



PLATEFORME DES RESEAUX DE LA MISSION POUR L'INTERDISCIPLINARITE

Plan Pluriannuel de Formation
2016-2018
réseau Calcul

1. Identification du réseau national

Description du réseau (objet, objectifs) :

Le réseau Calcul a pour objectif de favoriser les échanges au sein de la communauté du calcul, qu'il soit scientifique ou intensif. Pour y parvenir, le réseau mène des actions qui peuvent prendre la forme d'ateliers, de journées thématiques ou d'ANF. Ses activités favorisent la formation, la veille technologique,... L'ensemble de ces actions permet également d'avoir une bonne visibilité sur les acteurs du calcul en France.

Nom du responsable/coordonnées :

Romarc David (david@unistra.fr)

Loïc Gouarin (loic.gouarin@math.u-psud.fr)

Vincent Miele (Vincent.Miele@univ-lyon1.fr)

Nom du référent formation/coordonnées :

Loïc Gouarin (loic.gouarin@math.u-psud.fr)

Site(s) WEB du réseau :

<http://calcul.math.cnrs.fr/>

2. Bilan des 2 années écoulées

a. Bilan des actions nationales (co-)financées par le CNRS pour les années 2013 et 2014 (ANF ou autres)

En 2013

Ecole optimisation

Les porteurs de l'action étaient Romaric David et Loïc Gouarin. Le budget était pris dans le cadre [d'equip@meso](mailto:equip@meso) à hauteur de 5000 euros.

Face au besoin croissant de puissance de calcul et à des applications numériques toujours plus complexes, s'exécutant sur des architectures parallèles en constante évolution (many-coeurs, GPU, machines hybrides, ...), il est nécessaire de pouvoir mieux contrôler l'efficacité et la performance de nos applications.

Pour cela, le réseau Calcul et la Maison de la Simulation ont organisé, dans le cadre d'equip@meso, une école sur l'optimisation à la Maison de la Simulation.

L'enjeu principal de cette école était d'être capable :

- d'identifier les points critiques d'une application dans son environnement d'exécution ;
- à l'aide d'outils adaptés, améliorer son fonctionnement et son utilisation.

En effet, la complexité des outils d'analyse suit celle des architectures et des applications, c'est pourquoi il est nécessaire de permettre à nos communautés d'acquérir rapidement les compétences nécessaires, en lien avec les équipes développant ces outils.

Le programme était le suivant

- une introduction aux architectures actuelles et aux évolutions futures,
- un cours sur les optimisations de base et la vectorisation,
- un cours sur une bonne utilisation des caches,
- un cours sur le benchmarking,
- un cours sur les optimisations parallèles,
- un cours sur l'optimisation des entrées-sorties,
- un cours sur l'optimisation d'applications sur cartes graphiques.

L'école a accueilli 26 stagiaires. Nous n'avons pas réalisé d'évaluation sous forme de questionnaire. Nous pouvons néanmoins affirmer que les avis lors de cette formation étaient très positifs. Les participants nous ont même encouragés pour en faire une deuxième pouvant intéresser leurs collègues (ce que nous avons fait en 2014).

ANF Python avancé en calcul scientifique

Les porteurs de l'action étaient Sylvain Faure et Loïc Gouarin. Le budget total était de 24 000 euros, de même que le budget du CNRS.

L'objectif de l'action était

- Améliorer et enrichir les compétences des personnes utilisant ce langage
- Prototyper et implémenter efficacement un code de calcul grâce aux modules Python dédiés au calcul : manipulation de tableaux multidimensionnels (Numpy), bibliothèques scientifiques (SciPy).
- Être capable d'optimiser son code de calcul en construisant des extensions à l'aide de Cython.
- Pouvoir visualiser les résultats numériques obtenus en temps réel (vtk), ou à l'issue du calcul (Paraview et Visit).
- Avoir des notions des outils de parallélisation sur machines à mémoires distribuées (mpi4py), ou sur carte graphique (PyCUDA, PyOpenCL).
- Avoir de bonnes connaissances en packaging pour la diffusion des codes.

44 stagiaires ont participé à la formation, avec des profils et des origines variés et des domaines d'applications divers (25 CNRS, 12 d'autres ESPT ou organismes publics tels que l'INRA, l'ONIRIS, l'Université du Développement durable et 7 externes du secteur privé, du CEA et du Synchrotron Soleil).

Les objectifs de la formation ont été pleinement atteints et la formation a parfaitement répondu aux attentes des stagiaires.

Cette formation a permis aux utilisateurs d'acquérir de nouvelles compétences notamment en termes de packaging, de profiling et d'optimisation de codes. La répartition entre les aspects plus « théoriques » et les travaux pratiques a été bien articulée. Cette action a favorisé les échanges entre les utilisateurs de Python et les problématiques rencontrées.

La durée de la formation a été estimée trop courte pour pouvoir aborder l'ensemble des modules de façon encore plus approfondie. D'autres besoins de formation complémentaires sont apparus tels que packaging et la conception de documents, la conception d'interfaces graphiques...

En 2014

Atelier "Profilage de codes de calcul" (2 sessions)

Les porteurs de l'action étaient Anne-Sophie Mouronval et Laurent Series.

