Почти все о данных

ЧАСТЬ 2 Кто владеет информацией, тот владеет миром. **Андрей Е. Шевель** покажет, где ее брать и как передавать, а также кратко рассмотрит информационный сервис *gLite*.

Наш эксперт

Андрей Е. Шевель
Зав. отделом вычислительных систем в Отделении физики высоких энергий Петербургского института ядерной физики Российской академии наук. Основной научный интерес – распределенные вычислительные системы.

Данные и файл-каталог

Как правило, крупные задачи предполагают использование больших объемов данных. Термин «большие» в настоящее время означает много ТБ (10^{12} байт или 1024 ГБ). Может также оказаться, что эти данные будут представлены миллионом файлов (или около того). Естественно, они будут распределены между несколькими (или даже многими) компьютерными SE-установками (**LXF111**). Как следствие, требуется где-то хранить и поддерживать информацию обо всех файлах, имеющихся в распределенной системе. При этом полезно иметь два пространства имен:

- **Логические имена** – не зависят от фактического местоположения файла (единое пространство).
- **Физические имена**, т.е. «адреса», имеющиеся в реальности, например, **cluster17:/home/abcd/file18563**. Это означает, что файл находится на кластере с именем **cluster17** в директории **/home/abcd**.

Вполне очевидно, что должен иметься некоторый механизм отображения логических имен файлов в физические и обратно – он и зовется «файл-каталогом». В этом же каталоге должно указываться, имеются ли копии конкретного файла (реплики) на других машинах или кластерах, какая копия является главной и т.п. Понятно, что если появляется новый файл, то он должен быть добавлен к файл-каталогу, при удалении файла – удален из файл-каталога. При перемещении файла с одного кластера на другой, естественно, требуется корректировать файл-каталог. Все действия с файл-каталогом необходимо выполнять синхронно с операциями над файлом (или группой файлов).

Все это (и многое другое) реализуется в *gLite* механизмом *LCG File Catalog* (LFC). LFC обеспечивает отображение между именами файлов в нескольких ипостасях:

- Уникальный идентификатор файла (Grid Unique Identifier), GUID. Он имеет вид **guid:38ed3f70-c608-12d5-f6c1-41ff69d7a449** (строка из 36 байт).
- Логическое имя файла (Logical File Name, LFN). LFN есть некоторое имя, с которым удобно иметь дело потребителю информации, например, **lfn:MyOwnPhysicsTest13246**.
- Физическое имя файла (PFN) или Storage URL (SURL). Оно имеет форму **<protocol>://<SE-hostname>/<Directory hierarchy><file-name>**. Иерархия директорий выглядит так: **/grid/<VO>/<directory>**. Например, **sfn://tbed0101.cern.ch/data/dteam/doe/file144**.

В целях дальнейших рассуждений условимся называть грид-файлом файл, существующий физически на одном из SE и зарегистрированный в LFC. Говоря о данных, мы будем рассматривать их как множество грид-файлов, имена которых имеют форму, принятую в *gLite*, если специально не оговорено иное.

В *gLite* имеется набор команд для работы с LFC. Список команд приведен в таблице 1.

Таблица 1. Команды работы с каталогом LFC

Команда работы с каталогом LFC	Краткое описание
lfc-chmod	Изменить права доступа к файлу/директории
lfc-chown	Изменить владельца и группу файла/директории
lfc-delcomment	Удалить комментарий, связанный с файлом/директорией
lfc-getacl	Получить список доступа к данному файлу/директории
lfc-ln	Создать логическую ссылку на файл/директорию
lfc-ls	Вывести информацию о файле или о составе директории
lfc-mkdir	Создать директорию
lfc-rename	Переименовать файл/директорию
lfc-rm	Удалить файл/директорию
lfc-setacl	Установить список доступа для файла/директории
lcf-setcommet	Добавить/заменить комментарий для файла/директории
lfc-entergrpmap	Определить новую группу в таблице виртуальной идентификации
lfc-enterusrmap	Определить нового пользователя в таблице виртуальной идентификации
lfc-modifygrpmp	Модифицировать запись о группе, соответствующей виртуальному GID в таблице виртуальной идентификации
lfc-modifyusrmap	Модифицировать запись о пользователе, соответствующую виртуальному UID в таблице виртуальной идентификации
lfc-rmgrpmap	Запрещает запись о группе, соответствующей данному виртуальному GID или имени группы
lfc-rmusrmap	Запрещает запись о пользователе, соответствующую данному виртуальному UID или имени пользователя

