Письма в ЭЧАЯ. 2010. Т. 7, № 6(162). С. 699–704

МЕТОД КЛОНИРОВАНИЯ ГРИД-ЭЛЕМЕНТА

И. В. Бедняков, А. Г. Долбилов, Ю. П. Иванов

Объединенный институт ядерных исследований, Дубна

В работе описан простой, надежный и быстрый метод ввода в строй новых вычислительных мощностей для работы в грид-среде. Метод успешно опробован в рамках грид-сегмента ЛЯП ОИЯИ, используемого для работы коллаборацией АТЛАС.

This paper presents a simple, reliable and fast method of putting into operation new computational capabilities for operation within the grid environment. The method is successfully tested within the grid segment of the DLNP of JINR and has been used for operation in the ATLAS collaboration.

PACS: 31.15.A-; 81.15.at

ВВЕДЕНИЕ

В современном мире работает множество самых разных компьютеров. Причем далеко не все они загружены вычислениями в полную силу круглые сутки. Одни считают с утра до вечера, другие делают это вполсилы, а третьи, вообще, большую часть времени простаивают, часами ожидая хотя бы случайного прикосновения к клавиатуре. В тех странах, где сейчас ночь, компьютеры, как правило, слабо загружены. В то же время на другой стороне Земли, где сейчас день, порой катастрофически не хватает вычислительной мощности: метеорологи не могут точно предсказать погоду, нефтяники не справляются с расчетом контура месторождения, автомобилестроители годами моделируют, скажем, самую обтекаемую и безопасную машину, и т. д. и т. п. По существу, на основе этого противоречия возникала очень простая идея: обеспечить тому, кому это надо, доступ к свободным компьютерным ресурсам. Несколько упрощая ситуацию, можно сказать, что именно эта идея и легла в основу концепции GRID.

На самом деле современное понятие грид очень широкое, тем не менее в контексте рассматриваемой проблемы можно сказать, что грид — это географически распределенная инфраструктура, объединяющая множество ресурсов разных типов (процессоры, долговременную и оперативную память, хранилища и базы данных, сети и т. д.), доступ к которым пользователь может получить из любой точки мира независимо от места ее расположения (см., например, [1–3]). Создание и быстрое распространение грид стало необходимым ввиду сильно возросшего числа сложных научно-прикладных задач, для решения которых требуются огромные вычислительные ресурсы. Этому также способствовала возросшая скорость и надежность коммуникационных каналов связи, ставшая достаточной для эффективной передачи больших объемов информации.

Задача передачи, хранения и обработки беспрецедентных объемов информации стала особенно актуальна в связи с участием ОИЯИ в уникальных экспериментах на большом

адронном коллайдере (Large Hadron Collider — LHC) в европейской организации ядерных исследований ЦЕРН (Женева, Швейцария). В Лаборатории информационных технологий ОИЯИ, где были созданы все необходимые коммуникационно-вычислительные ресурсы для хранения, обработки и анализа данных с LHC, с 2002 г. работает элемент всемирной грид-структуры LCG (LHC Computing Grid) [4]. Это дает все основания надеяться на успешное участие ОИЯИ в экспериментах на LHC.

В свою очередь, Лаборатория ядерных проблем является базовой структурой, посредством которой ОИЯИ участвует в эксперименте ATLAS (A Toroidal LHC ApparatuS). Этот многоцелевой эксперимент является одним из четырех основных экспериментов на коллайдере LHC. Он будет проводиться на одноименном детекторе, предназначенном для всестороннего исследования разнообразных явлений, возникающих в результате протон-протонных столкновений при высоких энергиях (см., например, [5]).

Для обеспечения эффективного участия в эксперименте ATLAS в ЛЯП было принято решение создать свой грид-кластер, ориентированный первоначально на потребности и специфику математического обеспечения именно этого эксперимента (см., например, [3]). В 2005 г. этот кластер стал вторым в ОИЯИ грид-сегментом общей структуры JINR-LCG2, входящей в LCG. Наличие в ОИЯИ нескольких связанных вместе, но «географически разделенных» грид-сегментов полностью отвечает главной идее GRID о распределенных вычислениях.

