

The logo for France Grilles features a stylized blue map of France with a white grid pattern overlaid on it. The text "France Grilles" is written in a bold, black, sans-serif font across the map. The entire logo is set against a dark background with a pattern of blue binary code (0s and 1s) and green lines that suggest a network or data flow.

France **Grilles**

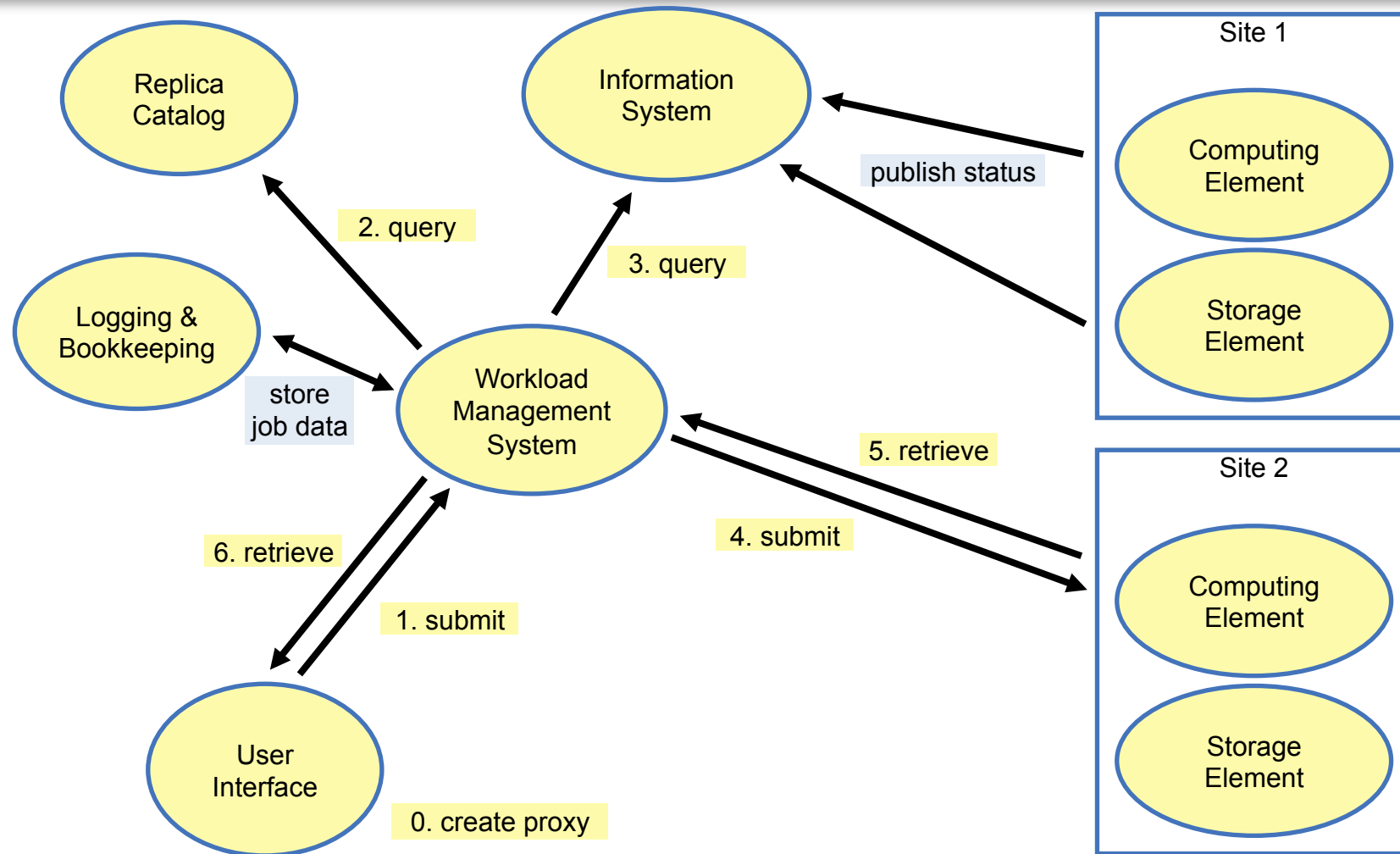
Soumission des jobs

C. Loomis (CNRS/LAL)

- **Les calculs sur la grille sont fait avec les jobs comme un grand système de batch.**
- **Les différents composants**
- **Les principales commandes**
- **Job Description Language (JDL)**

- **UI (User Interface)**
 - Point d'accès pour les utilisateurs contient les outils clients grilles
- **WMS (Workload Management System)**
 - Un service intermédiaire responsable de trouver les « meilleures » ressources où soumettre les jobs
 - Anciennement appelé « Resource Broker » (RB)
- **LB (Logging and Bookkeeping)**
 - Stocke les informations concernant les états successifs des jobs
- **BDII (Information Index)**
 - Un serveur LDAP qui collecte les informations concernant les ressources grille