Как упоминалось ранее, все операции с LFC должны быть синхронизированы с действиями над данными. Иначе говоря, реальное расположение файлов в гриде должно соответствовать состоянию каталога LFC. Перемещение данных в грид может происходить различными способами, утилитами разных уровней. Здесь мы кратко остановимся на клиентском интерфейсе к базовым средствам *gLite*

МЕСЯЦ НАЗАД Азбука гридов: мы рассмотрели возможности данной технологии и познакомились с базовыми понятиями.

по управлению данными. Эти средства существуют в виде команд и API.

Перемещение и копирование

Данные утилиты (команды) маскируют технические сложности взаимодействия с SE и LFC, одновременно выполняя всю необходимую синхронизацию между ними. Кроме того, этот инструментарий предоставляет пользователю возможность копировать файлы между UI, CE, WN и SE, корректировать соответственно LFC и реплицировать (копировать) файлы с одного SE на другом SE. Краткий обзор указанных средств приведен в таблицах 2 и 3.

Таблица 2. Список команд управления репликой файла

Команда управления репликой	Краткое описание
lcg-cp	Копирование грид-файла на локальный компьютер
lcg-cr	Копирование локального файла в один из элементов SE и регистрация файла в каталоге LFC
lcg-del	Удалить один файл (одну реплику или все реплики)
lcg-rep	Скопировать файл из одного элемента SE в другой элемент SE и зарегистрировать реплику в каталоге LFC
lcg-gt	Получить TURL и протокол обмена для данного SURL
lcg-sd	Установить состояние DONE (выполнено) для данного SURL при запросе по протоколу srm

Таблица 3. Команды взаимодействия с каталогом

Команда взаимодействия с каталогом	Краткое описание
lcg-aa	Добавить в каталог псевдоним для данного GUID
lcg-ra	Удалить псевдоним для данного GUID
lcg-rf	Зарегистрировать в каталоге файл, который находится на конкретном элементе SE
lcg-uf	Удалить из каталога регистрацию файла на конкретном SE
lcg-la	Вывести все псевдонимы файла для данного LFN или GUID или SURL
lcg-lg	Получить GUID для данного LFN или SURL
lcg-lr	Перечислить все реплики для данного GUID или LFN или SURL

Рассмотрим две практических ситуации. Пусть нам требуется поместить локальный файл **/home/shevel/file13476** в грид и присвоить ему логическое имя «**my_file13476**». Это делается так:

```
lcg-cr - -vo dteam -d lxb0710.cern.ch -l lfn:my_file13476 file:/home/shevel/file13476
```

После передачи и регистрации система ответит строкой с указанием GUID.

Возможна и обратная ситуация – копирование из грида на клиентскую машину:

```
lcg-cp --vo dteam lfn:my_file13476 file:/home/shevel/file-13476_from_grid
```

В *gLite* имеется ряд других сервисов по передаче данных, в частности, предназначенных для массовой надежной передачи большого числа файлов.

