

# Conférence et Ateliers Développeurs Intel® Software

## Calcul HPC et Intelligence Artificielle - Paris 15 & 16 Octobre 2019

Espace La Rochefoucauld- 11 Rue de la Rochefoucauld, 75009 Paris

**Important:** certaines sessions seront des ateliers pratiques où vous pourrez vous-même expérimenter avec les outils à travers votre navigateur / browser sur une instance dans le Cloud (aucune installation de logiciels nécessaire). **N'oubliez pas votre ordinateur portable / laptop!**

### AGENDA Jour 1 – 15 Oct. 2019

#### Ateliers Intelligence Artificielle et Machine Learning

Heure	Sessions
8:30 – 9:15	<b>Enregistrement &amp; Petit Déjeuner</b>
9:15 – 9:30	<b>Introduction et lancement de la première journée avec Micro Sigma - Intel</b>
9:30 - 10:00	<b>Using Intel Architecture in the world of Data Science and AI (Session in English)</b> Learn how Intel Architecture is successfully being used by data scientists in their artificial intelligence projects. We look at the three ingredients, ( hardware, software, and compilation tools ) that are part of Intel's success story. Stephen Blair-Chappell, Bayncore
10:00 -11:30	<b>Lab: Maximizing Deep Learning Training on Intel® Architecture - Part 1 (Session in English)</b> In this hands-on session on the cloud we show the techniques you can use to get best performance when training Deep Learning models on Intel® architecture. We look at the benefits of both training on your existing Intel based infrastructure and training on the latest generation of Intel® Xeon® Scalable Processors. The results of this session – a trained model using Intel optimized TensorFlow* - will then be used in our next hands-on session. Georg Zitzlsberger, Bayncore
11:30 – 12:00	<b>Pause Café</b>
12:00 - 13:00	<b>Optimisation d'un algorithme de traitement d'image</b> Une grande majorité des tâches en intelligence artificielle est centrée sur le traitement d'images (Convolutional Neural Network) généralement développé en langage Python. Des prétraitements (déconvolution) sont souvent appliqués aux images pour rendre plus robuste les phases d'apprentissage (training) et d'inférence. Python offre la possibilité de rapidement tester et de prototyper ces algorithmes mais ceci se fait souvent au détriment de la performance. Cette présentation introduira un algorithme de déconvolution faisant partie du framework HIPE (Hierarchic Image Processing Engine). Les différents codes présentés sont basés sur la librairie numpy. Nous verrons comment : 1. Profiler du code Python et améliorer sa performance avec Intel® VTune™ Amplifier 2. Rapidement intégrer la version optimisée Intel de numpy 3. Porter des codes en C++ et bénéficier de la vectorisation Damien Dubuc, ANEO
13:00 – 14:00	<b>Cocktail Déjeuner</b>
14:00 – 15:00	<b>Efficient Model Deployment using Intel® OpenVINO™ Toolkit</b> Here we show how easy it is to optimize and deploy a trained model using the Intel® OpenVINO™ toolkit on a range of hardware. We also take a peek into the sample AI solutions that come packaged with OpenVINO™, and see how those could be leveraged for quick prototype building. Stephen Blair-Chappell, Bayncore
15:00 – 16:00	<b>How to get your Python Codes provide near-native performance</b> The optimisation work that Intel has provided to the Python community is considered 'best-in class'. In this session we look at three different ways to get your Python codes to run at near-native performance. We start by describing the underlying performance issues of Python, and then go on to show how these barriers can be overcome. We first look at the issue of composability and the problem with the serialized nature of Python. We then show how your codes can take best advantage of both vectorisation and parallelism following three implementation paths : 'standard' python, numba and then Cython. Stephen Blair-Chappell, Bayncore
16:00 – 16:30	<b>Pause Café</b>
16:30 – 17:30	<b>Machine Learning on Intel Architecture using scikit-learn</b> In this session, we show how to use the optimised version of Intel Python to accelerate the performance of your machine learning algorithms using scikit-learn Georg Zitzlsberger, Bayncore
17:30-18:00	<b>Questions &amp; Réponses</b>
18:00	<b>Cocktail &amp; Networking on the Rooftop</b>