Soumission d'un Job



- **Machine « en dehors » de la grille qui contient les outils pour interagir avec les services grilles**
 - Acquérir un proxy
 - Soumission et gestion des jobs
 - Transfert et gestion des données
- **Soumission de jobs**
 - Commandes comme `glite-wms-job-*`
 - APIs complets C, C++ et Java sont disponibles
 - Commandes `glite-job-xxx` et `edg-job-xxx` ne sont plus supportées

- **glite-wms-job-submit [-d deleg_proxy-a] fichier_jdl**
 - Soumets un job; retourne le jobID
- **glite-wms-job-status [-v 1|2] jobID**
 - Donne le statut du job
- **glite-wms-job-output jobID**
 - Récupère les fichiers spécifiés dans l'attribut OutputSandbox
- **glite-wms-job-cancel jobID**
 - Annule un job
- **glite-wms-job-delegation-proxy -d identifier**
 - Crée un delegation proxy à utiliser lors du submit
- **glite-wms-job-list-match fichier_jdl**
 - Liste les ressources compatible avec la description du job
 - Effectue le « matchmaking » sans soumettre le job
- **glite-wms-job-logging-info jobID**
 - Donne des informations sur les jobs soumis (tout les événements répertoriés par les divers composants du WMS)

- **Commandes glite-job-* et edg-job-* ne sont plus supportées**

- **Les commandes de soumission utilisent un fichier de description du job (JDL)**
 - On ne soumet pas directement son programme
 - L'application peut être préinstallée sur la grille
- **La soumission retourne un job identifier**
 - Indispensable de le conserver pour pouvoir récupérer des informations et les résultats
 - Option `-o` permet de sauvegarder le(s) jobID(s) dans un fichier
 - Option `-i` permet l'utilisation des jobIDs sauvegardés avec les autres commandes
- **Proxy délégation : nécessaire pour interagir avec le WMS**
 - Automatique : option `-a`, effectuée lors soumission
 - Explicite : `glite-wms-job-delegate-proxy + -d` à la soumission

- **On peut suivre le statut du job pendant son exécution et aussi après il se termine.**
- **Le statut normale : glite-wms-job-status**
 - Utiliser ‘—v 1’ sinon verbosité insuffisante
 - Utiliser l’option ‘—i jobids_file’ si le jobid enregistré dans un fichier (—o lors de la soumission)
 - Utiliser ‘watch —n secondes..’ pour avoir un suivi « temps réel »
 - ‘—all’ permet de voir le statut de tous les jobs de l’utilisateur
- **Statut détaillé : glite-wms-job-logging-info**
 - Utiliser ‘—v 1’ ou ‘—v 2’ pour plus des informations
 - Utile pour déboguer les problèmes



Résultats

- **Récupération des résultats :**
 - glite-wms-job-output**
 - A l'initiative de l'utilisateur
 - Fichier listés dans le « output sandbox »
 - Résultats stocké dans une répertoire locale
 - Conservé environ trois semaines sur le WMS

Exemple d'un Fichier JDL

attribut job

```
Executable = "gridTest";  
StdError = "stderr.log";  
StdOutput = "stdout.log";  
InputSandbox = {"/home/joda/test/gridTest"};  
OutputSandbox = {"stderr.log", "stdout.log"};
```

attribut
données

```
InputData = "lfn:testbed0-00019";  
DataAccessProtocol = "gridftp";
```

attributs
ressources

```
Requirements =  
    other.Architecture == "INTEL" &&  
    other.OpSys == "LINUX" &&  
    other.FreeCpus >= 4;  
Rank = "other.GlueHostBenchmarkSF00";
```

Les Attributs JDL

➤ Attributs du job

- Défini le job lui-même

➤ Attributs sur les ressources

- pris en compte par le WMS et utilisé par l'algorithme de matchmaking (choix du site)
- ressources de calcul (CE et clusters associés) principalement

➤ Attributs sur les données

- Complémentaires des ressources
- Prise en compte de la localisation des données pour le choix du site



Définition des Jobs

- **Executable (obligatoire)**
 - Le nom de la commande
- **Arguments (optionnel)**
 - Arguments de la ligne de commande du job
- **StdInput, StdOutput, StdError (optionnel)**
 - Standard input/output/error du job
- **Environment (optionnel)**
 - Liste de variables d'environnement
- **InputSandbox (optionnel)**
 - Liste de fichiers nécessaire lors de l'exécution du job
 - Les fichiers à transférer entre le UI et grille vers le WMS
 - Taille maximal est une paramètre du WMS (généralement < 1 Mo)
- **OutputSandbox (optionnel)**
 - Liste des fichiers, générés par le job, qui seront récupérés
 - Taille maximum généralement limitée à 10 Mo

- **Uniquement les ressources qui satisfaites *toutes les constraints* listés comme « Requirements » peuvent être sélectionnés pour le job**

