

VIP et GateLab : retour d'expérience

Sorina Camarasu-Pop

Université de Lyon, CREATIS ; CNRS UMR5220 ; Inserm U1044 ; INSA-Lyon ; Université Lyon 1, France.
CREATIS - INSA LYON
7 Avenue Jean Capelle
69621 Villeurbanne Cedex

Rafael Ferreira da Silva

Université de Lyon, CREATIS ; CNRS UMR5220 ; Inserm U1044 ; INSA-Lyon ; Université Lyon 1, France.
CREATIS - INSA LYON
7 Avenue Jean Capelle
69621 Villeurbanne Cedex

Tristan Glatard

Université de Lyon, CREATIS ; CNRS UMR5220 ; Inserm U1044 ; INSA-Lyon ; Université Lyon 1, France.
CREATIS - INSA LYON
7 Avenue Jean Capelle
69621 Villeurbanne Cedex

Résumé

Le portail VIP/Gate-Lab¹ compte plus de 450 utilisateurs enregistrés et donne accès à une dizaine d'applications qui s'exécutent de manière transparente sur les ressources de calcul distribué de l'organisation virtuelle (VO) biomed dans EGI (European Grid Infrastructure). A travers ce portail web, les utilisateurs ont la possibilité de s'authentifier, lancer l'exécution d'une application, la suivre ou bien accéder à l'historique et aux résultats des exécutions précédentes. Le transfert de données vers et depuis les éléments de stockage est géré aussi par la plate-forme. De nouvelles fonctionnalités, telles que des catalogues de modèles et des ontologies, ont été récemment ajoutées pour faciliter l'intégration et l'échange de modèles et de nouvelles applications.

Démarré en 2010 dans le cadre de l'ANR VIP, le portail est devenu aujourd'hui un outil quotidien pour ses utilisateurs qui y font tourner des simulateurs et des algorithmes de traitement d'images médicales. A titre d'exemple, cette année, en sept mois (de janvier à juillet 2013), plus de 1700 exécutions réussies ont été lancées par 70 utilisateurs différents. Le temps CPU moyen consommé chaque mois est de 19 années-CPU. D'après les statistiques publiées par la grille de calcul européenne EGI, le portail VIP est l'un des plus utilisés pour accéder à cette infrastructure.

Pour proposer ses services de bout en bout, la plateforme VIP/Gate-Lab s'appuie sur des ressources comme le moteur de workflows MOTEUR et le système de jobs-pilotes Dirac. Elle utilise d'ailleurs l'instance nationale du service Dirac mise en production pour plusieurs communautés scientifiques. Déployée au CC-IN2P3, cette instance nationale est administrée à tour de rôle par des experts localisés dans plusieurs laboratoires. Cette organisation permet de répartir la charge de travail sur plusieurs sites et d'obtenir un meilleur résultat en termes de support aux utilisateurs.

Mots clefs

calcul distribué, portail web, imagerie médicale, services de bout en bout

1. <http://vip.creatis.insa-lyon.fr>

1 Introduction

Les images médicales peuvent être simulées à partir de modèles numériques en utilisant différentes modalités, parmi lesquelles l'imagerie par résonance magnétique, la tomographie par émission de positons, l'imagerie ultrasonore et la tomodensitométrie. La simulation d'images reste cependant difficile à maîtriser, en particulier du fait de la complexité et de la lourdeur des processus de simulation. Les codes de simulations comme les modèles physiques des objets à imager sont complexes et spécifiques à la modalité d'imagerie. De plus, les temps de calcul importants limitent aussi le réalisme et la taille des scènes de simulation. L'utilisation d'infrastructures de calcul distribué est possible mais est une contrainte technique supplémentaire pour les utilisateurs de simulation. La plate-forme VIP (Virtual Imaging Platform), développée dans le cadre du projet ANR VIP, facilite l'accès aux modèles multi-physiques d'organes et structures du vivant, aux simulateurs, et aux moyens de calcul et de stockage nécessaires à la simulation d'images médicales.

2 Portail VIP

2.1 Architecture

La plate-forme VIP donne accès en ligne à des simulateurs d'imagerie médicale. Les simulateurs sont décrits comme des flux de travail (workflows) et exécutés avec MOTEUR [1] en utilisant des tâches pilotes DIRAC [2].

L'architecture globale de la plate-forme est schématisée sur la figure 1. Le portail web a été développé en Java en utilisant le Google Web Toolkit (GWT).