Имеющиеся в настоящий момент основные компоненты грид-кластера ЛЯП приведены в таблице. Здесь компьютер lgdce01 является «вычислительным элементом» (CE — Computing Element), компьютеры lgdwn01... lgdwn10 представляют собой «рабочие узлы» (WN — Worker Node) и lgdui01 — «интерфейс пользователя» (UI — User Interface). Именно через эту машину (UI) осуществляется доступ пользователей к кластеру ЛЯП для подготовки и запуска как локальных задач, так и с использованием системы грид. В кластер входят также компьютеры lgdfs01 и lgdfs02, являющиеся компонентами файловой системы AFS ОИЯИ (распределенная файловая система), на которой расположены как программы и данные различных экспериментов, так и файлы пользователей. На всех машинах установлена операционная система (ОС) Scientific Linux 4 (SL4), а в качестве программного обеспечения GRID middleware используется gLite 3.1.

Схема грид-кластера ЛЯП

Компьютер	Система	Функция
lgdce01	SL4	CE
lgdwn01... lgdwn10	SL4	WN
lgdui01	SL4	UI
lgdfs01... lgdfs02	SL4	AFS server

Кластер ЛЯП успешно работает, и его вычислительные мощности расширяются. Грид-сегмент ЛЯП непосредственно участвует в обработке информации эксперимента ATLAS, а также используется для тестирования нового оборудования (hardware, software, middleware) и обучения сотрудников работе в среде грид.

В целом Лаборатория ядерных проблем насчитывает примерно 1300 компьютеров, подключенных к локальной сети ОИЯИ. Из них 300 компьютеров занимаются обеспечением различных сервисов таких, как почтовые серверы, веб и т. п. Из остальных 1000 компьютеров примерно половина обеспечена достаточной вычислительной мощностью

и может быть использована для работы в грид-среде. Таким образом, уже сегодня в ЛЯП примерно 500 компьютеров по желанию пользователей могут стать рабочим элементом грид-кластера ЛЯП.

Таким образом, наращивание вычислительной мощности грид-сегмента ЛЯП имеет вполне ясные перспективы, более того дальнейшее увеличение числа компьютеров в грид-кластере необходимо, поскольку растет как число научно-прикладных задач, требующих все больших вычислительных затрат, так и число пользователей, желающих проводить быстро и эффективно свои вычисления.

Массовое подключение компьютеров к грид-сегменту требует разработки надежной и быстрой процедуры установки и настройки системы и грид-программ.

В данной статье описывается один из возможных методов такой установки.

Метод заключается в непосредственном клонировании «рабочего элемента» с одного компьютера, уже работающего в кластере, на другой, новый компьютер. После такого клонирования необходима лишь минимальная настройка нового элемента. Этот метод, по мнению авторов, может существенно облегчить процедуру ввода в строй новых грид-элементов.

УСТАНОВКА НОВОГО ГРИД-ЭЛЕМЕНТА

Согласно традиционной процедуре установки программного обеспечения (ПО) грид на произвольный компьютер необходимо выполнить определенную последовательность действий. Основные моменты включают в себя сначала установку операционной системы (ОС Linux) и пакета Java, после этого настройку синхронизации времени, установку и настройку NTP-клиента, наладку средства конфигурации YAIM (необходим для создания рабочих аккаунтов) и, наконец, установку так называемого middleware (собственно ПО GRID), настройку поддержки виртуальных организаций (ВО). Этот стандартный метод установки рабочего элемента грид с нуля на каждом компьютере в кластере требует значительного рабочего времени специалиста.

Для начала охарактеризуем рабочий элемент, именно его мы будем клонировать в дальнейшем.

Рабочий элемент кластера должен быть максимально близок по программному обеспечению к остальным элементам кластера и иметь свой уникальный IP, что, естественно, приводит нас к разработке способа клонирования грид-элемента и последующей его минимальной настройке.

Итак, предполагаемый метод состоит из трех основных этапов.

