

DIET Batch management system v.1.1

Daouda Traoré **Gael Le Mahec** **David Loureiro**
daouda.traore@ens-lyon.fr gael.le.mahec@ens-lyon.fr david.loureiro@ens-lyon.fr
Eddy Caron **Frédéric Desprez**
eddy.caron@ens-lyon.fr frederic.desprez@ens-lyon.fr

19 octobre 2010

Table des matières

1	Objectif	2
2	Présentation de l'ensemble logiciel "DIET batch management"	2
2.1	Implantation des différentes commandes	3
2.2	Pré-requis	3
2.3	Scénarios d'utilisation	4
2.3.1	Soumission d'un job	4
2.3.2	Affichage de la liste des jobs	4
2.3.3	Annulation d'un job	4
3	Logiciels et documents fournis	5

Version du document : 1.1

Ce document détaille les objectifs et la mise en œuvre des différents logiciels composant l'ensemble logiciels *DIET batch management system*. La première section présente les objectifs tels qu'ils ont été définis avec le destinataire de cet ensemble logiciel. La seconde section présente la mise en œuvre effective des différents logiciels nécessaires à la bonne réalisation des objectifs définis dans la première section. La troisième et dernière section donne la liste des codes et documents fournis.

1 Objectif

Le service de gestion des batchs par l'intermédiaire de l'intergiciel DIET présenté dans ce document a pour objectifs :

- De permettre à partir d'une machine distante de gérer les jobs sur les machines supportées à savoir de :
 - Soumettre un nouveau job
 - Afficher la liste des jobs en cours d'exécution ou en attente d'exécution
 - Supprimer des jobs soumis
 - Afficher le nombre de jobs inactifs ou en cours d'exécution

2 Présentation de l'ensemble logiciel “DIET batch management”

L'intergiciel DIET développé par l'équipe GRAAL de l'INRIA Rhône-Alpes a été utilisé pour construire le logiciel *batch management system* décrit dans ce document. Cet intergiciel est basé sur le paradigme client/serveur et d'appels de procédures distantes. Il donne la possibilité de définir des politiques d'ordonnancement avancées et permet l'utilisation d'un gestionnaire de données permettant la réplication et l'accès transparent aux données. Comme dans le document *file management system*, le document présente aussi DIET par ces trois éléments principaux qui sont :

- Un client : il s'agit d'une application qui utilise DIET pour résoudre des problèmes. Plusieurs clients pouvant se connecter à DIET, ces clients peuvent être des applications web, des applications PSE (Problem Solving Environment) comme Matlab ou Scilab, ou un simple programme compilé.
- Un “Server Daemon” (SeD) : c'est un serveur qui fournit les services permettant de résoudre les problèmes des clients. Un SeD encapsule un serveur de calcul. Il peut être par exemple placé sur le point d'entrée d'une machine parallèle (super-calculateur, cluster etc.) ou simplement sur une machine individuelle du réseau. Un SeD stocke la liste des services qu'il offre aux utilisateurs, mais aussi des informations sur sa charge et ses capacités matérielles (nombre de processeurs, charges des processeurs, quantité de mémoire installée etc.).
- Un agent : c'est un élément de DIET qui reçoit les requêtes des clients, et qui effectue la sélection du ou des serveurs utilisés pour l'exécution d'un service. Les agents sont interconnectés suivant une topologie d'arbre dont le sommet est un agent “maître” (Master Agent ou MA) qui sera le seul pouvant être contacté par les applications clientes à la recherche d'un service.

L'ensemble logiciel de *DIET Batch management* présenté dans ce document se décompose en différentes catégories :

- Les logiciels *serveur* qui permettent la fourniture ou l'exécution des différents services de batch de la machine sur laquelle ils sont exécutés.
- Les logiciels *client* qui permettent l'appel de manière transparente des différents ser-

vices proposés par le *batch management system* en choisissant la machine concernée par l'opération à effectuer.

- Un ensemble d'utilitaires permettant l'installation et la configuration des différents éléments de l'ensemble logiciel. Il s'agit des fichiers de configuration DIET et des fichiers permettant la compilation de l'ensemble logiciel (*les makefile*).

2.1 Implantation des différentes commandes

L'ensemble logiciel *DIET batch management system* permet la réalisation générique et distante des opérations batch courantes. Ces opérations sont présentées dans le tableau 1.

Commande Loadleveler	Commande Torque	Rôle de la commande	Commande générique DIET
<code>lsubmit</code>	<code>qsub</code>	permet de soumettre un job à partir d'un script de soumission	<code>diet-submit</code>
<code>llq</code>	<code>qstat</code>	permet d'afficher la liste des jobs en cours ou en attente d'exécution	<code>diet-list</code>
<code>lcancel</code>	<code>qdel</code>	permet de tuer un job à partir de son identification	<code>diet-cancel</code>

TABLE 1 – Commandes fournies par l'ensemble logiciel *DIET batch management*

A ces trois commandes s'ajoutent trois commandes qui permettent d'obtenir le nombre de jobs en cours d'exécution, en attente dans la queue, et le nombre total de jobs. Ces commandes sont disponibles via des options de `diet-list` : `--nbrun` pour le nombre de jobs en état *running*, `--nbwait` pour le nombre de jobs en état *waiting* et `--nball` pour le nombre de jobs total dans la queue.