# Conférence et Ateliers Développeurs Intel® Software

## Calcul HPC et Intelligence Artificielle - Paris 15 & 16 Octobre 2019

Espace La Rochefoucauld- 11 Rue de la Rochefoucauld, 75009 Paris

**Important:** certaines sessions seront des ateliers pratiques où vous pourrez vous-même expérimenter avec les outils à travers votre navigateur / browser sur une instance dans le Cloud (aucune installation de logiciels nécessaire). **N'oubliez pas votre ordinateur portable / laptop!**

### AGENDA Jour 2 – 16 Oct. 2019

#### Ateliers Calcul Haute Performance et outils d'optimisation

Heure	Sessions
8:30 – 9:15	<b>Enregistrement &amp; Petit Déjeuner</b>
9:15 – 9:30	<b>Introduction et lancement de la deuxième journée avec Micro Sigma - Intel</b>
09:30 -10:00	<b>Tout savoir sur la nouvelle version Intel® Parallel Studio XE 2020</b> Intel® Parallel Studio XE 2020 est la suite complète d'outils de développement simplifiant la création et la modernisation d'application avec les dernières techniques en matière de vectorisation, de multithreading, de parallélisation multi-nœuds et d'optimisation de la mémoire. Laurent Duhem, Intel
10:00 – 11:00	<b>Les compilateurs Intel® pour le calcul haute performance</b> Cette présentation vous donnera les bases nécessaires à la bonne utilisation des compilateurs Intel® avec une priorité donnée au calcul haute performance. La présentation vous expliquera quelles sont les options essentielles pour accélérer efficacement vos applications. Nous introduirons aussi des notions de parallélisations et de vectorisation avec la norme OpenMP. Enfin, cette présentation abordera les problèmes de répétabilité en introduisant quelques bonnes habitudes à respecter lorsque les résultats doivent être reproduits le plus fidèlement possible. Georg Zilzlsberger, Bayncore
11:00 – 11:15	<b>Pause Café</b>
11:15 – 13:00	<b>Lab: Optimisation d'un code de propagation d'onde en 3D</b> Ce TP propose de découvrir un code de propagation d'onde sur lequel de nombreuses optimisations telles que le threading, la vectorization et le cache blocking vont être implémentées. Vous apprendrez à utiliser Intel® Advisor pour vectoriser votre code, vérifier l'efficacité de vos accès mémoire, comprendre la performance de votre application en utilisant le Roofline Model. Nous utiliserons aussi Intel® VTune Amplifier pour profiler le code, implémenter le threading et détecter des problèmes liés à la mémoire. A la fin de cette présentation, vous serez capable d'utiliser Intel® VTune et Intel® Advisor. Vous pourrez ainsi mettre en pratique vos connaissances dans vos applications. Cédric Andréolli, Intel
13:00 – 14:00	<b>Cocktail Déjeuner</b>
14:00 – 15:00	<b>Legolas++ : vectorisation automatique pour des algorithmes à base de tenseurs.</b> Cet exposé présente la bibliothèque générique Legolas++ qui permet de vectoriser et de paralléliser facilement des algorithmes opérants sur des tableaux multidimensionnels C++ (tenseurs). Par le biais de types paramétriques, Legolas++ permet en particulier d'adapter l'organisation des données des tenseurs à différentes architectures SIMD (SSE, AVX2, AVX512...). La combinaison du parallélisme multi-cœur et des instructions SIMD permet par exemple d'atteindre une accélération de <b>22x</b> sur un processeur Intel à 8 cœurs par rapport à une exécution séquentielle. Laurent Plagne, TriScale innov
15:00 - 16:00	<b>Lab: Comprendre la répétabilité sur les nombres à virgule flottante</b> Ce TP propose de faire découvrir les problèmes liés à la répétabilité des opérations en virgule flottantes. Les architectures actuelles ne permettent pas de garantir l'associativité et la représentation de nombre réels réduite à 32 ou 64 bits implique de nombreuses approximations. L'implémentation du parallélisme dans les applications HPC implique généralement que l'ordre des opérations puisse changer, notamment lorsque l'on utilise des réductions. Nous verrons à travers de simples exemples comment utiliser les compilateurs Intel pour reproduire vos résultats exécutions après exécutions. Cédric Andréolli, Intel
16:00 – 16:15	<b>Pause Café</b>
16:15 – 17:30	<b>Verrou: débogage et optimisation de codes en arithmétique flottante</b> L'arithmétique des ordinateurs est un sujet dont l'étude est plus nécessaire que jamais pour le calcul à haute performance. Les subtiles différences entre l'arithmétique réelle (pour laquelle sont conçus la plupart des algorithmes) et l'arithmétique flottante (qui sera in fine employée par le microprocesseur) se manifestent au travers de symptômes de plus en plus visibles : perte de reproductibilité, accumulation d'erreurs d'arrondis et dérive des résultats... L'utilisation de techniques comme l'arithmétique stochastique, notamment au travers de l'outil Verrou, permet d'aider les développeurs d'outils de calcul scientifique à chaque étape du processus d'analyse : diagnostic des problèmes, identification de leur origine dans le code source, et enfin étude du potentiel d'amélioration en arithmétique à précision mixte François Févotte, TriScale innov
17:30 - 18:00	<b>Questions &amp; Réponses</b>