L'utilisation des flux de travail fournit une représentation formelle accessible et de haut niveau utilisée pour déterminer les sous-processus qui peuvent être exécutés simultanément sur des plates-formes distribuées. Les flux de travail sont

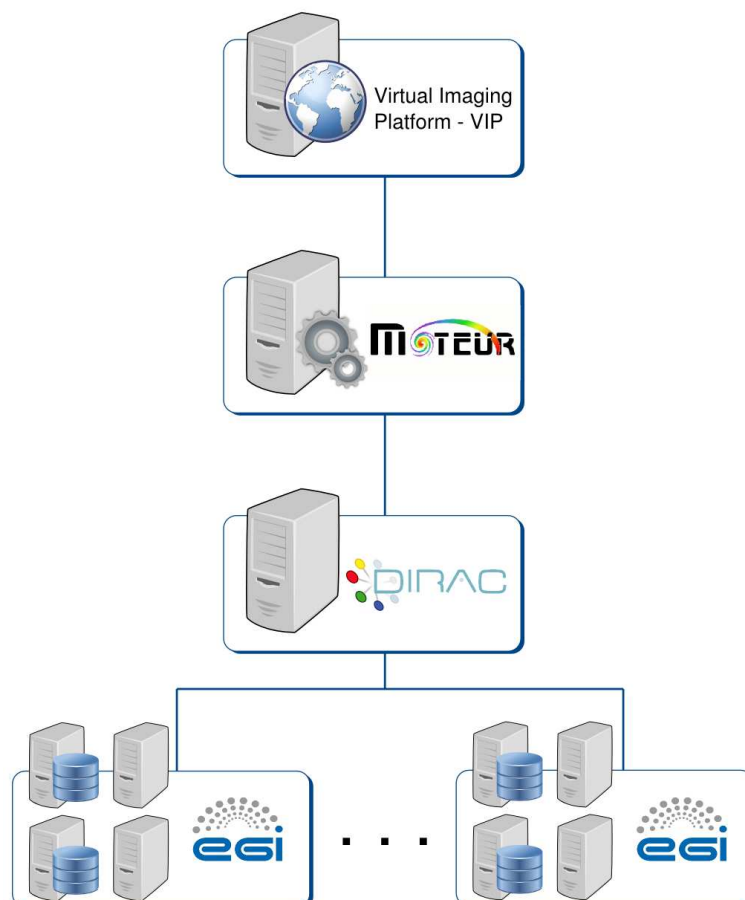


Figure 1 - Architecture de la plate-forme VIP.

décrits en utilisant GWENDIA, un langage dédié à la description des applications scientifiques traitant de large volumes de données. Les flux de simulation VIP sont décrits en détail dans [3].

Le système de gestion de tâches utilise des tâches pilotes gérées par le service DIRAC. VIP utilise d'ailleurs l'instance nationale du service Dirac mise en production pour plusieurs communautés scientifiques. Déployée au CC-IN2P3, cette instance nationale est administrée à tour de rôle par des experts localisés dans plusieurs laboratoires. Cette organisation permet de répartir la charge de travail sur plusieurs sites et d'obtenir un meilleur résultat en termes de support aux utilisateurs. Les tâches sont soumises sur les ressources de l'organisation virtuelle (VO) biomed de l'infrastructure de grille EGI.

2.2 Fonctionnalités

2.2.1 Lancement, exécution et suivi des calculs distribués

VIP offre une interface web haut niveau, qui masque complètement les ressources de calcul et de stockage sous-jacentes, ainsi que les méthodes utilisées pour les exploiter efficacement.

Environ dix applications différentes sont actuellement disponibles dans VIP pour la simulation d'images médicales en radio-thérapie, tomographie par émission de positons (TEP), imagerie par résonance magnétique (IRM) et ultrasons (US), ainsi que pour le traitement d'images médicales (FSL et Freesurfer).

L'intégration des simulateurs se fait sans modification de leur code, principalement par parallélisme de données. L'exécution des applications est ainsi itérée sur un ensemble de données en entrée et les résultats partiels sont ensuite fusionnés par un autre processus. Ce type de parallélisation est implicite au langage de description du flux de travail et, par conséquent, transparent pour l'utilisateur final.