Надежная передача

Задача массовой передачи файлов является весьма серьезной для потребителей, работающих с большими объемами данных. Перемещение (копирование), например, нескольких сотен тысяч файлов со средним объемом 100 МБ с одного элемента памяти SE на другой SE является здесь обычным действием. Так происходит, например, при распределении файлов из точки, где они генерируются, по кластерам, где они будут обрабатываться. Могут быть и другие причины. Очевидно, чтобы передать данные в такой форме, необходимы специальные системы, гарантирующие надежную передачу. Под этими словами понимается следующее:

- Передача производится автоматически или с минимальным ручным вмешательством;
- При передаче гарантируется, что любой переданный файл является точной копией передаваемого файла;
- Если в сети возникают какие-либо проблемы, они преодолеваются автоматически или полуавтоматически;
- Наконец, при передаче файлов автоматически выполняются необходимые операции с каталогом LFC. По окончании передачи состояние файлов должно соответствовать содержанию каталога LFC.

Для выполнения данных операций в *gLite* используется система надежной передачи файлов – File Transfer Service, или FTS. Этот сервис обеспечивает надежную передачу от одного элемента SE к другому по протоколу «точка–точка» (без маршрутизации). Вследствие того, что система FTS является развивающейся, в ней пока нет взаимодействия с каталогом LFC (видимо, это будет добавлено в будущих релизах). Иными словами, при использовании данного сервиса (копировании) следует дополнительно позаботиться о внесении в каталог необходимых записей о реплицированных (скопированных) файлах. Сервис FTS действует, как правило, между крупнейшими центрами обработки данных (Tier0, Tier1, некоторые из Tier2) в гриде. Сервис состоит из нескольких взаимодействующих программных подсистем (агентов). Центральным звеном организационной структуры FTS является база данных сервиса.

Фундаментальными понятиями в сервисе FTS являются следующие.

- **Transfer Job** – задание по передаче данных. Это набор (массив) файлов, которые должны быть переданы из одной точки в другую. Задание может содержать параметры для нижележащего транспортного слоя (GridFTP).
- **File** – пара адресов «откуда/куда» в формате SURL.
- **Job State** – состояние задачи передачи файлов.
- **Channel** (канал передачи) – логический сетевой механизм по передаче файлов. **Production Channel** – высокопроизводительные каналы передачи данных между крупнейшими центрами, как правило, имеющие гарантированный минимум пропускной способности. **Non-production Channel** – любой канал связи, не гарантирующий минимум пропускной способности.

Поскольку задание по передаче данных может выполняться продолжительное время (например, сутки или более), то оно может находиться в ряде состояний, а завершение выполнения такого задания обозначается несколькими исходами.

- **Submitted** – задание запущено в FTS, но ему пока не назначен канал передачи.
- **Pending** – заданию уже выделен канал передачи данных, но сама передача пока не началась.
- **Active** – задание находится в процессе передачи как минимум одного файла.
- **Cancelling** – задание находится в процессе непланового завершения.
- **Done** – задание завершено плановым образом, т.е. все файлы, которые были обозначены в задании, успешно переданы.
- **Failed** – задание по передаче файлов завершено, однако один или более файлов не удалось передать успешно.
- **Cancelled** – задание по передаче файлов завершено неплановым образом.
- **Hold** – задание требует вмешательства оператора.

Очевидными конечными состояниями задания по передаче данных являются **Done**, **Cancelled** и **Failed**.

Такие относительно детальные описания состояний передачи определяются тремя важными факторами: объемом данных и числом передаваемых файлов (например, 100 ТБ в 500 000 файлов), а также географической разнесенностью элементов **SE** (например, передача из Восточной Европы в Австралию, или из Северной Америки в Японию).

Перед началом передачи файлов требуется зарегистрировать персональный грид-сертификат на специальном прокси-сервере, который используется сервисом FTS.

```
myproxy-init -s myproxy-fts.cern.ch -d
```

Здесь указан прокси-сервер (параметр -s) и запрос на использование имени из вашего сертификата (параметр -d) в качестве имени пользователя в задании на передачу данных.

Теперь для начала передачи данных нужно лишь воспользоваться командой glite-transfer-submit. На самом деле, процесс стартует лишь после выделения канала передачи, который обеспечивается соответствующим администратором. Кроме того, пользователь должен позаботиться о свободном дисковом пространстве, на котором он планирует разместить данные.

