

Сравнительный анализ подхода к созданию грид-сервисов на основе технологий REST/JSON и разработками проекта European Middleware Initiative

Европейская инициатива по развитию промежуточного программного обеспечения (European Middleware Initiative, EMI, <http://www.eu-emi.eu>) имеет целью объединение крупнейших европейских поставщиков ППО, крупнейшими из которых являются: [gLite](#), [UNICORE](#) и [ARC](#).

Основными целями этого объединения является развитие и унификация ППО для использования в крупных грид-инфраструктурах, например, таких как инфраструктура European Grid Initiative (EGI, <http://www.eu-egi.eu>), повышение надежности работы грид-сервисов, развитие модели стабильно функционирующего и постоянно совершенствующегося ППО в интересах пользователей.

Более детально, целями проекта являются:

- упрощение и отладка различных грид-сервисов, создание самосогласованных, хорошо протестированных, соответствующих стандартам дистрибутивов ППО для грид-инфраструктур и сообществ пользователей;
- повышение унификации, управляемости и эффективности ППО путем разработки и интеграции новых функциональных возможностей;
- поддержка эффективной и надежной работы грид-инфраструктур путем совершенствования ППО;
- привлечение сообществ пользователей к выработке рекомендаций по дальнейшему совершенствованию ППО, а также расширение сотрудничества с национальными и интернациональными проектами в области распределенных вычислений.

Проект EMI рассчитан на три года и в настоящее время находится в начальной стадии (дата начала проекта - 1 мая 2010 г.). Поэтому реальных результатов пока не существует, и о тенденциях развития ППО в рамках проекта EMI можно судить только по заявленным целям и рабочим планам. По крайней мере в ближайшие годы основные усилия будут направлены на развитие и унификацию трех пакетов грид-ПО, указанных выше, а именно: [gLite](#), [UNICORE](#) и [ARC](#). Эти пакеты подробно анализировались на первом этапе выполнения данной НИР. Здесь кратко повторим основные характеристики этих пакетов.

ППО gLite

Пакет gLite является полным решением для грид, включая как базовые низкоуровневые программы, так и ряд служб высокого уровня. В нем интегрированы как компоненты из лучших на настоящий момент проектов ППО, таких как Condor и Globus Toolkit, так и компоненты, разработанные для проекта WLCG. gLite является одним из лучших базовых инструментальных средств, совместимых с такими планировщиками, как PBS, Condor и LSF. ППО gLite разработано с учетом свойств интероперабельности и содержит базовые службы, облегчающие построение

грид-приложений для любых прикладных областей.

Службы gLite соответствуют требованиям SOA (Service Oriented Architecture). Из этого следует, что при необходимости данный продукт можно легко связать с другими грид-службами, а также то, что будет существенно облегчен переход на грядущие стандарты грид. Пакет gLite спроектирован как модульная система, позволяющая пользователям развертывать различные службы в соответствии с их нуждами, а не быть вынужденными использовать всю систему целиком. Предполагается, что это позволит каждому пользователю приспособить систему к его конкретной ситуации.

По сравнению со своими предшественниками (EDG и LCG) ППО gLite гораздо лучше реализует безопасность, имеет лучшие интерфейсы для управления данными и запуска заданий, обладает переработанной информационной системой и многими другими усовершенствованиями, делающими gLite легким и эффективным в использовании.

Если главное направление развития Globus Toolkit – средства разработки и поддержки служб, то основное отличие комплекса gLite в том, что помимо инструментальных средств, в него входит более широкий набор служб.

ППО UNICORE

Унифицированный интерфейс с вычислительными ресурсами — UNICORE (UNiform Interface to COmputing REsources) предоставляет ученым и инженерам ресурсы суперкомпьютерных центров, объединенных в грид, и делает их доступными через интернет. Среда грид UNICORE поддерживает высокий уровень безопасности, а аутентификация осуществляется совместимым между ресурсами и прозрачным для пользователей способом. Различия между платформами полностью скрыты, так что UNICORE можно рассматривать как портал, открывающий бесшовный дистанционный доступ к суперкомпьютерам, компиляции и выполнению приложений и пересылке данных ввода-вывода.

Выигрыш, который получает пользователь UNICORE - однородный доступ к системам разного типа, а значит и больше возможностей по получению ресурсов. Вычислительные центры, применяющие UNICORE, извлекут пользу от уменьшения количества документации для пользователей, усилий на поддержку их работы, а также смогут более эффективно использовать свои вычислительные ресурсы.