Les utilisateurs finaux n'ont qu'à fournir l'ensemble de données en entrée pour lancer l'exécution distribuée des applications. Ils peuvent ensuite suivre l'évolution des calculs et télécharger le résultat à la fin. Des statistiques et des traces d'exécution sont aussi disponibles à travers l'interface web. L'ensemble des informations concernant l'exécution des tâches (temps d'attente et d'exécution, taux d'erreur, etc.) est géré à travers une base de données. Les traces d'exécution de l'application elle-même sont disponibles sous la forme des fichiers texte (un fichier de sortie et un fichier d'erreur par tâche).

2.2.2 Gestion de données

Le portail web permet aussi le transfert de fichiers entre le PC de l'utilisateur et les éléments de stockage d'EGI. Pour des raisons d'authentification et d'autorisation, le transfert de fichiers se fait en deux étapes :

- entre le PC de l'utilisateur et le serveur VIP ;
- entre le serveur VIP et les éléments de stockage d'EGI.

Chaque utilisateur dispose d'un dossier personnel sur EGI et a accès aux dossiers des groupes d'utilisateurs auxquels il appartient. VIP permet donc la gestion d'espaces privés et partagés.

2.2.3 Ontologies, catalogues de données et de modèles

L'ontologie OntoVIP [4] modélise les concepts et relations impliqués dans les modèles d'objets utilisés par les simulateurs, dans la description des processus de simulations et dans les données simulées. Cette ontologie permet de décrire uniformément les modèles d'objets, les simulateurs et leurs résultats, dans l'objectif de favoriser le partage de modèles entre simulations, le partage de données simulées et la réutilisation de composants de simulation.

VIP propose également un catalogue de modèles multi-physiques, et un catalogue de données simulées annotées automatiquement pendant l'exécution des workflows. Les modèles contiennent des informations non seulement sur l'anatomie, mais aussi sur les pathologies et les entités externes présentes dans le corps lors de l'acquisition de l'image. Les modèles peuvent être chargés dans VIP sous la forme de fichiers de données enrichies avec des annotations OntoVIP. Des règles sont appliquées pour vérifier si un modèle peut être utilisé dans une simulation d'une modalité particulière, c'est à dire, si tous les paramètres physiques requis sont définis.

Nombre d'exécutions réussies

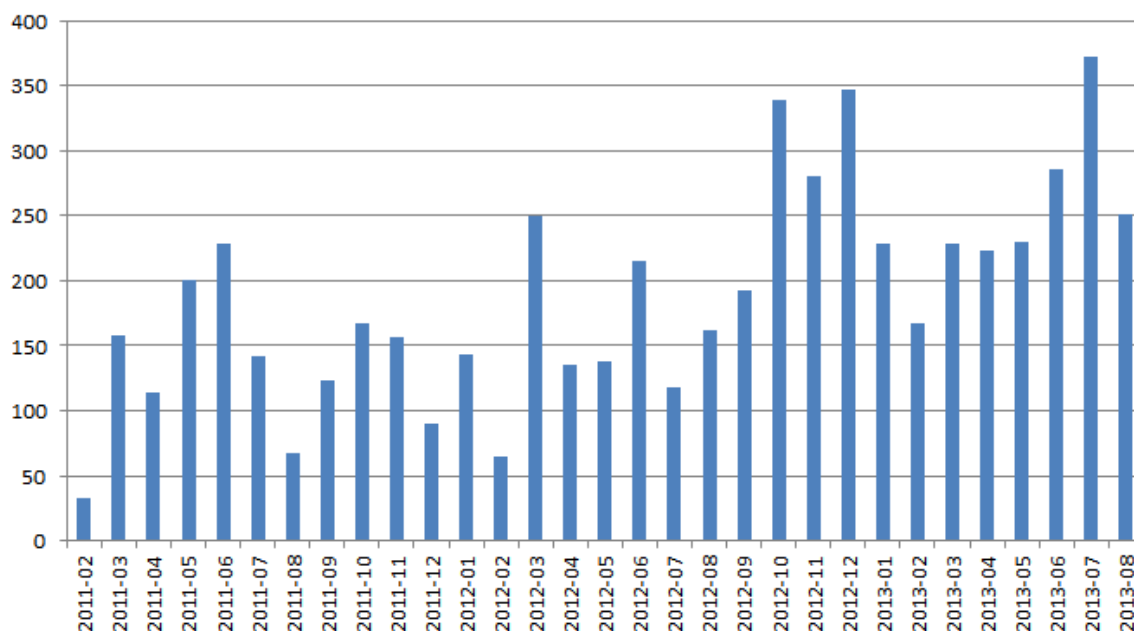


Figure 2 - Nombre d'exécutions réussies chaque mois depuis février 2011. L'activité de la plate-forme est stable et continue à croître.