Желающим познакомиться с архитектурой передачи данных более основательно рекомендуем статью В. Коренькова и А. Ужинского «Архитектура сервиса передачи данных в grid», опубликованную в журнале «Открытые системы» №2 (2008) и доступную по адресу http://www.osp.ru/os/2008/02/4926522/

В дополнение к технике запуска заданий и передачи данных, было бы неплохо знать, где находятся подходящие сервисы SE и CE. Такую информацию можно получить через информационный сервис *gLite*.

Информационный сервис

Информационный сервис в *gLite* представляет собой ряд серверов специального назначения, расположенных в Интернете. Они собирают, хранят и предоставляют по запросу сведения о различных сервисах грида. Например, информационный сервис помогает определить, на каком кластере следует запустить задание пользователя на обработку данных. Имеются две утилиты достаточно высокого уровня для получения информации с таких серверов: *lcg-infosites* и *lcg-info*.

```
$ lcg-infosites --vo alice ce
#CPU|Free|Total Jobs|Running|Waiting|ComputingElement
----------------------------------------------------------
  40    40    0        0     0    lcg06.sinp.msu.ru:2119/
jobmanager-lcgpbs-alice
  44    8    37        9    28    lcg02.sinp.msu.ru:2119/
jobmanager-lcgpbs-alice
 108    78    1        0     1    lcg38.sinp.msu.ru:2119/
jobmanager-lcgpbs-alice
   4    4    0        0     0    grid129.sinp.msu.ru:2119/
jobmanager-lcgpbs-alice
```

Здесь мы запросили информацию о вычислительных элементах CE для виртуальной организации ALICE. Примерно то же самое можно выполнить и посредством команды lcg-info:

```
lcg-info --list-ce --vo alice
CE: grid129.sinp.msu.ru:2119/jobmanager-lcgpbs-alice
CE: lcg02.sinp.msu.ru:2119/jobmanager-lcgpbs-alice
CE: lcg06.sinp.msu.ru:2119/jobmanager-lcgpbs-alice
CE: lcg38.sinp.msu.ru:2119/jobmanager-lcgpbs-alice
```

А вот так можно получить список ближайших (в том или ином смысле) элементов SE.

```
lcg-infosites --vo alice closeSE
Name of the CE: lcg06.sinp.msu.ru:2119/jobmanager-lcgpbs-alice
      lcg59.sinp.msu.ru
Name of the CE: lcg02.sinp.msu.ru:2119/jobmanager-lcgpbs-alice
      lcg59.sinp.msu.ru
Name of the CE: lcg38.sinp.msu.ru:2119/jobmanager-lcgpbs-alice
      lcg59.sinp.msu.ru
Name of the CE: grid129.sinp.msu.ru:2119/jobmanager-lcgpbs-alice
      lcg59.sinp.msu.ru
```

Содержательно информационный сервис связан с мониторингом, т.е. отображением состояния грид-системы. Он также является весьма нетривиальным и объемным по предоставляемой информации. Например, можно обратиться к странице http://pcalimonitor.cern.ch/map.jsp, где отражено состояние грида в виртуальной организации ALICE (**рис. 1**).

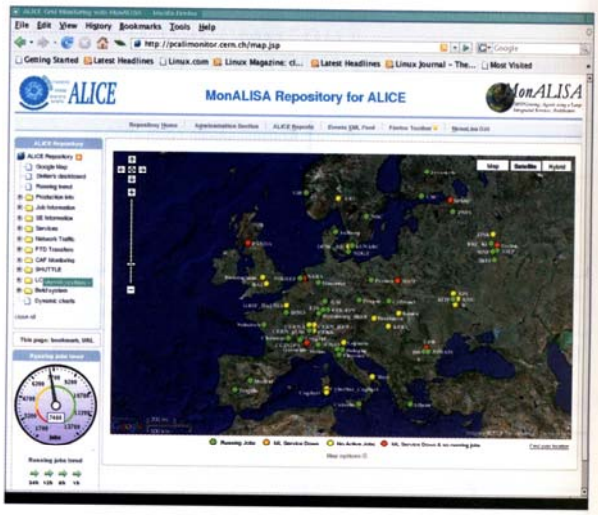

Рис. 1. Страница мониторинга VO ALICE.

