Université Grenoble Alpes sont propriétaires des droits sur Mcgdb.

Une vidéo a été enregistrée pour illustrer l'utilisation de Mcgdb. Elle est disponible avec les livrables D2 et D4.

Les livrables D2 et D4 portant sur le debugger de performance ont été fournis (logiciel et documentation).

Deux workshops du projet DEMA ont été organisés à Grenoble le 24 Juin 2016 et les 12/13 Décembre 2016. Nos collègues Parisiens (Sous-Projet 2) et de STMicroelectronics ont participé à ces workshops.

Statut du document : version 0.1	UGA-LIG/Corse	Date: Page 2 /8
-------------------------------------	---------------	--------------------

L'ingénieur de recherche du sous-projet 1 a également effectué une mission (5 jours) à Paris pour collaborer avec nos partenaires du sous-projet 2 (Inria Parkas).

Ces deux workshops et cette visite à Paris ont permis de réussir à faire interopérer les outils logiciels développés à Grenoble et à Paris. La génération des traces Aftermaths (Sous-Projet 2) est pilotable (activable) depuis Mcgdb (Sous-Projet 1). Cela permet de générer des traces d'exécution de plus faible taille et centrée sur une section de code que le programmeur souhaite analyser.

Rapport d'activité

Description des travaux effectués

L'année 2016 a donc été presque exclusivement consacrée au debugging interactif de performance. L'objectif est de pouvoir profiler en mode interactif des programmes et applications parallèles. Jusqu'à maintenant, l'analyse du profiling se fait généralement) en post-mortem (après exécution), avec une granularité assez importante sur le comportement du programme. L'objectif est de faire ce profiling pendant la phase de mise au point fonctionnel, d'utiliser le debugger pour les aspects interactifs et de laisser au programmeur la possibilité de choisir pendant une exécution les sections de son programme dont il souhaite connaître et évaluer le niveau de performance. Nous souhaitons mettre en avant et détecter des bugs de performance, et aider le programmeur à comprendre pourquoi son programme ne s'exécute pas aussi rapidement qu'il ne le pensait.

Au cours du premier semestre, les travaux du sous-projet 1 ont consisté à intégrer au debugger interactif une interface pour récupérer les compteurs de performance qu'on trouve sur les systèmes d'exploitation comme Linux. Le programmeur peut interagir avec l'exécution de son programme en définissant des points d'arrêt du debugger. Lorsque le programme atteint un point d'arrêt, le programmeur va pouvoir définir les sections de son programme qu'il souhaite analyser en terme de performance. Les compteurs de performance sont récupérés dans le répertoire `/proc` qu'on retrouve sur les systèmes Linux. Les compteurs de performances proposés par l'outil `perf` sont également disponibles au niveau du debugger.

Pendant le second semestre de cette année 2016, nous avons défini et intégré à `MCgdb` les informations que le debugger doit capturer et gérer au niveau du programme OpenMP. Le programmeur pourra par exemple profiler des boucles parallèles OpenMP (`#pragma omp parallel for`). Il est important de noter que `Mcgdb` permet de profiler des applications utilisant le runtime OpenMP GNU (fourni avec les compilateurs `gcc`, `g++`) et le runtime Intel OpenMP (fourni avec les compilateurs Intel `icc` et `clang/LLVM`). Ces différentes possibilités offertes par `Mcgdb` et `Aftermath` ont été utilisées pour découvrir un bug de performance observée sur le benchmark Multi Grille s'exécutant sur une plate-forme avec un grand nombre de coeurs (192) avec une hiérarchie mémoire NUMA.

Résultats obtenus

Au cours du second semestre, nous avons défini un workflow permettant d'aider un développeur d'applications à identifier et corriger des bugs de performance d'applications. Avec les outils développés par les sous-projets 1 et 2, nous avons été capable de détecter une erreur qui se produisait sur un des benchmarks NAS sur une machine à 192 coeurs. Le problème était lié à un mauvais placement des données du benchmark, ce qui provoquait un fort ralentissement de certaines parties du code. De cette expérience, nous avons essayé de définir un workflow pour permettre à des développeurs d'applications d'avoir des outils pour identifier des bugs et des erreurs de performance. Ce résultat a été décrit à la fin du livrable D4. De cette expérience, nous espérons pouvoir la présenter dans une conférence internationale du domaine au cours de l'année 2017.