2.3 Statistiques d'utilisation et résultats

D'après les statistiques publiées par la grille de calcul européenne EGI, le portail VIP est l'un des plus utilisés pour accéder à cette infrastructure. De janvier à juillet 2013, plus de 1700 exécutions réussies ont été lancées par 70 utilisateurs différents. La figure 2 montre le nombre d'exécutions réussies chaque mois depuis février 2011. On peut remarquer que l'activité de la plate-forme est stable et continue à croître. Le temps CPU consommé moyen est de 19 années-CPU chaque mois. La figure 3 présente la répartition des utilisateurs enregistrés dans VIP en juillet 2013.

La figure 4, extraite de [5], montre une simulation Monte-Carlo effectuée avec GATE [6]. Il s'agit de l'irradiation d'un cerveau de souris C57Bl6 en utilisant le modèle Xrad225Cx avec 225kV et 13mA. Cette simulation correspond à un temps CPU total de quelques mois-CPU et a été exécutée avec la plate-forme VIP/GATE-Lab en un jour.

3 Optimisations

Le succès de VIP est dû, d'un côté, au travail effectué au niveau de l'interface homme-machine et à sa facilité d'utilisation et, de l'autre, aux optimisations mises en place pour garantir de bonnes performances. Plus particulièrement, des stratégies d'équilibrage de charge et d'auto-administration ont été développées pour améliorer la qualité de service fournie par la plate-forme.

3.1 Equilibrage de charge

Les ressources à disposition de la VO biomed, qui comporte plus d'une centaine de sites de calculs, sont fortement hétérogènes. Cette hétérogénéité matérielle et logicielle, ainsi que l'absence d'une qualité de service (QoS) assurée, complexifient significativement l'ordonnancement des tâches. Typiquement, le makespan (le temps d'exécution d'une application perçu par l'utilisateur) est déterminé par la date de fin de la dernière tâche, qui peut pénaliser considérablement les performances si elle est exécutée sur une machine plus lente que les autres ou si elle doit être re-soumise suite à un échec. Pour remédier à ce problème, nous avons développé des méthodes d'équilibrage de charge particulièrement adaptées aux applications Monte-Carlo s'exécutant sur des systèmes distribués à très grande échelle [7].

Utilisateurs VIP

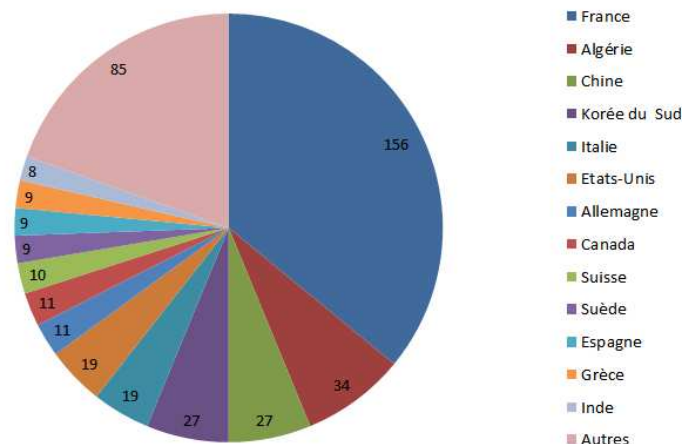


Figure 3 - Répartition des utilisateurs enregistrés en juillet 2013.

