

MSDA / Vigrid : La grille de calcul EGI au service de l'analyse protéomique

Christine CARAPITO^{1a}, Jérôme PANSANEL^{1b}, Patrick GUTERL^{1a}, Alexandre BUREL^{1a}, Fabrice VARRIER^{1a}, Fabrice BERTILE^{1a}, Stéphane GENAUD², Alain VAN DORSSELAER^{1a}, Christelle ROY^{1b}

¹ Institut Pluridisciplinaire Hubert Curien (IPHC), DSA^a-DRS^b, CNRS UMR7178, Université de Strasbourg, Strasbourg, France

² Laboratoire de Sciences de l'Image, de l'Informatique et de la Télé-détection (LSIIT), équipe ICPS, CNRS UMR 7005, Université de Strasbourg, Strasbourg, France

Overview

In line with genomics, proteomics has rapidly emerged as a promising field of research enabling the elucidation of numerous life sciences questions [1]. The tremendous progresses in mass spectrometry-based techniques over the last 20 years have led to the possibility to generate massive, highly accurate, qualitative and quantitative mass spectrometry data for very complex protein mixtures. The main bottleneck in proteomics research today resides in the interpretation of such massive data and new informatics/bioinformatics resources are strongly needed to support the emergence of the field [2].

The proteomists and computer scientists of the Hubert Curien Pluridisciplinary Institute (IPHC, Strasbourg, France) have interfaced open-source proteomics data interpretation tools on the EGI (European Grid Infrastructure) grid. Hence, by allowing higher throughput and larger scale proteomics projects to be solved, our MSDA pipeline (Mass Spectrometry Data Analysis, <https://msda.unistra.fr/>) offers important time gain factors. It also opens new fields of applications that were previously unreachable due to a lack of computing resources. The originalities of the use of the EGI grid infrastructure for this pipeline are described in this article.

Enjeux scientifiques, besoin en calcul, stockage et visualisation

En parallèle de la génomique, la protéomique a émergé ces 20 dernières années comme un domaine de recherche essentiel pour élucider de nombreuses problématiques en Sciences de la Vie [1]. La spectrométrie de masse a joué un rôle fondamental dans le développement de l'analyse protéomique. Les progrès instrumentaux ont conduit au développement d'instruments générant des données de spectrométrie de masse de plus en plus volumineuses (du fait d'une grande rapidité d'acquisition des spectres de masse). Par ailleurs, la soumission des résultats d'identification de protéines à partir de ces données est de plus en plus réglementée par les journaux du domaine qui recommandent l'utilisation d'algorithmes « open-source » et multiples si possible. Pourtant, les outils informatiques/bioinformatiques et les ressources de calcul disponibles constituent aujourd'hui encore un verrou majeur dans le domaine de la protéomique [2].

Ainsi, afin de répondre au besoin croissant de puissance de calcul nécessaire à l'interprétation des données de protéomique, la suite logicielle MSDA, bâtie sur des logiciels libres a été adaptée et améliorée (Vigrid) sur la grille de calcul.

Développements, utilisation des infrastructures, outils, difficultés rencontrées

I. MSDA : Interface utilisateurs

Le portail d'accès MSDA est une application web qui permet aux utilisateurs/protéomistes de réaliser l'ensemble du workflow d'interprétation des données de spectrométrie de masse allant de la génération des banques de données de séquences à l'interprétation finale des spectres de masse.

Son développement répond aux besoins suivants :

- Offrir un portail d'accès simplifié vers les applications, accessible à partir de tous types de postes de travail.
- L'adaptation d'applications pour la grille nécessite de découper les données en entrée pour paralléliser les traitements en autant de jobs que possible, de façon à bénéficier au maximum des ressources offertes par la grille de calcul. L'interprétation des données de spectrométrie est adaptée à un tel découpage car ce sont des tâches indépendantes dont les temps d'exécution sont courts (quelques minutes). Par conséquent, il faut prendre en compte les temps d'attente lors de la soumission de ces jobs afin d'éviter qu'ils ne soient supérieurs à leur temps d'exécution.
- Permettre de transférer des fichiers volumineux (banques de données de séquences >100Mo et données de spectrométrie de masse à interpréter), ce qui est requis par les applications utilisées.
- Assurer l'aboutissement de l'intégralité des calculs car un unique spectre de masse peut correspondre à la protéine d'intérêt recherchée, ce qui a nécessité le développement d'outils de supervision des tâches de calcul.