Statut du document : version 0.1	UGA-LIG/Corse	Date: Page 4 /8
-------------------------------------	---------------	--------------------

Positionnement par rapport aux objectifs

délivrables fournis

D2) Version 1.0 du debugger de performance (Logiciel, Documentation)

D4) Version 2.0 du debugger de performance (Logiciel, Documentation)

Les sources du debugger de performance sont disponibles à l'adresse suivante :

http://dema.gforge.inria.fr/delivrable/2017-01_mcgdb/

Pendant les premiers mois en 2015 du projet DEMA, nous avons principalement travaillé sur le support OpenMP 3 et 4. Nous nous sommes concentrés en 2016 sur le debugger de performance, ce qui a donné lieu aux deux delivrables D2 et D4.

retards éventuels

ressources utilisées par rapport aux ressources planifiées

21 PMs ont été consommés avec le recrutement de Kevin Pouget comme ingénieur de recherche. Son contrat a donc commencé le 1^{er} Avril 2016 et se termine le 31 Mars 2017. Kevin Pouget est en contact avec deux entreprises installées à Grenoble qui pourraient le recruter après son travail sur le projet DEMA du programme Nano2017.

Valorisation des résultats (Publications, brevets, recrutements pour le projet ...)

1 présentation à la conférence OpenMPCon à Nara (Japon) en Décembre 2016

<http://openmpcon.org/conf2016/conference-program/>

Kevin Pouget (UGA, ST Microelectronics), Jean-François Méhaut (UGA, LIG), Miguel Santana (ST Microelectronics)

Programming-Model Centric Debugging for OpenMP

Une vidéo de démonstration de nos outils a été fournie pour aider les développeurs à utiliser les outils DEMA. Elle est accessible à l'adresse suivante :

http://dema.gforge.inria.fr/delivrable/2017-01_mcgdb/

1 ingénieur recruté :

- Kevin Pouget, ingénieur senior, expérimenté en debugging et implémentation de runtime, mars 2015-mars 2017. Kevin Pouget a obtenu un doctorat en 2014 pour des travaux effectués en collaboration entre le LIG et la société STMicroelectronics (convention CIFRE). Son contrat s'arrête en Mars 2017. Kevin Pouget est déjà en contact avec deux entreprises grenobloises (un grand groupe international et une PME). Nous pensons que ses travaux pour le projet DEMA lui ont permis d'acquérir une expérience technique de haut niveau qui servira à une des entreprises du bassin grenoblois.

Statut du document : version 0.1	UGA-LIG/Corse	Date: Page 5 /8
-------------------------------------	---------------	--------------------

Le debugger Mcgdb a été déposé à l'agence de protection des programmes (APP) sous le numéro FR.001.340009.000.S.P.2016.000.10600 (voir la dernière page de ce document)

Les titulaires des droits sont l'Université Grenoble Alpes et la société STMicroelectronics (Crolles2). Mcgdb a été développé dans le cadre de la thèse CIFRE de Kevin Pouget et dans le contexte du projet DEMA du programme Nano2017.

Statut du document : version 0.1	<i>UGA-LIG/Corse</i>	Date: Page 6 /8
-------------------------------------	----------------------	--------------------

Tableau des contenus délivrés au titre de la présente convention

NB. Les livrables doivent être fournis sous forme informatique à la DGE

Sous Projet 1: Debugging Interactif

Contenu délivré au titre de la présente convention	Date prévue	Délivré	Date effective	Commentaires
D2) Version 1.0 du debugger de performance	30/06/2016	o	31/12/2016	
D4) Version 2.0 du debugger de performance	31/12/2016	o	31/12/2016	Livré à temps

Inter Deposit Digital Number

Certificat délivré par

Agence pour la Protection des Programmes

54 rue de Paradis - 75010 PARIS - FRANCE / T. +33(0)1 40 36 03 03 / F. +33(0)1 40 38 96 43

IDDN.FR.001.340009.000.S.P.2016.000.10600

Pour l'œuvre : **McGDB en date du 15 février 2016**

Identité des titulaires de droits* :

UNIVERSITÉ GRENOBLE ALPES
621 avenue Centrale
38401 SAINT-MARTIN-D'HÈRES
FRANCE
Siren : 130021397

STMICROELECTRONICS (CROLLES 2)

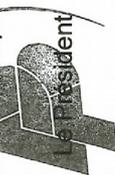
SAS
850 rue Jean Monnet
38920 CROLLES
FRANCE
Siren : 399395581



Adhérent sous le numéro : **16.38.16169**
Support utilisé : 1 CD en double exemplaire

Fait à Paris, le 16/08/2016

Logibox conservée par l'adhérent : 97317
Logibox conservée par l'APPR : 97318 LA
AGENCE POUR LA
PROTECTION DES
PROGRAMMES



(9) Zone géographique : 75010 PARIS
(10) Signature : 305 344 - APE498Z
APP.ASSO.FR

(7) Type de l'œuvre
(8) Année d'enregistrement

(5) Numéro de version
(6) Type d'enregistrement

(3) Numéro de l'organisme d'enregistrement
(4) Numéro d'ordre de l'enregistrement

(1) Inter Deposit Digital Number
(2) Nationalité de l'œuvre

* Le titulaire s'engage à informer l'APPR de toute cession ou aliénation, totale ou partielle, de ses droits de propriété intellectuelle. Seules les inscriptions de type S et C permettent un éventuel accès au programme source.