1. Создание образа клона (существующего компьютера), в этот этап входят подэтапы такие, как:

а) нахождение свободного рабочего элемента и его отключение от кластера;

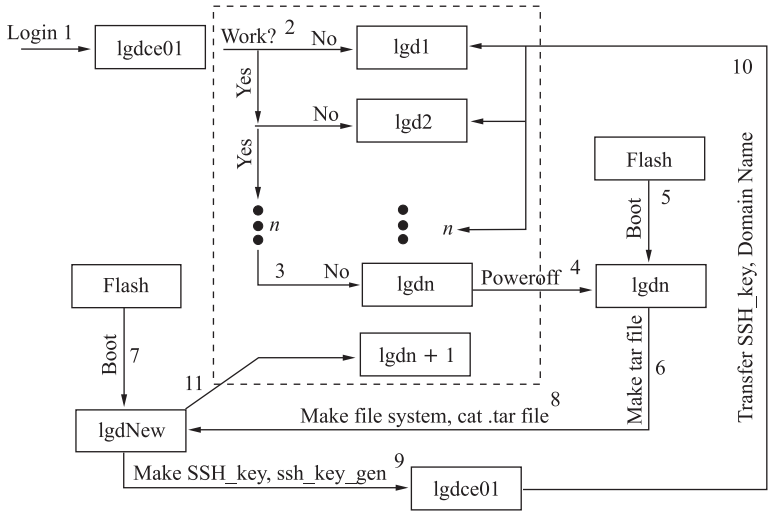
б) создание образа диска или образа файлов.

2. Установка имиджа на новую машину.

3. Настройка новой машины.

Ниже подробно дано описание нового метода настройки, схема которого приведена на рисунке, где цифрами отмечена последовательность действий.

Первый этап проводится на конкретном рабочем элементе кластера.



Метод клонирования рабочего элемента. Цифрами отмечена последовательность действий

Этап первый «Создание образа клона». Проверяем занятость кластера. Такая проверка необходима для сокращения потерь производительной мощности кластера во время отключения одного из работающих компьютеров. Команда showq показывает список задач, команда pbsnodes — short выводит список работающих машин.

После этого выбирается любой незанятый (не загруженный работой) в этот промежуток времени компьютер (например, lgdn, см. рисунок).

С помощью команды poweroff выключаем этот компьютер (lgdn). На этом работа с компьютером lgdce01 завершена.

Диски и разделы. Рассмотрим диски и разделы рабочего элемента ЛЯП.

Файловая система расположена на трех разделах диска:

Лейбл	Диск
/root	/dev/sda1
/tmp	/dev/sda2
/swap	/dev/sda3

На диске /dev/sda1 с лейблом /root расположена основная файловая система, на /tmp расположены временные файлы, на /swap раздел подкачки.

В реальных ситуациях дисковых разделов может быть больше, но для клонирования нам необходим только раздел диска — /root.

Загрузка и создание имиджа. Сначала монтируем с помощью команды mount диск /dev/sda1 в /mnt/sda1.

Переходим в директорию mnt, где с помощью команды tar создаем файл с образом нашей системы cd /mnt/sda1.

Команда tar используется для обслуживания файлового архива на дисках.

Для создания имиджа используем команду tar с параметрами cf(creat file):

tar cf /mnt/sda2/lcgimg.tar — для клонирования линукс системы достаточно сделать tar файл системы, где lcgimg.tar и есть образ нашей системы.

На этом первый этап заканчивается.

Этап второй «Создание клона на новой машине». Загружаемся на новую машину с boot cd или flash-накопителя.

В нашем случае авторы использовали flash-накопитель с RIP linux (http://www.tux.org/pub/people/kent-robotti/looplinux/rip/).

После загрузки с внешнего носителя с помощью команды fdisk с параметром -l находим диск, на котором непосредственно будет создан новый клон. Создаем новые разделы и монтируем их. /mnt.

Для простоты предположим, что у нас компьютер свободен и, следовательно, разделы /sda1 sda2 sda3 не заняты другой операционной системой. После этого с помощью команды mkfs (make file system) создается новая файловая система Linux на компьютере lgdNew. Сначала командой mkfs.ext3 -j /dev/sda1 -L / создаем разделы, соответствующие оригинальному компьютеру.

С помощью параметра -l разделу sda1 присваивается лейбл root. Затем, аналогично,

mkfs.ext3 -j /dev/sda2 -L /tmp создает раздел с названием data и

mkfs swap /dev/sda3 -L /swap — раздел swap. Такая процедура позволяет полностью сохранить основную структуру оригинальной машины.