L'accès à ce portail, sécurisé via un système d'authentification (identifiant/mot de passe), permet de soumettre et de superviser les différentes tâches soumises à des Computing Elements (CE) de la grille EGI. Chaque CE contrôle un ensemble de worker nodes (les unités de calcul) et est chargé de l'ordonnement des jobs sur ces nœuds.

II. Vigrid : Gestionnaire des requêtes de calcul par MSDA

Nous avons développé Vigrid, un nouvel outil permettant d'améliorer les soumissions de jobs faites à partir de MSDA. Cet outil se différencie de l'approche gLite [3] classique par deux aspects :

- Un stockage intermédiaire des données sur des serveurs dédiés au stockage de la grille EGI : Storage Elements (SE).
- Une stratégie de soumission optimisée en considérant les files d'attente individuellement.

Le logiciel Vigrid peut être utilisé dans n'importe quel environnement et n'est pas lié à la suite logicielle MSDA. C'est un outil autonome qui peut être utilisé pour toute application faisant appel aux ressources des grilles de calcul.

La figure 1 résume le schéma de l'utilisation de la grille EGI par le serveur MSDA et Vigrid. Les objets matérialisés en pointillés représentent les modules développés pour répondre aux besoins spécifiques de nos applications, en particulier la supervision.

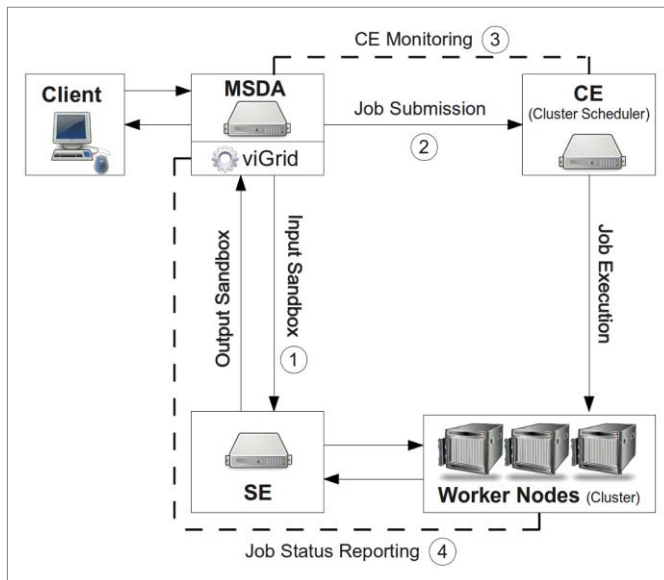


Figure 1 :

Utilisation de la grille EGI par la plate-forme applicative MSDA/Vigrid. Les procédures de surveillance intégrée sont mises en évidence en pointillés.

1/ Transfert sur un Storage Element (SE)

La politique de bon usage de la grille EGI recommande de limiter la taille de l'Input Sandbox, c'est-à-dire l'ensemble des fichiers téléchargés vers le CE. Par conséquent, nous avons opté pour un mécanisme utilisant un SE comme espace de stockage intermédiaire. Ce mécanisme a deux avantages :

- S'affranchir des limites imposées à la taille des Input Sandbox sur les CE ou les WMS (Workload Manager System).
- Limiter les transferts de fichiers volumineux entre les différents CE et le serveur MSDA.

A la fin de l'exécution des tâches, les fichiers transférés sur le SE sont automatiquement effacés.