# Conférence et Ateliers Développeurs Intel® Software

## Calcul HPC et Intelligence Artificielle - Paris 15 & 16 Octobre 2019

Espace La Rochefoucauld- 11 Rue de la Rochefoucauld, 75009 Paris

**Important:** certaines sessions seront des ateliers pratiques où vous pourrez vous-même expérimenter avec les outils à travers votre navigateur / browser sur une instance dans le Cloud (aucune installation de logiciels nécessaire). **N'oubliez pas votre ordinateur portable / laptop!**

### Bios:

#### Laurent Plagne (TriScale innov)

Durant sa thèse de physique au CEA, Laurent a mis au point un logiciel de simulation numérique parallèle tirant le meilleur parti du plus grand superordinateur de France (Cray-T3E). Au cours des vingt dernières années, il a participé à l'évolution du calcul haute performance (multi-processeurs, multi-cœurs, unités vectorielles, GPU...) en devenant un spécialiste reconnu des méthodes mathématiques (particulaires, PIC, éléments finis, octree...) et des techniques informatiques (MPI, OpenMP, TBB, CUDA, PaRSEC...) du HPC.

Laurent a mis au point des logiciels HPC dans des domaines aussi divers que la physique nucléaire, la neutronique, la nanophysique, la physique des plasmas, la thermohydraulique ou la gestion de production.

Tout au long de sa carrière (CEA, RWTH, EDF R&D, TriScale innov), Laurent s'est attaché à transmettre ses connaissances sur le calcul à haut rendement par la rédaction de nombreuses publications scientifiques, l'organisation d'écoles d'été et de workshops, l'encadrement de thèses et l'enseignement du HPC aux professionnels de la simulation.

#### François Févotte (TriScale innov)

D'abord doctorant au CEA puis ingénieur-chercheur à EDF R&D et maintenant au sein de TriScale innov, François se consacre depuis 2005 à la mise au point de méthodes numériques à l'état de l'art et leur implémentation dans des solveurs performants. Il a notamment contribué au développement de la plate-forme industrielle Cocagne, dédiée à la simulation neutronique des cœurs de réacteurs nucléaires.

François a aussi été responsable des sujets exploratoires et méthodes avancées au sein du projet "Performance et Qualité des Simulations" d'EDF R&D. Ses activités l'ont conduit à étudier les problèmes liés à l'utilisation de l'arithmétique flottante dans les codes de calcul. Il est l'un des deux développeurs à l'origine de l'outil Verrou, qui permet d'analyser les instabilités numériques des outils de calcul scientifique industriels.

Toujours enthousiaste à l'idée de partager ses connaissances, François intervient régulièrement comme enseignant à l'ENSTA et dans diverses écoles d'été.

#### Georg Zitzlsberger (Bayncore)

Georg is a former Technical Consulting Engineer at Intel, he is an expert in Intel's Software Development Tools portfolio and Intel Architectures. His current activities include research in the area of Machine Learning and Deep Learning with a particular focus on minimizing simulation errors for weather/climate prediction in the temporal and spatial domains. He is a regular contributor/presenter at Intel Code Modernization and AI Workshops.

#### Stephen Blair-Chappell (Bayncore)

Stephen is Technical Director at Bayncore and was for 18 years a Technical Consulting Engineer at Intel. He currently leads the team of consultants at Bayncore providing HPC and AI training on Intel Architecture.