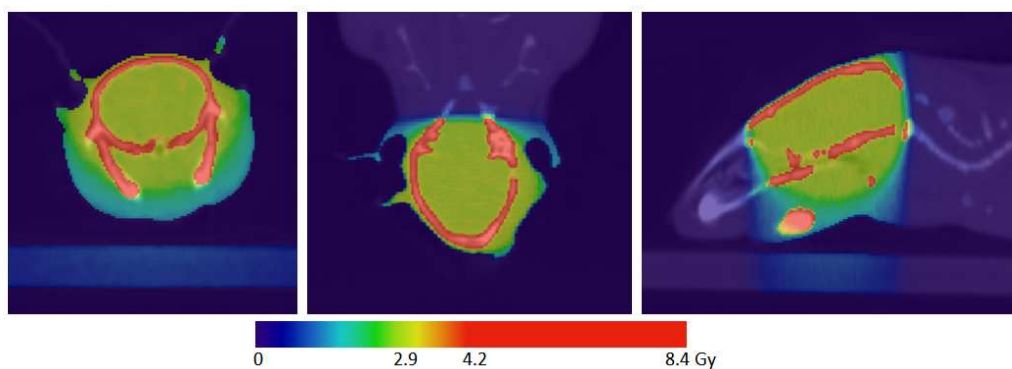


Figure 4 - Irradiation d'un cerveau de souris : simulation Monte-Carlo effectuée avec GATE sur une souris C57Bl6 en utilisant le modèle Xrad225Cx avec 225kV et 13mA. Cette simulation correspond à un temps CPU total de quelques mois-CPU et a été exécutée avec la plate-forme VIP/GATE-Lab en un jour. Figure extraite de [5].

L'équilibrage de charge dynamique proposé consiste en une boucle "tant que" sans découpage initial. Chaque tâche d'une simulation est créée avec le nombre total d'événements et continue à s'exécuter jusqu'à ce que le nombre souhaité d'événements soit atteint avec la contribution de toutes les tâches. Par conséquent, chaque tâche peut simuler l'ensemble de la simulation si les autres tâches échouent ou ne démarrent pas. Le nombre total d'événements simulés est obtenu par la somme de tous les événements simulés par des tâches indépendantes. Ainsi, chaque ressource contribue à l'ensemble de la simulation jusqu'à ce qu'elle se termine.

Algorithm 1 Algorithme du maître pour l'équilibrage de charge dynamique des simulations Monte-Carlo

```
N=nombre total d'événements à simuler
n=0
Tant que n<N faire
    n = nombre d'événements simulés par les tâches en exécution ou terminées avec succès
Fin tant que
Envoyer signal d'arrêt à toutes les tâches
Annuler les tâches en attente
```

Algorithm 2 Algorithme des pilotes pour l'équilibrage de charge dynamique des simulations Monte-Carlo

```
Télécharger les données en entrée
N=nombre total d'événements à simuler
n=0, dernièreMiseàJour=0, delaiMiseàJour=5min
Tant que signal d'arrêt non reçu ET n<N faire
    Simuler l'événement suivant
    n++
    Si (getTime() - dernièreMiseàJour) >delaiMiseàJour alors
        Envoyer n au maître
        dernièreMiseàJour = getTime()
    Fin si
Fin tant que
Charger le résultat
```

Les algorithmes 1 et 2 présentent le pseudo-code du maître et des pilotes. Le maître somme régulièrement le nombre d'événements simulés et envoie des signaux d'arrêt aux pilotes en cas de besoin. Chaque pilote exécute une seule tâche, en commençant dès que le pilote arrive sur un noeud de grille et s'arrêtant à la fin de la simulation. Des communications très ponctuelles entre les tâches et le maître sont nécessaires pour mettre à jour le nombre actuel d'évènements calculés, ainsi qu'à la fin de la simulation lorsque le maître envoie des signaux d'arrêt.

La figure 5 montre le flot d'exécution d'une simulation GATE exécutée sur EGI en utilisant l'algorithme d'équilibrage de charge dynamique avec 500 tâches de calcul. Cette simulation correspond à un temps total de calcul de 282 jours-CPU et à été exécutée en utilisant la plate-forme VIP/GATE-Lab en 17,3 heures. Avec un taux d'erreur de 25% (dû principalement aux problèmes de transfert de fichiers), cette exécution a connu une accélération² remarquable de 392.

3.2 Réplication automatique

La mise à disposition de plates-formes masquant l'exploitation des ressources de calcul distribuées nécessite de gérer complètement les pannes et autres incidents pouvant survenir lors de l'exécution d'applications. Cette gestion peut s'effectuer soit manuellement, au prix d'un effort d'administration important, soit automatiquement, si des stratégies adaptées sont disponibles. Nous avons développé de telles stratégies [8, 9] qui permettent notamment de répliquer les tâches critiques des applications, de réagir aux pannes les plus courantes, et d'améliorer l'équité entre les utilisateurs. Ces stratégies s'appuient sur des boucles de contrôle exploitant les traces d'exécution de la plate-forme [10]. L'algorithme 3 présente le pseudo-code du mécanisme de réplication automatique.

2. L'accélération est définie ici comme étant le temps CPU total des tâches de calcul divisé par le temps d'exécution perçu par l'utilisateur.