Cette action s'est déroulée en deux étapes :

- profilage des codes de calcul séquentiels en juin 2014 ;
- profilage des codes de calcul OpenMP en décembre 2014.

Le budget alloué pour la première session était de 630 euros (100 % CNRS) et de 1125 euros pour la deuxième session (100 % CNRS).

Face au besoin croissant de puissance de calcul des applications scientifiques, l'optimisation des codes de calculs est devenue un point clef de la simulation numérique. L'étape préalable à toute action d'optimisation consiste à analyser les performances du code (profilage) afin d'identifier les points critiques de celui-ci dans son environnement d'exécution. Pour ce faire, il existe de nombreux outils d'analyse qu'il convient de savoir choisir et utiliser en fonction de ses besoins.

Le réseau Calcul en collaboration avec le Mésocentre de Centrale Paris ont accueilli Laurent Gatineau (Support applicatif, NEC HPC Europe) pour un premier atelier sur le profilage des codes de calcul séquentiels. Cette formation a eu lieu en juin 2014 dans les locaux de l'Ecole Centrale Paris.

L'enjeu de cette première session de l'atelier "Profilage de codes de calcul" était de donner aux participants de bonnes bases pour débiter en profilage.

Une deuxième session, dédiée au profilage des applications OpenMP, a eu lieu le 10 et 11 décembre dans les locaux de Centrale Paris. L'intervenant de cette session était Brian Wylie du Jülich Supercomputing Center.

L'enjeu de cette deuxième session était de donner aux participants de bonnes bases pour débiter en profilage des applications OpenMP.

La première session a accueilli 20 stagiaires et la deuxième 13.

La première session de l'atelier "Profilage de codes de calcul" a été dédié aux applications séquentielles et a eu lieu en juin 2014.

Cet atelier pratique comportait :

- une introduction au profilage des codes séquentiels ;
- une présentation des outils Gprof et PAPI ;
- des travaux pratiques sur machine utilisant ces outils.

L'atelier a accueilli 20 participants dont 16 ont rendu un questionnaire d'évaluation.

Ces participants étaient principalement des doctorants et post-doctorants (10 d'entre eux). Plusieurs ingénieurs (9) et un chercheur ont également suivi l'atelier. Plus de 80% des participants se sont déclarés très satisfaits de l'atelier, celui-ci ayant tout à fait répondu aux attentes de plus de 60% d'entre eux.

D'un point de vue général, les compétences pédagogiques de l'intervenant ont été soulignées par les participants. Le fait que l'atelier était bien adapté aux débutants et alternait théorie et pratique a été très apprécié, certains regrettant tout de même de ne pas avoir pu passer plus de temps sur les exercices.

Enfin, le questionnaire ayant mis en évidence un fort besoin de formation concernant le profilage des applications multithreadées, une session dédiée à ces applications a été proposée en décembre.

Cette deuxième session comportait :

- une introduction au profilage des codes OpenMP ;
- une présentation d'outils comme Scalasca ;

- des travaux pratiques sur machine utilisant ces outils.

Cet atelier a accueilli 13 participants composés de 9 doctorants et post-doctorants et 4 ingénieurs.

Le questionnaire d'évaluation a été rendu par 8 participants.

Ces participants ont donné une appréciation globale très satisfaisante pour une grande majorité d'entre eux.

L'atelier a répondu aux attentes de chacun des participants qui ont apprécié les compétences pédagogiques de l'intervenant.

Enfin, le questionnaire a mis en évidence un besoin en profilage d'applications de type MPI et GPGPU.

Ecole "optimisation"

Les porteurs de l'action étaient Romaric David et Loïc Gouarin. Le budget était de 10000 euros avec 50 % pour le CNRS.

Face au besoin croissant de puissance de calcul et à des applications numériques toujours plus complexes, s'exécutant sur des architectures parallèles en constante évolution (many-coeurs, GPU, machines hybrides, ...), il est nécessaire de pouvoir mieux contrôler l'efficacité et la performance de nos applications.

L'enjeu principal de cette école est d'être capable :

- d'identifier les points critiques d'une application dans son environnement d'exécution ;
- à l'aide d'outils adaptés, améliorer son fonctionnement et son utilisation.

En effet, la complexité des outils d'analyse suit celle des architectures et des applications, c'est pourquoi il est nécessaire de permettre à nos communautés d'acquérir rapidement les compétences nécessaires, en lien avec les équipes développant ces outils.

Il s'agissait de la 2ème édition de cette école, qui a rencontré un vif succès en 2013. L'école était organisée en non-résidentiel sur le site de l'Université de Strasbourg. Grâce au Centre de Calcul de l'Université de Strasbourg, elle a pu bénéficier du sponsoring d'intégrateurs et de constructeurs informatiques.

Le nombre de stagiaires était de 25.

Cette école de 4.5 jours permet de rassembler un public très intéressé par l'optimisation d'applications existantes, développées dans un contexte de recherche. Le propre des modules proposés est de trouver une application presque immédiate dans les logiciels des chercheurs.