2.2 Pré-requis

Comme dans le document *file management system*, les pré-requis identifiés pour le bon fonctionnement de ces opérations sont les suivants :

- L'installation sur les machines "hôtes" d'un *server Daemon* DIET développé à cet usage. Il s'agit ici du logiciel serveur DIET *diet-bmd batch management daemon*. L'utilisation de ce logiciel requiert :
 - La disponibilité d'une installation de DIET fonctionnelle sur la machine hôte. Ceci comprend la présence des bibliothèques nécessaires à sa bonne exécution et la présence sur le réseau accessible à la machine hôte d'un bus CORBA géré par l'ORB *omniORB* ainsi que le service de nommage CORBA *omniNames*.
 - Des paramètres et fichiers de configuration fixés à cet effet.
- La possibilité d'effectuer les appels aux logiciels *client* avec les paramètres adéquats dépendant du type d'opération à effectuer. Ces appels pourront être effectués en ligne de commande *shell* traditionnelle ou par l'intermédiaire d'une interface utilisateur permettant leur exécution avec des paramètres requis.

2.3 Scénarios d'utilisation

Cette section présente différents scénarios d'utilisation des commandes de l'ensemble logiciel DIET *batch management system*.

2.3.1 Soumission d'un job

Le client souhaite soumettre un job sur une machine distante. Pour cela il utilise la commande **diet-submit** en lui passant en paramètre le script de soumission.

L'opération se déroule en plusieurs étapes transparentes pour l'utilisateur :

1. Le client entre la commande de soumission en précisant la machine concernée.
`$ diet-submit -config <fichier de configuration DIET> <script> -host <hote>.`
2. L'application cliente contacte le master agent (MA) afin d'obtenir une référence au SeD de l'hôte concernée.
3. Le logiciel client transmet le script à soumettre au serveur.
4. Le serveur soumet le script localement et transmet les erreurs éventuelles au client.

2.3.2 Affichage de la liste des jobs

Le client souhaite afficher la liste des jobs sur une machine distante. Pour cela il utilise la commande **diet-list** en lui passant en paramètre le nom de la machine distante.

L'opération se déroule en plusieurs étapes transparentes pour l'utilisateur :

1. Le client entre la commande d'affichage en précisant la machine concernée.
`$ diet-list -config <fichier de configuration DIET> -host <hote> [--nball
|--nbrun |--nbwait].`
2. L'application cliente contacte le master agent (MA) afin d'obtenir une référence au SeD de l'hôte concernée.
3. Le logiciel client transmet la demande au serveur.
4. Le serveur transmet la liste des jobs au client.

Les option `--nball` , `--nbrun` , et `--nbwait` sont facultatives et permettent respectivement de fournir le nombre total de job sur la plate-forme, le nombre de jobs running sur la plate-forme et le nombre de job waiting sur la plate-forme.

2.3.3 Annulation d'un job

Le client souhaite annuler un job en attente ou en cours d'exécution sur une machine distante. Pour cela il utilise la commande **diet-cancel** en lui passant en paramètre l'identification du job et le nom de la machine distante.

L'opération se déroule en plusieurs étapes transparentes pour l'utilisateur :

1. Le client entre la commande d'annulation en précisant l'identification du job et le nom de la machine concernée.
`$ diet-cancel -config <fichier de configuration DIET> <Job ID> -host <hote>.`

2. L'application cliente contacte le master agent (MA) afin d'obtenir une référence au SeD de l'hôte concernée.
3. Le logiciel client transmet la demande d'annulation avec l'identification du job au serveur.
4. Le serveur exécute la commande d'annulation et transmet les erreurs éventuelles au client.

3 Logiciels et documents fournis

Le logiciel DIET *batch management system* décrit dans ce document comprend :

- Le code source de l'ensemble des logiciels fournis :
 - Le code source du serveur DIET *batch management system* permettant la soumission, la consultation de la liste des jobs et de l'annulation d'un job.
 - Le code source des clients DIET *batch management system* permettant les exécutions des différentes commandes, ainsi que les binaires pour les différentes machines cibles.
 - Les fichiers *Makefile* permettant la compilation des exécutables *client* et *server*.
 - Le code source de la librairie de l'API ainsi que les fichiers d'en-têtes nécessaire pour son utilisation.
- La documentation des logiciels fournis comprend :
 - Un guide de compilation/installation des logiciels.
 - Un guide de configuration.
 - Un guide d'utilisation.
 - Une documentation au format html du code et des pages man d'utilisation des différentes commandes disponibles
 - Un ensemble de pages man pour les différentes commandes et les structures utilisées par l'API.