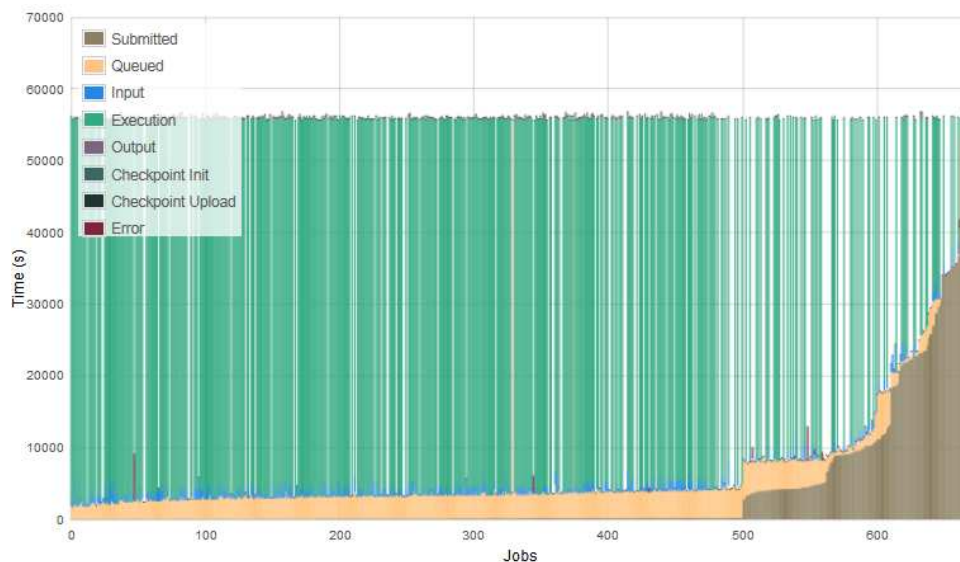


Figure 5 - Flot d'exécution d'une simulation GATE exécutée sur EGI en utilisant l'algorithme d'équilibrage de charge dynamique avec 500 tâches de calcul. Cette exécution a une accélération remarquable de 392, malgré un taux d'erreur de 25% compensé automatiquement grâce à l'approche d'équilibrage de charge dynamique proposée. Les barres vides correspondent aux tâches pour lesquelles nous manquons d'information sur leur temps d'exécution (cela peut arriver dans le cas des tâches en erreur ou annulées).

Algorithm 3 Algorithme de réplication automatique

```

Si La tâche est en retard alors
  Si Tous les réplicas de la tâche sont en retard alors
    Si Aucun réplica est en attente alors
      Répliquer la tâche
    Fin si
  Fin si
Fin si
Si Un réplica de la tâche est en retard alors
  Annuler le réplica
Fin si

```

4 Discussions et conclusion

Le portail VIP a connu un réel succès grâce à sa facilité d'utilisation et aux optimisations mises en place pour améliorer la qualité des services fournis. VIP est devenu aujourd'hui un outil quotidien pour ses utilisateurs qui y font tourner des simulateurs et des algorithmes de traitement d'images médicales. Cependant, il y a encore des limites et des choses à améliorer.

L'intégration de simulateurs dans la plate-forme d'exécution est grandement facilitée par leur description sous forme de workflows. Néanmoins, le développement de workflows reste une activité nécessitant une expertise certaine, notamment à cause de la nécessité de tenir compte du contexte d'exécution des applications sur des ressources distribuées. En pratique, l'effort nécessaire au développement de workflows reste un facteur limitant l'extension de la plate-forme VIP car il demande l'intervention d'un développeur.

Les simulations très courtes (moins d'une vingtaine de minutes de temps CPU), très longues (plus d'un an de temps CPU), manipulant de gros fichiers (1 Go ou plus) ou nécessitant beaucoup de mémoire (plus de 2 Go) restent problématiques. Ces problèmes sont d'autant plus difficiles à résoudre que l'activité des utilisateurs de la plate-forme n'est pas prévisible et que les caractéristiques techniques des simulations (durée, volume de données manipulées) sont en général inconnues avant l'exécution. L'utilisation de serveurs de calcul locaux et la personnalisation des workflows permettent de réduire l'impact de ces difficultés, mais elles restent présentes.

Concernant les outils sémantiques, leur exploitation reste à ce jour limitée à certains simulateurs.