Les intervenants viennent d'horizons très différents, souvent de laboratoires d'excellence dans le domaine. Grâce à la pédagogie des intervenants, les sujets abordés, parfois ardu, ont été bien reçus par l'audience. La période choisie (Juillet) permettait aussi aux enseignants chercheurs de participer à l'école sur toute sa durée.

Cette 2ème édition de l'école a montré que le programme était parfaitement cohérent. Néanmoins, lors d'une prochaine édition de l'école, il serait souhaitable d'éviter certaines répétitions sur des aspects techniques, permettant de dégager du temps pour des nouveautés.

École "Informatique pour le calcul scientifique, bases et outils"

Les porteurs de l'action étaient Anne-Sophie Mouronval, Laurent Series, Romaric David, Bernard Dussoubs. Le budget total était de 22 000 euros, de même que le budget du CNRS.

La simulation numérique est un outil majeur en recherche fondamentale et appliquée, et s'impose comme une nouvelle discipline entre théorie et expérience. L'évolution des matériels informatiques et des logiciels a ont débouché sur l'augmentation considérable de la puissance de calcul et de la taille des problèmes pouvant être traités. Le développement et l'optimisation de codes de calcul pour les nouvelles architectures matérielles deviennent des étapes critiques pour atteindre les performances promises à un niveau d'excellence. C'est la raison pour laquelle de nombreuses formations existent sur ces thématiques pointues (programmation parallèles, optimisation sur architectures hybrides, etc.).

Cependant, les connaissances requises pour suivre ces formations sont rarement acquises au cours des formations initiales et il est souvent nécessaire d'avoir une longue expérience en développement de codes en calcul scientifique pour profiter pleinement des concepts abordés. On se rend compte alors qu'il n'existe pas ou peu de formations abordant les connaissances de base et les outils associés pour développer des codes de calcul dans les meilleures dispositions. Les personnes se forment seules et n'ont pas forcément les bons automatismes.

Partant de ce constat, nous proposons un enseignement des connaissances indispensables au développement et à la compréhension des méthodes de l'informatique scientifique moderne, à destination de ce public. Il n'en demeure pas moins qu'acquérir des compétences en développement de codes scientifiques est un processus long et itératif. Il s'agira donc ici de permettre aux participants de mettre le pied à l'étrier, afin d'éviter les écueils d'une démarche autodidacte. L'objectif principal est de donner des bases solides pour le développement de codes scientifiques. A l'issue de la formation, les participants auront gagné en autonomie et sauront identifier les méthodes et outils logiciels les plus pertinents pour leur problématique.

La formation a accueilli 37 stagiaires dont 35 ont rendu un questionnaire d'évaluation. Ces participants étaient principalement des ingénieurs (22 d'entre eux), le reste se partageant entre lesdoctorants/post-doctorants (7) et les chercheurs/enseignants chercheurs (6).

Pour chaque cours, nous avons demandé l'adéquation entre celui-ci et les besoins des participants (de très satisfaisant à très insuffisant) et le cas échéant, la mise en pratique du cours lors de TPs. Chaque cours a reçu un pourcentage d'avis positifs (très satisfaisant ou satisfaisant) nettement supérieur à 50%, puisqu'il varie de 69% à 100%. De même, l'adéquation entre cours et TD a toujours été plébiscitée.

Plus généralement, 100% des participants ont considéré que l'École répondait de façon satisfaisante ou très satisfaisante à leurs attentes, et tous ont vanté l'organisation et la qualité de l'hébergement,

au détail près du Wifi dont plusieurs ont souligné la mauvaise qualité. Certains auraient préféré disposer des supports de cours avant ceux-ci.

Enfin, pour ce qui est du rythme, quelques participants auraient préféré qu'il n'y ait pas de session après le repas du soir et que la durée de la pause déjeuner soit plus longue.

En conclusion, la formation a été bien accueillie, les participants ont apprécié le programme ainsi que la qualité des cours des intervenants. Parmi les thématiques qui n'ont pas été abordées lors de cette école, plusieurs participants auraient souhaité que l'on parle du profilage de code. Enfin, le fait que cette Ecole puisse être organisée en région a souvent été évoqué.

Atelier "C++ : vers 2011 et au-delà"

Le porteur de l'action était Loïc Gouarin.

Le budget alloué pour cette formation était de 3000 euros de même que le budget du CNRS.

Le langage C++ a connu un regain d'intérêt depuis l'adoption du standard 2011 et de 2014. Ces évolutions du langage ont pour objectif de rendre plus sûr, plus performant et plus simple le développement d'applications critiques et de logiciels d'infrastructures.

Cette formation avait pour objectif de parcourir les différentes nouveautés du langage en s'attachant à démontrer les applicabilités et leurs avantages dans des scénarios concrets.

La formation a réuni une trentaine de stagiaires. Ils étaient tous satisfaits voire très satisfaits. Certains ont néanmoins trouvé le niveau un peu élevé et ont demandé si il serait possible d'avoir une formation un peu plus sur les bases (ce qui a été fait en 2015).

b. Bilan des actions régionales les plus importantes (co-)financées par le CNRS pour les années 2013 et 2014

Merci de vous limiter aux actions qui vous sembleront avoir eu l'impact le plus important selon vous. Un maximum de 5 actions par an – celles-ci pouvant être regroupées par thème même si elles se sont tenues dans des lieux différents.