2/ Job Submission / Gestion des temps de latence

L'utilisation classique de gLite conduit à soumettre les jobs à des WMS, qui sont des métaschedulers, qui accèdent ensuite aux CE. Cette utilisation engendre des temps d'attente (Status Waiting) qui peuvent atteindre jusqu'à 45-60 minutes, ce qui ne répond pas aux attentes des utilisateurs de ces applications spécifiques où des réponses rapides sont nécessaires pour la poursuite des expériences. L'alternative est de soumettre directement les jobs à des CE sélectionnés en évitant l'utilisation des WMS, en suivant une stratégie particulière de soumission.

Pour optimiser le temps de latence, le logiciel JJS (Java Job Submission [4]) avait été utilisé dans un premier temps puis Vigrid, basé sur la distribution JSAGA (Implémentation en Java de l'API *Simple API for Grid Applications* [5]), a été développé en interne. Cet outil permet de soumettre des collections de jobs sur la grille et de sélectionner des ressources pertinentes. Il peut être déployé depuis n'importe quel poste disposant d'un certificat permettant d'accéder à la VO, sans nécessiter de privilège administrateur ni interférer avec les logiciels d'exploitation installés sur les grilles de calcul.

Le ratio (temps d'attente) / (durée d'exécution du programme) devant être le plus faible possible, nous utilisons à la fois une liste des CE spécifiques, ainsi qu'un paramètre de soumission, *TimeOutWait*. Ce paramètre définit le temps maximal durant lequel un calcul peut rester en file d'attente. Si ce temps est dépassé sur un CE, le calcul est annulé sur ce CE, qui est alors retiré de la liste des ressources disponibles. Le calcul est alors resoumis sur un CE ayant plus de ressources disponibles. Il est à noter que, contrairement à JJS, Vigrid considère chaque file de manière distincte plutôt que le site en entier, ce qui

permet d'exclure une file de CE tout en conservant les autres files disponibles, le cas échéant, sur un CE donné.

La liste des CEs est construite par rapport aux ressources disponibles, associées à la VO d'appartenance (Biomed dans notre cas), publiées par l'annuaire (TopBDII). Nous y ajoutons le résultat de l'outil de supervision de Vigrd qui soumet périodiquement (par défaut toutes les heures) une commande pour évaluer le temps de réponse. Dans cet outil, trois états ont été définis :

- ^ OK : la commande a été exécutée dans un temps inférieur au TimeOutWait (par défaut 10min).
- ^ ERROR : le job a été rejeté par le CE.
- ^ TIMEOUT : la durée TimeOutWait a été dépassée.

Afin qu'un CE apparaisse dans la liste des ressources disponibles, il doit remplir les critères suivants :

- Les attributs publiés par le TopBDII pour le CE doivent correspondre à nos prérequis. Par exemple, l'attribut GlueCEMaxCPUTime doit être supérieur à 120 minutes.
- Le CE doit avoir obtenu un état « OK » lors de la dernière soumission.
- Le CE doit avoir un rapport du nombre de succès par rapport au nombre de soumissions durant les N derniers jours supérieur à X. Fixer la valeur X à 0.75 par défaut semble être optimal puisque, d'après l'historique des mesures, il apparaît qu'aucun site n'est disponible en permanence en raison d'aléas techniques.

L'outil de supervision de Vigrd permet donc de définir une liste ordonnée des files d'attente des CE les plus réactives. Les soumissions de jobs se font en priorité sur les files d'attente en haut de la liste. Les CE ou les files d'attente ne répondant pas ne seront pas dans cette liste tant que l'outil de supervision n'aura pas obtenu de réponse de leur part. Les jobs sont soumis directement aux différents CEs et pris en charge par le gestionnaire de batch de chaque CE de manière banalisée.

3/ CE Monitoring / Gestion des tâches

Pour répondre aux besoins de supervision et de gestion des tâches, Vigrd se base sur l'approche utilisée par JJS et permet de suivre l'état des jobs sur la base des états définis dans l'API (« start », « submitted », « waiting », « running », « end », « aborted », « cancelled »).