- **Expression**
 - Peut être complexe (syntaxe Condor ClassAds)
 - Spécifié en utilisant les attributs des ressources publiées
 - Si non spécifié, la valeur par défaut (de l'UI) est utilisé

- **Les valeurs ne sont pas passées au site**
 - Les valeurs peuvent pas être respecter aux sites
 - Va devenir possible avec le CREAM CE

- **Exprime comment classer les ressources sélectionnées par les Requirements**

- **Expression**
 - Peut être complexe (syntaxe Condor ClassAds)
 - Spécifié en utilisant les attributs des ressources publiées
 - Si non spécifié, la valeur par défaut (de l'UI) est utilisé

- **Ressource sélectionnée**
 - Toujours le ressource avec le plus grand valeur



Attributs pour les données

➤ **InputData (optionnel)**

- Données se trouvant sur la grille utilisées en entrée : elles sont généralement publiées dans le catalogue LFC de la VO
- Site (CE) sélectionné doit avoir pour 'close SE' un des SE détenant un des réplicas (si possible)
- A ne pas confondre avec « InputSandbox »

➤ **DataAccessProtocol (obligatoire avec InputData)**

- Le protocole ou la liste des protocoles avec lesquels l'application est susceptible d'accéder aux données

➤ **But : faire une seule soumission pour plusieurs jobs**

- Collection arbitraire : 1 JDL par job
 - --collection directory
- DAG (Direct Acyclic Graph) : enchainement de jobs
 - --dag directory
- Le répertoire contient le JDL des sous-jobs

➤ **Contrôler les jobs**

- jobID: 1 pour la collection et 1 pour chaque sous-job
- Output: 1 répertoire par sous-job plus fichier ids_nodes.map
- Statut de la collection reflète celui de l'ensemble des jobs

- **But : permettre d'examiner les fichiers produits par un job pendant son exécution**
 - Peut s'appliquer à tout fichier
 - Requiert deux lignes supplémentaires dans le JDL :
 - `PerusalFileEnable = true;`
 - `PerusalTimeInterval = 120; # In seconds, not too low`
- **Définition et récupération des fichiers à examiner :**
`glite-wms-job-perusal [--set | --get | --unset] -f file jobid`
 - `--set` : définit les fichiers à examiner
 - `--get` : récupère la différence avec la version précédente
 - `--all` : force la récupération de tous les fichiers
 - `--nodisplay` : stocke le fichier plutôt que de l'afficher
 - `--unset` : annule l'examen (la récupération périodique) du fichier
- **A utiliser avec modération**
 - Peut avoir un impact sur les performances du WMS
 - Ne surtout pas utiliser en phase de production

- **Chaque VO dispose d'un espace spécifique pour installer ses applications sur un CE**
 - Espace partagé par le CE et ses WNs
 - Référencé à travers une variable d'environnement :
 - VO_VONAME_SW_DIR
 - VONAME est le nom de la VO avec les '.' remplacés par des '_'
- **Droit d'écriture restreint au seul VO Software Manager**
 - Accessible en lecture à tout le monde (toutes les VOs)
 - Software Manager définit avec un rôle VOMS (au choix de la VO)
- **Les logiciels installés :**
 - Mise à jour effectuée en soumettant des jobs avec le rôle Software Manager
 - Contenu peut être publié en définissant des tags depuis un UI ou un WN (job)
 - lcg-ManageVOTag -host CE -vo voname ...



Soumission Directe (CREAM CE)

- **Nouvelle génération du CE (CREAM CE) permet la soumission directe au CE.**
 - En cours de déploiement : pas disponible sur tous les sites
 - Architecture très différente du LCG CE, plus performant

- **Plusieurs mode de soumissions**
 - Via un WMS : pas de différence avec le LCG CE
 - Soumission directe : plus efficace mais pré-sélection du site
 - Via Condor : quelques VO's utilisent ce système

- **Soumission directe**
 - JDL Identique au WMS mais seulement la section « job attributes »
 - Même structure des commandes que le WMS
 - Commandes : glite-ce-xxx
 - Pas (encore) de commande glite-ce-output : peut être complexe

- **Le grille fonctionne comme un grand système de batch**
 - Le composant central est le WMS
 - Son rôle est de trouver la meilleure ressource à partir des contraintes (Requirements et Rank) données par l'utilisateur
 - Il utilise le système d'information (BDII) pour sélectionner le site
- **Le service WMS simplifie la soumission de job :**
 - « Bulk » soumission : job collections, DAG jobs, ...
 - Shallow resubmission, Fuzzy Ranking...
 - VOMS proxy renewal (y compris les attributs VOMS)
 - Peut soumettre plus de 20 kjobs/jour/WMS
- **Pas la seule solution :**
 - Autres brokers comme GridWay, DIRAC, PanDA...
 - « Workflow managers » comme TAVERNA, MOTEUR, ...
 - Soumission directe (CREAM CE) pour des utilisations avancées