Pour chaque action financée, merci d'indiquer :

- *Le nom & l'année de l'action*
- *Le nom du porteur de l'action*
- *Le budget total alloué et la part CNRS*
- *L'objectif détaillé de l'action*
- *Le bilan quantitatif (nombre de personnes concernées)*
- *Le bilan qualitatif et une évaluation critique de cette action*
- *Le cas échéant, comment s'articulent les formations organisées en région par rapport aux formations organisées nationalement (et réciproquement) ?*

c. Bilan général

Présentez ici une analyse globale des actions de formation des 2 années écoulées. Cette analyse se devra d'être critique et de montrer les points forts et les points faibles du plan mis en œuvre.

Les formations du réseau Calcul ont réuni en moyenne une trentaine de stagiaires. Nous pouvons compter huit actions de formations sur deux ans (deux n'étant pas financées par le CNRS). Les avis sont toujours positifs voire très positifs. Le choix des formations reflète donc un réel besoin au sein de notre communauté. La réalisation du plan de formation de l'an dernier nous a permis d'avoir une vision claire sur les thèmes à aborder.

Pour les années 2013 et 2014, nos formations ont balayé le spectre des langages de programmation et des notions d'optimisation de codes de calcul parallèles et séquentiels : un spectre plus large aurait pu être envisagé. Ainsi le gros travail fourni l'an dernier nous permet aujourd'hui de diversifier nos offres de formations et nous serons vigilants à maintenir ce cap pour les années futures.

Enfin, le choix des thèmes de formations à proposer se fait -la plupart du temps- en comité de spécialistes ou à partir des réponses aux questionnaires des différentes formations où il est demandé les besoins futurs des stagiaires. Nous pourrions envisager de faire émerger les thèmes depuis la communauté; dans cette voie, nous avons sondé en 2014 les abonnés de notre liste (1400 abonnés) sans réel succès (45réponses).

Il est également nécessaire de trouver plus de personnes pour organiser ces formations si nous voulons monter le nombre de formations annuelles. Il est évident que ça ne peut pas se faire avec 3 ou 4 personnes. Néanmoins, l'organisation de telles actions peut faire peur à un agent sans expérience. Il faudra donc être vigilant de faire pendant deux ou trois ans des formations avec deux organisateurs (un novice et un expérimenté) afin d'augmenter le pool d'organisateur.

3. Plan de Formation 2016 – 2018

a. Prospective Métier et / ou technologique pour la communauté visée

Donnez ici une analyse prospective de l'évolution à court, moyen et long termes des métiers et de la technologie qui concernent votre réseau. Explicitez les éléments sur lesquels reposent cette analyse (enquête, document, réflexions internes). Donnez des indications de rapidité de ces évolutions et des difficultés que pourraient rencontrer les agents et in fine les expériences scientifiques si ces difficultés existent.

Les réflexions synthétisées par le comité de pilotage ont permis le regroupement en cinq grandes thématiques. Pour chacune d'elles, nous décrivons le contexte actuel, les besoins ainsi que les formations qui nous semblent nécessaires.

Génie Logiciel pour le calcul

Constats

Le développement de code ou logiciel de calcul fait partie intégrante du travail de recherche et prend une place de plus en plus importante. Les publications scientifiques s'appuient en grande majorité sur les résultats obtenus avec ces codes pour valider les parties théoriques. Ces codes peuvent être de quelques lignes ou de plusieurs milliers et constituer une brique d'un code existant ou un nouveau logiciel. Il existe également un engouement croissant sur la reproductibilité des codes de calcul afin de retrouver les résultats publiés. Ainsi la qualité et la sécurité (au sens de la garantie de justesse des calculs) des codes de calcul est une pierre angulaire du processus de recherche; il est donc nécessaire d'importer les savoirs et pratiques du génie logiciel pour les transposer dans le contexte du calcul numérique.

De plus, le développement logiciel fait intervenir un grand nombre de personnes ayant des niveaux très variés. Or, le travail pour développer, diffuser et maintenir du code nécessite des compétences pointues, spécifiques afin de garantir une qualité logicielle, une efficacité pour des logiciels robustes, pérennes, validés. Ces objectifs ne peuvent être atteints sans connaissances spécifiques, sans l'application de méthodes, l'utilisation d'outils de génie logiciel largement utilisés et éprouvés.

Enfin, les techniques, les outils, les méthodes de développement sont en perpétuelle évolution depuis des décennies et il est probable que le génie logiciel connaisse encore des transformations majeures dans les prochaines années.

Besoins

Depuis un certain nombre d'années, le réseau Calcul et maintenant DevLog ont travaillé avec le projet Plume pour organiser les journées ENVOL (Formation pour le dEveloppement et la ValOrisation des Logiciels en environnement de recherche). Malheureusement, le projet Plume est dans un état végétatif. Il est important que le réseau Calcul et DevLog travaillent ensemble pour continuer à organiser ces journées dont l'engouement témoigne de l'attente de la communauté à proposer des actions visant à améliorer les pratiques de développement.