UNICORE дает возможность пользователю подготовить или изменить задания с использованием графического интерфейса на рабочей станции с платформой Unix или ПК с Windows. Задания могут быть запущены на любую из платформ UNICORE грид, и далее пользователь может осуществлять мониторинг и управление запущенными заданиями, используя часть интерфейса, называемую монитором заданий.

Задание UNICORE является структурированным и состоит из набора взаимосвязанных задач. Зависимости определяют временные соотношения между задачами (порядок их выполнения) или передачу данных. В настоящее время поддерживается: выполнение скриптов, компиляция, сборка, выполнение задач и директивы передачи данных. Ресурсный запрос определяет исполнительную систему, на которой будут выполняться задачи данного задания. Задачи могут быть сгруппированы в подзадания, создавая иерархическую структуру и позволяя разным шагам выполняться на разных системах в пределах грид UNICORE.

В условиях единой бесшовной среды задачи и ресурсы представляются в абстрактном виде. Перед выполнением заданий серверы UNICORE, транслируя абстрактные задания и ресурсные запросы в зависящие от платформы команды и опции, производят диспетчеризацию задач с учетом зависимостей между ними. Для каждой задачи входные и выходные файлы автоматически импортируются/экспортируются из/в файловое пространство пользователя или передаются от более ранних задач в том же задании. Реальная высокоскоростная передача данных между различными сайтами выполняется генерируемыми задачами, причем для каждой передачи серверы UNICORE подбирают наиболее эффективный механизм.

Для каждого задания пользователь определяет нужную исполнительную систему и требования к ресурсам для каждой задачи, входящей в данное задание. Пользовательский интерфейс проверяет, могут ли ресурсные запросы быть удовлетворены исполнительной системой, и разрешает запуск только в том случае, если запрос действительно может быть выполнен. Чтобы перезапустить задание на другой системе, пользователь просто меняет исполнительную систему. Пользователи могут следить за своими заданиями и управлять ими посредством монитора заданий, который графически отображает состояния.

Пользовательский интерфейс (клиент) UNICORE дает возможность пользователю создать, запустить и управлять заданием с любой рабочей станции или ПК в интернете. Клиент соединяется со шлюзом UNICORE, который аутентифицирует и клиента, и пользователя, после чего передает обращение серверам UNICORE, которые в свою очередь управляют запущенными заданиями. Сервер преобразует абстрактные задачи, предназначенные для локальных вычислительных узлов, в задания для пакетного режима и выполняют их на собственной пакетной подсистеме. Задачи, которые должны быть выполнены на удаленном сайте, пересылаются на тот же шлюз UNICORE. Все необходимые передачи данных и синхронизация осуществляются серверами также без участия пользователя. Они также хранят информацию о состоянии заданий и их выходных данных, пересылая ее клиенту по запросу.

ППО ARC

Основные архитектурные решения ARC следуют общепринятым подходам построения грид. Используется организация ресурсов, аналогичная применяемой в EU DataGrid. Узлами инфраструктуры служат кластеры, управляемые системами пакетной обработки, или отдельные компьютеры. Узлы комплектуются Элементами хранения (Storage Elements).

ARC обеспечивает следующие функции:

- информационная;
- включение ресурсов в грид и их мониторинг;
- запуск и управление заданиями;
- брокеринг (распределение заданий по ресурсам);
- управление данными и ресурсами.

Все функции реализованы в виде служб, которые опираются на известные программные средства с открытым кодом OpenLDAP, OpenSSL и SASL. Реализация выполнена с помощью библиотек Globus Toolkit 2 (GT2), безопасность достигается путем использования протоколов и инфраструктурных решений GSI.

Отличительной особенностью NorduGrid ARC является то, что хотя эта платформа и опирается на протоколы GT2 и реализована посредством API GT2, в ней предложен собственный набор

служб, заменяющий службы GT2. ARC не использует GRAM, утилиты управления заданиями, Gatekeeper и скрипты Job-manager, сервер Wuftp, схемы и информационные поставщики MDS. Для всего этого предложены собственные варианты:

- Grid Manager;
- gridftpd (ARC/NorduGrid GridFTP server);
- User Interface;
- Broker (“персональный” брокер, встроенный в пользовательский интерфейс);
- система мониторинга.