Les critères de gestion, schématisés dans la figure 2, sont les suivants :

- Si le job est en état « start », « submit » ou « waiting » lorsque la valeur TimeOutWait est atteinte, le job est annulé et relancé sur un autre CE de la liste.
- Si le job est en état « running », aucune action n'est menée.
- Si le job est en état « end », l'ensemble des fichiers de sortie est téléchargé.
- Si le job est en état « aborted » ou « cancelled », le job est relancé sur un autre CE et le CE non fonctionnel est retiré de la liste.

Il est ainsi possible de piloter de manière dynamique l'ensemble du processus de soumission des tâches de calcul.

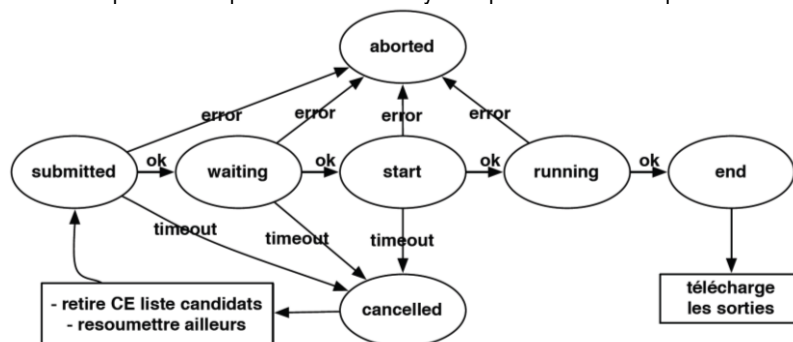


Figure 2 : Spécifications du système de gestion des tâches de Vigrd.

4/ Status Reporting

Après chaque commande du fichier d'exécution le code de retour doit être vérifié. En cas d'erreur, le programme est interrompu. Pour affiner la supervision des programmes, chaque fichier d'exécution (Executable Shell) soumis à un CE contient une fonction de reporting qui permet de renvoyer vers le portail les informations définies par le développeur. Ce mécanisme est basé sur un mode client/serveur ssh avec une sécurité renforcée sur un port non standard. Le serveur est configuré pour n'accepter aucune commande interactive ni de session. L'authentification est réalisée avec un couple clé privée/clé publique, la clé publique étant insérée dans le fichier d'exécution de commandes. Périodiquement, les informations de reporting sont transmises vers le portail via la commande scp. Grâce à ce mécanisme, nous avons pu mettre en évidence certains problèmes de configuration de CE et mieux comprendre le comportement des CEs.

Résultats scientifiques

La suite logicielle MSDA développée à l'IPHC permet :

- De créer, d'extraire, de concaténer, de formater des banques de données de séquences protéiques, notamment à partir des banques de séquences publiques accessibles (NCBIInr, UniProtKB, UniProtKB/SwissProt, ...).
- D'interpréter à haut débit des données de spectrométrie de masse par recherches en banques de données (OMSSA : Open Mass Spectrometry Search Algorithm [6]).
- D'interpréter à haut débit des données de spectrométrie de masse acquises sur des organismes non séquencés par séquençage *de novo* suivi de recherches d'homologies de séquences.
- D'extraire automatiquement l'ensemble des annotations fonctionnelles disponibles sur les protéines identifiées.

L'utilisation de la grille pour ces applications permet des gains de temps considérables par rapport à l'utilisation de serveurs locaux :

- **Un gain de temps net** : Après optimisation des paramètres de lancement sur la grille et un découpage optimal des données en fonction du type d'expérience mené, un set de données tests composé de 4 expériences typiques d'analyses protéomiques par spectrométrie de masse a été utilisé pour évaluer les gains de temps atteints (Tableau 1). Par exemple, sans la grille, l'expérience 4 n'était que difficilement envisageable car elle nécessite la mobilisation du serveur du laboratoire durant 3 jours complets.