Afin que les organismes publics et plus particulièrement le CNRS promeuvent les logiciels développés en leur sein, il est important que les agents aient de bonnes compétences en génie logiciel : utilisation d'outils de type forge, intégration continue, tests, documentations, packaging, ...

Il est également important d'avoir de bonnes connaissances sur les licences permettant de diffuser librement le logiciel tout en protégeant ses auteurs.

Les nouveaux entrants et plus particulièrement les doctorants (qui fournissent une grande part du développement des logiciels durant leur thèse) n'ont pas forcément reçu de formations en génie logiciel. L'apport de ces connaissances les aiderait grandement dans leur travail et leur permettra de gagner en efficacité et en qualité.

Perspectives

Pour répondre à ces besoins, une ou plusieurs formations pourront être organisées afin de permettre au développeur d'avoir les bons automatismes, de l'aider dans sa veille technologique sur les outils qui peuvent améliorer grandement son travail ...

Une ANF ENVOL tous les deux ans serait nécessaire. Il pourrait s'en suivre des ateliers de quelques jours (dont certains pourront être réalisés avec DevLog) sur des points très ciblés du génie logiciel : le débogage, l'intégration continue, les gestionnaires de versions ...

Optimisation, parallélisation et runtime

Constats

Afin de proposer des machines toujours plus puissantes, les constructeurs se tournent vers des architectures de plus en plus complexes.

Cette complexité se manifeste non seulement dans la nature massivement parallèle des calculateurs, mais aussi dans l'hétérogénéité des nœuds (CPU + GPU ou Xeon Phi) utilisés afin de répondre à des contraintes énergétiques.

Dans ce contexte, le développement d'applications efficaces est très difficile, d'autant plus que ces architectures, leurs composants (processeurs, accélérateurs, systèmes de fichiers...) et les outils pour en tirer parti (compilateurs, bibliothèques...) sont en constante évolution.

Besoins

Savoir se repérer dans ce paysage est difficile pour les développeurs de code de calcul :

- que peut-on attendre des nouvelles architectures ?
- une application peut-elle être adaptée pour l'ensemble des architectures proposées et pour quel effort ?
- quel type de programmation faut-il privilégier dans un souci de pérennité, portabilité et efficacité ?

Perspectives

Pour répondre à ces besoins, une ou plusieurs formations (écoles ou ateliers thématiques) pourront être organisées afin de :

- sensibiliser les développeurs aux concepts clés des nouvelles architectures (many-cores, GPU...)
- maîtriser les outils qui permettent d'en tirer parti, en particulier ceux qui permettent d'utiliser efficacement plusieurs architectures (stratégies de compilation, programmation par directive, runtime ...)
- maîtriser les outils de profilage afin d'identifier les points critiques de l'application dans son environnement d'exécution (CPU, accélérateur)

Bibliothèques de calcul

Constats

Les bibliothèques logicielles scientifiques offrent plusieurs avantages pour le développement de codes de calcul. En effet, dans le cas où elles sont utilisées pour les fonctionnalités basiques mais essentielles des codes (produit scalaire, matriciel, calcul de vecteurs propres...), elles peuvent simplifier le codage et surtout améliorer grandement ses performances. Par ailleurs, elles permettent de pouvoir tirer parti des architectures de plus en plus complexes proposées par les constructeurs des machines.

Besoins

Les bibliothèques sont nombreuses et il est difficile de faire un choix en fonction du langage, de l'architecture cible de l'application. De plus, leur interfaçage avec l'application et leur réglage pour tirer pleinement parti de leurs fonctionnalités peuvent constituer des tâches difficiles.

Perspectives

Pour répondre à ces besoins, il serait intéressant d'organiser :

- des formations courtes (de type atelier) dédiées à une seule bibliothèque afin de bien maîtriser ses fonctionnalités pour en tirer parti efficacement
- des formations plus longues (de type école) dédiées à plusieurs bibliothèques dont les fonctionnalités sont similaires (comme la résolution de système linéaires par exemple) afin que les participants puissent choisir les outils qui répondent le mieux à leurs besoins

Données et visualisation

Constats

Le monde du calcul scientifique et du calcul intensif subit une mutation due à l'omniprésence des besoins liés aux données (accessibilité, stockage, traitement, visualisation). Dans ce nouveau paysage, le manque de compétences sur les spécialités liées au traitement des données au sens large est un constat majeur auquel il est nécessaire de faire face (cf livre blanc sur l'informatique en appui à la recherche au CNRS).

Besoins

Les besoins spécifiquement liés aux pratiques du calcul scientifique et du calcul intensif sont essentiellement associés à des problématiques de visualisation et d'analyse des données.

On peut distinguer plusieurs types d'usages :

- Les besoins en terme de visualisation de la plupart des utilisateurs peuvent être satisfaits par des logiciels libres tels que ParaView ou Visit. Si il peut être relativement facile de se former seul à ces logiciels grâce aux différents tutoriels se trouvant sur le net, ces outils restent très spécialisés et un accompagnement à leur emploi en améliore beaucoup l'efficacité. Par ailleurs, pour des utilisations plus avancées (par exemple via des scripts ou sur un cluster graphique parallèle avec une visualisation déportée), les formations sont un atout important.