После создания пустых разделов нам необходимо поместить клон. Имидж-файл можно передать с помощью носителя либо используя сеть.

Монтируем для дальнейшей работы /dev/sda1 /mnt/sda1.

Командой tar -xf lcgimg.tar -C /mnt/sda1 переносим и распаковываем имидж в /mnt/sda1. После этой процедуры получаем распакованный имидж-файл в разделе sda1.

Теперь на новом компьютере в разделе sda1 расположен клон.

Загрузка. Для загрузки операционной системы на новой машине необходима настройка загрузчика (групп).

Нам необходимо убедиться в правильности настройки файла /mnt/sda1/boot/group.conf. В этом файле находятся параметры загрузки.

Проверяем файл idevais.map, в котором находятся параметры загрузки, на соответствие новой машине.

Файлы /mnt/sda1 /boot/group.conf /mnt/sda1 /boot/devaise.map/etc/fstab должны содержать правильную информацию о компьютере.

После проверки запускаем команду

grub-install —root-directori /mnt/sda1 /dev/sda.

Далее перед загрузкой клона нам необходимо поправить файлы настройки сети

mnt/sda1/etc/sysconfig/networkscrypt,

/etc/sysconfig/network,

/etc/sysconfig/network-scritps/ifcfg-eth0

и убедиться в том, что параметры соответствуют сетевым параметрам нового компьютера.

На этом второй этап завершается.

Этап третий «Настройка новой машины». При первой загрузке компьютера будет необходимо установить множество новых драйверов в зависимости от найденного нового оборудования. Для этого необходимо с помощью команды chmod установить права доступа к директории tmp.

/chmod777 /tmp.

Создание SSH-ключей. Как известно, чтобы данные между компьютерами могли беспрепятственно передаваться (каждый раз не выдавая уведомления о том, что тот или

иной компьютер получает данные), необходимо настроить ключи SSH (Secure Shell) и разместить их на всех компьютерах грид-кластера.

Для корректной работы нового элемента нам необходимо создать SSH-ключи. С помощью команды ssh_keygen создаем ключи.

ssh_keygen -t RSA -N " " -f ssh_host_rsa_key.

Сгенерированные ключи вместе с именем компьютера необходимо предоставить администратору для добавления компьютера в список элементов кластера.

Администратор разместит новые ключи в файл hosts.equiv, который находится в директории etc/ssh на центральной машине (lgdce01). С помощью скрипта все файлы на всех машинах кластера изменяются и дополняются ключами новой машины.

На этом заканчивается третья часть.

ЗАКЛЮЧЕНИЕ

В работе описан простой, надежный и быстрый метод ввода в строй новых компьютеров для работы в грид-среде посредством клонирования.

Метод не требует знания о грид-структуре, необходимы лишь простейшие знания ОС Linux.

С его использованием при минимальных навыках система может быть клонирована в течение получаса.

Не требуется постоянное присутствие грид-профессионалов, а также сокращается время. Он был успешно опробован в рамках грид-сегмента ЛЯП ОИЯИ. Метод может быть особенно полезен, например, когда в отсутствие высококвалифицированных грид-специалистов требуется в достаточно короткие временные сроки ввести в строй значительное количество новых рабочих элементов грид-кластера.

Авторы признательны сотрудникам ЛИТ за помощь в создании сегмента грид в ЛЯП. Без советов и разъяснений В. В. Мицина разработка и поддержка грид-кластера ЛЯП была бы более трудоемким процессом. Авторы благодарны также В. А. Беднякову за постоянную моральную поддержку и мотивацию к написанию статьи.

СПИСОК ЛИТЕРАТУРЫ

1. *Foster I., Kesselman C., Tuecke S.* // Intern. J. Supercomp. Appl. 2001. V. 15(3).
2. *Foster I. et al.* // Open Grid Service Infrastructure WG, Global Grid Forum, June 22, 2000.
3. *Долбилов А. Г., Иванов Ю. П.* Сообщ. ОИЯИ Р11-2008-68. Дубна, 2008.
4. LCG Computing Grid in JINR, http://lcg.jinr.ru
5. *ATLAS Collab.* The ATLAS Experiment at the CERN Large Hadron Collider. JINST 3, S08003. 2008.

Получено 25 марта 2010 г.