Type d'expérience de spectrométrie de masse	Lancement sur un serveur local (h)	Lancement sur la grille (h)	Facteur de gain de temps
1.Spectrométrie de masse à haute résolution et spécificité enzymatique complète (65 fichiers d'entrée: 62Mo)	0,3	0,2	1,5
2.Spectrométrie de masse à haute résolution et semi-spécificité enzymatique (65 fichiers d'entrée: 62Mo)	5,8	0,5	12
3.Spectrométrie de masse à basse résolution et spécificité enzymatique complète (223 fichiers d'entrée: 551 Mo)	3,6	0,7	5
4.Spectrométrie de masse à basse résolution et semi-spécificité enzymatique (223 fichiers d'entrée: 551 Mo)	74,6	0,95	79

Tableau 1 : Facteurs de gains de temps obtenus pour le lancement de 4 types d'expériences d'interprétations de données de spectrométrie de masse sur un serveur local (4 CPUs / 8Go RAM / Scientific Linux 5.4 64bits) et sur la grille de calcul (GlueCEPolicyMaxCPUTime et GlueCEPolicyMaxWallTime:60min / 250 CPUs minimum / GlueCEStateStatus: Production). Ces valeurs sont des moyennes (environ 100 valeurs de temps mesurées par type d'expérience) obtenues suite au lancement régulier de ces recherches sur une période de 3 mois.

- **Un gain de temps global** : Les algorithmes d'interprétation des données de spectrométrie de masse sont, en général, lancés sur un serveur dédié dans les laboratoires de protéomique. Ce système de ressource limitée impose une hiérarchisation des priorités des recherches. Avec la grille, l'ensemble des utilisateurs peut lancer ses requêtes simultanément ce qui entraîne un gain de temps global important, directement croissant et proportionnel au nombre d'utilisateurs simultanés.

Cette suite logicielle a été présentée à la communauté de protéomique lors du congrès annuel de la Société Française de Spectrométrie de Masse et d'Analyse Protéomique (SMAP, Avignon, Septembre 2011). L'application compte aujourd'hui environ 31 utilisateurs internes et 43 utilisateurs externes à l'IPHC.

Conclusion & Perspectives

Le logiciel Vigrid peut être intégré dans toute application souhaitant accéder à la grille de calcul et de bénéficier des spécifications suivantes :

- Une gestion fine de l'accès aux différentes files des CEs.
- Une gestion permettant de différencier les états des jobs (Queuing/Waiting/Running/Abort) et de gérer dynamiquement ces états dans l'application.

Ce logiciel, associé à la librairie JSAGA, peut être déployé sur des ordinateurs dont le système d'exploitation permet d'exécuter des programmes en java, sans disposer de privilèges administrateur.

Par ailleurs, l'architecture de la suite logicielle MSDA est en cours d'évolution. Un mécanisme de Web-service basé sur WSDL est en cours d'implémentation et permettra de séparer la partie « Modèle » sur un serveur distinct et de diversifier l'accès aux ressources. Pour la partie applicative, les développements se poursuivent également avec l'implémentation de nouveaux modules (e.g. séquençage *de novo*, outils de quantification) dans MSDA. L'utilisation de la grille a permis de répondre aux importants besoins de puissance de calcul non accessibles à ce jour dans les laboratoires de protéomique et ce type de développement contribuera certainement à briser le verrou majeur que constitue le manque de ressources informatiques / bioinformatiques dans le domaine de la protéomique.

Références

- [1] Bensimon A, *et al.* Annu Rev Biochem. 2012, 81:379-405.
- [2] Aebersold R. Nat Methods. 2009, 6(6):411-412.
- [3] gLite - Lightweight Middleware for Grid Computing (<http://glite.cern.ch>)
- [4] JJS : Java Jobs Submission P.Calvat CCIN2P3 (<http://cc.in2p3.fr/docenligne/269>)
- [5] JSAGA : A Java implementation developed at CCIN2P3 (<http://grid.in2p3.fr/jsaga/>);
- [6] Geer LY, *et al.* J Proteome Res. 2004, 3(5):958-964.