- Lorsque les logiciels précédents ne conviennent pas ou lorsque l'on doit développer une visualisation « sur mesure », par exemple pour l'intégrer dans un logiciel via une interface utilisateur, il devient nécessaire de savoir utiliser directement des bibliothèques graphiques (OpenGL, VTK,...). Peu de formations existent dans ce domaine, d'autant que chaque année apparaissent de nouveaux outils susceptibles d'être intéressants pour de la visualisation scientifique.

- Visualiser de gros volumes de données est aussi un enjeu important, la visualisation « in situ » (pendant que le calcul tourne en utilisant les données disponibles en mémoire) est un concept très récent, dont la pratique va forcément se développer.

Perspectives

Pour répondre à ces différents besoins, une ou plusieurs formations (écoles ou ateliers thématiques) à différents niveaux (débutant, avancé) pourront être organisées.

La partie Data/Visualisation est aussi un domaine important de veille technologique.

Ainsi, outre les besoins décrits ci-dessus, des outils de rendu 3D arrivent petit à petit dans la communauté scientifique, il faut se tenir informé de ce qu'ils peuvent apporter à la visualisation de résultats de simulations numériques.

Langages de programmation

Constats

Les langages de programmation permettent aux développeurs et aux utilisateurs de communiquer avec les composants de la machine (processeur, carte graphique, mémoire ...). Les langages existants sont en perpétuelle mutation afin d'utiliser pleinement les nouvelles architectures. Les normes des langages évoluent donc en offrant de nouvelles fonctionnalités et en enlevant des parties devenues obsolètes. La bonne connaissance du langage utilisé permet d'avoir des codes de calcul de qualité et performant.

Il faut également rappeler que la liste des langages de programmation n'est pas figée. Des langages meurent et naissent chaque jour. La naissance de nouveaux langages peut fortement intéresser la communauté du calcul en France. On peut par exemple citer la venue du nouveau langage Julia qui n'a que deux ans, mais que l'on verra sûrement utilisé, à terme, dans de nombreux codes de calcul scientifiques et intensifs.

Besoins

Comprendre toutes les possibilités d'un langage permet d'obtenir des codes de calcul concis et de qualité. Les formations initiales n'abordent que très peu de langages. Il est donc primordial d'avoir des formations sur les fonctionnalités de tel ou tel langage avec des niveaux très variés : débutant, intermédiaire et expert.

Les formations initiales évoluent également et proposent de nouveaux langages. C'est le cas de Python qui est maintenant étudié au lycée et en classe préparatoire. Il est important de poursuivre ce processus d'apprentissage tout au long de la vie professionnelle en proposant des formations avancées et ciblées sur des applications en lien avec le calcul scientifique et intensif.

Perspectives

Pour répondre à ces différents besoins, une ou plusieurs formations (écoles ou ateliers thématiques) à différents niveaux (débutant, avancé) pourront être organisées.

Le réseau Calcul a déjà organisé plusieurs formations autour de Python et poursuivra son effort, car l'utilisation de ce langage est de plus en plus importante au sein de la communauté. Une collaboration en cours avec DevLog pour diffuser les connaissances nécessaires à l'utilisation de Python, doit être renforcée pour les prochaines années. Ce type d'actions communes pourra s'étendre à d'autres langages.

Les normes des nouveaux langages ainsi que la bonne utilisation des compilateurs sont également un point important qui devra être abordé.

b. Orientations / Stratégie de formation du réseau

Donnez ici les 2 ou 3 grands lignes de la stratégie que va mettre en œuvre le réseau selon les besoins identifiés ci-dessus ainsi que le type et le format d'action qui vous semblera le plus pertinent pour y répondre. Veillez à ne pas limiter ces formats aux ANF et à mobiliser d'autres types d'actions (tutorat, atelier, éditions...).

Le réseau Calcul propose d'accompagner les personnels dans la mise en place de bonnes pratiques, applicables dans leur quotidien ou diffusable en interne auprès des collègues (dissémination de la connaissance depuis notre réseau).

La stratégie de formation du réseau Calcul est holistique; elle balaye en effet un large spectre depuis les étapes premières liées à l'organisation de l'environnement de calcul (choix du langage, pratique du génie logiciel) jusqu'aux étapes finales de valorisation des résultats des calculs (traitement des données, visualisation), en proposant d'utiliser au mieux l'existant (bibliothèques) ou en préparant à opérer des développements de pointe (parallélisation, runtime).

Les formats sont divers et pourront prendre les formes suivantes.

ANF

Les ANF (actions nationales de formation) permettent au réseau Calcul d'avoir de gros financements pour réaliser des formations en résidentiel sur quatre ou cinq jours dans des structures permettant d'accueillir une cinquantaine de personnes dont la majorité sont CNRS ou dans une unité mixte.

La forme résidentielle permet de favoriser les échanges entre les intervenants – participants mais également entre les participants.

Atelier

Un atelier est un événement sur deux ou trois jours où seul le ou les intervenants se déplacent. Contrairement aux ANF, il n'y a pas de résidentiel puisque la plupart des participants vivent proche du lieu de formation. Le réseau Calcul expérimente cette formule depuis un peu plus d'un an maintenant.