Bibliographie

- [1] Tristan Glatard, Johan Montagnat, Diane Lingrand, et Xavier Pennec. Flexible and efficient workflow deployment of data-intensive applications on grids with MOTEUR. *International Journal of High Performance Computing Applications (IJHPCA)*, 22(3) :347–360, 2008.
- [2] A Tsaregorodtsev, N Brook, A Casajus Ramo, Ph Charpentier, J Closier, G Cowan, R Graciani Diaz, E Lanciotti, Z Mathe, R Nandakumar, S Paterson, V Romanovsky, R Santinelli, M Sapunov, A C Smith, M Seco Miguelez, et A Zhelezov. DIRAC3 . The New Generation of the LHCb Grid Software. *Journal of Physics : Conference Series*, 219 062029(6), 2009.
- [3] A Marion, G Forestier, H Benoit-Cattin, S Camarasu-Pop, P Clarysse, R Ferreira da Silva, B Gibaud, T Glatard, P Hugonnard, C Lartizien, H Liebgott, J Tabary, S Valette, et D Friboulet. Multi-modality medical image simulation of biological models with the Virtual Imaging Platform (VIP). Dans *IEEE CBMS 2011*, Bristol, UK, 2011.
- [4] B. Gibaud, G. Forestier, H. Benoit-Cattin, F. Cervenansky, P. Clarysse, D. Friboulet, A. Gaignard, P. Hugonnard, C. Lartizien, H. Liebgott, J. Montagnat, J. Tabary, et T. Glatard. Ontovip : An ontology for the annotation of object models used for medical image simulation. Dans *Healthcare Informatics, Imaging and Systems Biology (HISB), 2012 IEEE Second International Conference on*, pages 110–110, 2012.
- [5] S. Supiot J. Suhard F. Paris A. Lisbona C. Noblet, S. Chiavassa et G. Delpon. Simulation of the xrad225cx preclinical irradiator using gate/geant4. Dans *Workshop Radiobiology applied to Oncology*, April 2013.
- [6] J Allison, K Amako, J Apostolakis, H Araujo, P A Dubois, M Asai, G Barrand, R Capra, S Chauvie, R Chytraccek, G A P Cirrone, G Cooperman, G Cosmo, G Cuttone, G G Daquino, M Donszelmann, M Dressel, G Folger, F Foppiano, J Generowicz, V Grichine, S Guatelli, P Gumplinger, A Heikkinen, I Hrivnacova, A Howard, S Incerti, V Ivanchenko, T Johnson, F Jones, T Koi, R Kokoulin, M Kossov, H Kurashige, V Lara, S Larsson, F Lei, O Link, F Longo, M Maire, A Mantero, B Mascialino, I McLaren, P M Lorenzo, K Minamimoto, K Murakami, P Nieminen, L Pandola, S Parlati, L Peralta, J Perl, A Pfeiffer, M G Pia, A Ribon, P Rodrigues, G Russo, S Sadi-lov, G Santin, T Sasaki, D Smith, N Starkov, S Tanaka, E Tcherniaev, B Tome, A Trindade, P Truscott, L Urban, M Verderi, A Walkden, J P Wellisch, D C Williams, D Wright, et H Yoshida. {G}eant4 {D}evelopments and {A}pplications. *{IEEE} {T}ransactions on {N}uclear {S}cience*, 53 :270–278, 2006.
- [7] S Camarasu-Pop, T Glatard, J T Mościcki, H Benoit-Cattin, et D Sarrut. Dynamic partitioning of {GATE} {M}onte-{C}arlo simulations on {EGEE}. *Journal of Grid Computing*, 8(2) :241–259, Mars 2010.
- [8] R da Silva, T Glatard, et Frédéric Desprez. Self-healing of workflow activity incidents on distributed computing infrastructures. *Future Generation Computer Systems*, 2013.
- [9] R. Ferreira da Silva, T. Glatard, et Frédéric Desprez. Self-healing of operational workflow incidents on distributed computing infrastructures. Dans *12th IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing - CCGrid 2012*, pages 318–325, Ottawa, Canada, 05/2012 2012.
- [10] R da Silva et T Glatard. A Science-Gateway Workload Archive to Study Pilot Jobs, User Activity, Bag of Tasks, Task Sub-Steps, and Workflow Executions. Dans *CoreGRID/ERCIM Workshop on Grids, Clouds and P2P Computing*, Rhodes, GR, 2012.