А на странице http://rocmon.jinr.ru:8080/display?page=site_jobs_rt показано состояние выполняемых на кластерах RDIG заданий (**рис. 2**).

Рис. 2. Распределение вычислительных заданий по кластерам проекта RDIG (Россия).

Организационные аспекты

Как мы видели выше, структура грида содержит массу разнообразных сервисов (и серверов) – кто-то должен их поддерживать. Иными словами, необходим некоторый оплачиваемый персонал, который обеспечивал бы следующее:

- Работоспособность всех сервисов грида, включая постоянный мониторинг готовности системы в целом;
- Описания программ/систем в соответствии с реально работающими компонентами;
- Консультации пользователей по различным аспектам применения *gLite*.

Необходимо заметить, что в гриде используются вычислительные ресурсы из административно различных и независимых организаций (университетов, исследовательских лабораторий, компаний, государственных органов). Каждая организация, предоставляющая свои вычислительные ресурсы, должна подписывать специальный документ о взаимопонимании, где изложен порядок и объем предоставления упомянутых ресурсов пользователям грида и указано, какие виртуальные организации могут использовать эти вычислительные ресурсы.

Естественно, должны иметься специальные регистрационные (сертификационные) центры, которые выдают электронные сертификаты потенциальным пользователям и вычислительным элементам, а также устанавливают принадлежность пользователей к той или иной виртуальной организации.

Эффект от использования грида конкретными пользователями той или иной конкретной виртуальной организации сильно зависит от степени скоординированности действий как администрации грид-систем, так и администрации виртуальной организации.

Что дальше?

Желаете освоить работу в гриде более глубоко? Неплохой источник информации по компонентам *gLite* находится по адресу: **https://grid-deployment.web.cern.ch/grid-deployment/the-LCG-Directory/the-LCG-directory.html**. Можно использовать демонстрационный сайт для обучения – **https://gilda.ct.infn.it**. Любой человек может зарегистрироваться здесь с тем, чтобы попробовать те или иные демонстрационные возможности грида. Российский сегмент гридов для интенсивных вычислений с большим объемом данных представлен проектом RDIG (**http://www.egee-rdig.ru**). Информацию о различных грид-проектах, а также по кластерной технологии можно найти на сайте **www.ClusterGate.ru**.

Из краткого рассмотрения простых грид-средств видно, что эта архитектура предназначена главным образом для весьма сложных задач обработки данных, которые не могут быть решены другими способами. В решении таких задач часто принимают участие многие десятки или сотни человек, распределенных географически, которые действуют относительно независимо и могут даже не знать о существовании друг друга. Неудивительно, что попытки решения таких задач без использования грида все равно приводят к созданию тех или иных вариантов данной архитектуры.

Все сказанное выше о сложности задач для гридов и, частично, об использовании данной структуры не означает, что такое положение останется навсегда. Мы являемся свидетелями происходящей смены парадигмы выполнения вычислений: происходит переход от локальных и относительно небольших вычислений к распределенным и крупномасштабным вычислениям. Не исключено, что в будущем использование гридов станет более очевидным и простым решением для любых или практически любых задач. Такое будущее становится более вероятным с каждым годом, поскольку насыщенность окружения любого человека компьютеризированными устройствами растет огромными темпами (смартфоны, КПК, плейеры, настольные домашние компьютеры, ноутбуки и т.п.). Факт такой насыщенности выносит на повестку дня интегрирование разнородных компьютерных устройств для скоординированного обслуживания ими человека. LXF