Les intérêts de ce format sont

- Le faible coût financier : frais de transport et hébergement du ou des intervenants, pauses cafés et éventuellement prise en charge des repas. Il n'y a pas de frais pour la location de salle car, en règle générale, les locaux sont prêtés par la structure qui accueille.
- Il est facile de rejouer la formation dans différentes régions de France.
- Les thèmes sont très précis et il ne faut donc généralement qu'un intervenant pour l'ensemble de ces deux ou trois jours. Il permet donc de faire du perfectionnement.
- Les participants n'ont pas besoin de se libérer de leurs obligations familiales.

En revanche, le gros inconvénient de ce format est que les échanges sont limités puisque la formation n'est pas en résidentiel et que les participants rentrent directement chez eux.

MOOC

Les MOOC, ou Massive Open Online Courses, ont récemment fait leur apparition dans le paysage de l'enseignement supérieur et de la recherche. Ce nouvel outil pédagogique innovant ouvre des perspectives très intéressantes pour le réseau calcul.

Plusieurs utilisations se dessinent :

- assurer un niveau homogène aux participants (pré-requis) en amont d'une formation plus avancées,
- pérenniser une formation récurrente (comme la formation python pour le calcul),
- permettre l'accès à une formation à des personnes n'ayant pas pu la suivre.

Plusieurs MOOCs autour du calcul et des aspects plus numériques (MOOC INPROS : <http://inpros.univ-lyon1.fr/> par exemple) sont en cours de montage, et dans lesquels certains membres du réseau sont très actifs.

Cette expérience, ainsi que les contacts créés (notamment avec FUN, France Université Numérique) vont permettre au réseau d'investir ces nouveaux dispositifs dans les prochaines années.

Site web

Enfin, à chaque formation sous forme d'ANF ou d'atelier, les supports des cours et Tps sont mis en ligne avec une correction. Les participants peuvent alors refaire les exercices facilement.

L'autre avantage d'avoir les supports en ligne est que les personnes n'ayant pu venir à la formation soit par manque de places, soit pour des contraintes personnelles, peuvent s'autoformer.

Notebook

On voit de plus en plus de cours en ligne utilisant les notebooks de IPython (<http://lorenabarba.com/blog/cfd-python-12-steps-to-navier-stokes/>). Ces supports permettent de faire des cours de qualité avec l'ajout de vidéos, d'images, de codes, ... Les rendant interactifs.

Ce support pourrait être très intéressant pour proposer des cours de base ou de remise à niveau sur les thèmes cités précédemment.

L'autre avantage est que ces notebooks peuvent être téléchargés et modifiés à volonté permettant ainsi de tester différentes approches pour se perfectionner ou avoir ces propres annotations.

c. Plan de Déploiement

Donnez Les détails de la mise en œuvre des orientations du réseau pour les années 2017 et 2018.

Pour chaque formation, les publics sont toujours les mêmes : chercheurs, ingénieurs, doctorants et post-doctorants. Des personnes du privé sont bien entendu les bienvenues.

Génie logiciel pour le calcul

Le réseau Calcul et le réseau DevLog souhaitent s'assurer que d'autres éditions de l'école ENVOL puissent voir le jour, indépendamment de l'avenir de PLUME. Nous avons été de plus contacté par la DSI du CNRS pour faire cette école sous forme d'ANF en 2016. Cette formation est importante puisqu'elle offre une initiation à l'ensemble des outils fondamentaux pour développer des applications. Le réseau Calcul et le réseau DevLog souhaitent que cette action soit récurrente tous les deux ans. Il est donc à prévoir une nouvelle édition en 2018.

Fin 2015, nous allons proposer une journée sur les outils pour de l'intégration continue. Lorsque l'on développe des codes de calcul seul ou à plusieurs, il est important de ne pas casser les développements précédents, de s'assurer que nos développements fonctionnent sur différents systèmes d'exploitation, ... Prendre cette problématique en compte améliore la qualité des codes et offre donc une meilleure image du travail réalisé. Un atelier d'initiation de trois ou quatre jours est envisagé en 2017.

Optimisation, parallélisation et runtime

Il est important de continuer à former la communauté à l'optimisation des applications sur les architectures actuelles.

Cependant, il ne sera peut-être pas nécessaire de demander des ANF ou des ateliers sur ce point puisque ces projets de formations rentrent directement dans le cadre de l'animation scientifique [d'equip@meso](mailto:equip@meso) où le réseau Calcul est partenaire.

Les nouvelles architectures de calcul seront dotées de plusieurs composants dits accélérateurs interconnectés entre eux et au processeur : carte GPU, XEON PHI, ... On voit apparaître de gros changements dans les interfaces de programmation pour le calcul parallèle comme openMP ou openACC permettant aux développeurs de s'affranchir du choix du composant effectuant les calculs. Le réseau Calcul devra proposer des ateliers d'initiation et de perfectionnement sur ces outils permettant ainsi à la communauté du calcul d'utiliser pleinement ces nouvelles ressources de calcul. On peut envisager un atelier par an.

Bibliothèque d'algèbre linéaire

Nous avons organisé en janvier 2015 une journée sur la résolution du problème de Poisson en présentant les différentes manières et les différents outils existants pour résoudre ce problème. Ce thème est particulièrement intéressant car il touche de nombreuses communautés qui sont bien différentes (informatique, mathématiques, plasmas froids, ...).

La journée a été bien accueillie et utilisateurs et développeurs de bibliothèques d'algèbre linéaire se sont rencontrés. Il serait intéressant de poursuivre cette démarche en proposant un atelier en 2017 ou 2018 sur la mise en œuvre de la résolution de ce problème en utilisant des bibliothèques d'algèbre linéaire. Cet atelier de trois jours sera d'un niveau intermédiaire ou avancé car il s'adressera à un public averti.

Il existe également de nombreux outils « tout en un » offrant une grande variété de méthodes pour résoudre des systèmes linéaires et non linéaires : PETSc, Trilinos, paralution. Ils offrent de nombreux formats de données facilitant grandement la résolution de ces systèmes par des méthodes numériques bien connues. Elles sont néanmoins peu connues ou font peur car leur taille peut faire croire qu'il sera difficile d'obtenir des résultats dans des délais raisonnables. Des ateliers de deux ou trois jours sur chacune d'elles pourraient permettre aux personnels de bien utiliser ces outils et de montrer leur efficacité. Le niveau sera initiatique.

Données et visualisation

L'an dernier nous avons écrit :

« Les besoins concernant les aspects visualisation et analyse des données sont de différents niveaux :

- Dans les 3 prochaines années, il sera nécessaire de mettre en place des formations de niveau moyen pour la maîtrise des outils classiques de visualisation. Elles prendront la forme d'ANF afin d'offrir à la communauté un large panel de ce qui existe dans le domaine.*
- Il faudra envisager aussi la tenue d'ateliers sur des méthodologies plus avancées et plus ciblées comme la visualisation in-situ. Ces ateliers devront être mis en place tout de suite après les ANF décrites précédemment.*
- La particularité du traitement et de la visualisation des gros volumes de données nécessitera un rapprochement avec des structures plus spécialisées, et la mise en place d'action de type formation pour installer dans la communauté calcul scientifique des pratiques encore balbutiantes. »*

Outre l'ANF que nous allons proposer pour 2016 sur la visualisation de gros volume de données, nous avons cette année discuté avec plusieurs réseaux ayant des problématiques similaires preuve que ce thème est très important pour tout un ensemble de communautés.

Une action commune est envisagée avec les réseaux Calcul, RESINFO, DevLog, RBDD, RDE autour du « cycle de vie des données, de la production à l'archivage ». Cette action devra être discutée plus concrètement avec la MI sur sa forme. Elle est envisagée pour 2017.

Sous l'impulsion de la MI, un groupe de travail autour du traitement d'image en microscopie. Là encore, les données à traiter sont gigantesques et les instruments toujours de plus en plus précis génèrent des données que ne sont pas toutes utiles pour une expérience données. Il faut alors des applications de pointe pour pouvoir traiter ces images. Une journée aura lieu fin décembre en partenariat avec les réseaux Calcul, DevLog, RBDD, RTMFM, REMISOL et RCCM. Il faudra poursuivre cette dynamique en proposant en 2017 ou en 2018 des ateliers ou un ANF autour de ces sujets.

Langages de programmation

Les actions autour de Python pour le calcul sont toujours très demandées. Trois ateliers sur Python, organisés par DevLog, sont en cours de réalisation auxquels le réseau Calcul participe activement. Ils sont néanmoins pour des débutants.

Le réseau Calcul a un très bon savoir faire sur les formations Python autour du calcul scientifique et en a maintenant organisé un certain nombre. Afin de ne pas faire double emploi avec l'initiative de DevLog, le réseau Calcul propose de faire de manière récurrente une formation de maîtrise des techniques pour avoir des codes de calcul performants et bien écrits en utilisant ce langage. Nous espérons en réaliser une en 2016. Une tous les deux ans semble un choix raisonnable.

Depuis un peu plus d'un an maintenant, nous organisons des formations d'initiation et de perfectionnement autour du langage C++. Le langage C++ a connu un regain d'intérêt depuis l'adoption du standard 2011 et de 2014. Ces évolutions du langage ont pour objectif de rendre plus sûr, plus performant et plus simple le développement d'applications critiques et de logiciels d'infrastructures. Le formateur qui réalise ces sessions est un expert du langage et est très actif dans les évolutions du langage pour les années à venir. Il nous semble important de maintenir jusqu'en 2018 un atelier d'initiation et un atelier de perfectionnement par an sur 3 jours.

Nous avons réalisé un atelier sur le langage Julia qui est un tout nouveau langage (3 ans) et est très prometteur. Néanmoins, la recherche de formateurs en France a été fastidieuse et il ne semble donc pas raisonnable de faire une ANF en 2016 sur le sujet comme proposé dans le précédent plan de formation. On peut supposer que d'ici 2017 voire 2018, ce langage sera assez mature et la communauté française assez développée pour organiser une ANF sur le sujet.