Кроме того, определена новая информационная схема и для нее разработаны поставщики данных, расширен язык описания ресурсов (xRSL). Таким образом, платформа ARC, хотя и построена на библиотеках GT2, она не сводится к GT2, а имеет свой набор служб.

На ресурсном узле работают следующие программы ARC: Grid Manager, gridftpd и информационные агенты. Служба Grid Manager локально запускает задание и контролирует процесс выполнения. Служба gridftpd осуществляет дистанционный прием заданий, создавая для каждого отдельный каталог на все время обработки. На узлах работают также поставщики информации о состоянии ресурсов, собирающие данные и передающие их индексным службам, которые представляют собой упрощенный вариант базы данных GT2 GIS и могут быть связаны в произвольную сетку: по связям происходит динамическое размножение информации. К индексным службам обращается пользовательский интерфейс (например, для выбора исполнительного узла) и Grid Manager.

Дистанционный запуск заданий производится посредством пользовательского интерфейса (UI), который представляет собой набор команд для запуска, мониторинга и управления заданиями, а также перемещения файлов и опроса информации о состоянии ресурсов. В состав UI входит Брокер, функция которого - подбор наиболее подходящего ресурса для задания. Другой специальный клиент - Grid Monitor через любой веб-браузер периодически опрашивает распределенную информационную систему, представляя результаты в виде взаимосвязанных страниц.

При разработке ARC преследовалась цель создания ПО производственного качества, используя в качестве основного принципа максимально полную децентрализацию, поэтому на каждом рабочем месте пользователя грид устанавливается персональный брокер, независимо подбирающий ресурсы для запускаемых заданий. Этот подход отличен от централизованной схемы EU DataGrid с единственным брокером на все рабочие места.

Как видно из приведенного обзора, развиваемое в рамках проекта EMI промежуточное программное обеспечение при существенных различиях в архитектуре и составе компонент имеет следующие общие черты:

1. при создании всех пакетов в качестве одного из важнейших требований декларировалось создание простого в эксплуатации и надежного ППО;
2. изначально в основе взаимодействия географически удаленных друг от друга компонент лежали специфические, разработанные для данного ППО, протоколы взаимодействия;
3. на более поздних этапах разработки создавалась специальная “обертка” исходных компонентов, превращающая эти компоненты в веб-сервисы; основной целью этой модернизации является унификация и возможность взаимодействия с сервисами, разработанными сторонними организациями;
4. создаваемые веб-сервисы следуют `стеку спецификаций/протоколов SOAP/WSDL/WS-*/WSRF.`

Первое из перечисленных выше требований совпадает с основной мотивацией для выполнения данной НИР. Причем как показывает опыт эксплуатации веб-сервисов на основе стека SOAP/WSDL/WS-*/WSRF (в частности, инструментария Globus Toolkit и gLite), и результаты тестовых испытаний в рамках данной НИР (раздел), грид/веб-сервисы на основе архитектурного стиля REST/JSON являются более простыми в реализации и обслуживании, а также более надежными в работе.

Достоинством подхода, развиваемого в рамках данной НИР является также то, что компоненты грид-системы с самого начала разрабатывались как веб-сервисы (без разделения на исходное “ядро” и веб-сервисную “обертку”). Это также способствует простоте и надежности работы ППО.

Важным остается вопрос унификации и возможности взаимодействия как с другими грид-системами в целом, так и отдельными сервисами, разработанными сторонними организациями и использующими другие протоколы и спецификации. Для исследования этого вопроса в рамках тестового полигона было организовано взаимодействие REST/JSON-сервиса выполнения многошаговых заданий с сервисом Grid Resources Allocation and Management (GRAM) из инструментария Globus Toolkit 4, выполняющего роль грид-шлюза к вычислительным ресурсам. Сервис GRAM (как и Globus Toolkit 4 в целом) является как бы реализацией стека спецификаций/протоколов SOAP/WSDL/WS-*/WSRF. Этот опыт показал, что организация такого взаимодействия не представляет серьезных проблем.

Таким образом, новаторский подход на основе архитектурного стиля REST, развиваемый в рамках данной НИР, имеет существенные преимущества по сравнению с проектом EMI, что в значительной степени объясняется историческими причинами - опорой проекта EMI на пакеты ППО, которые разрабатываются уже в течение значительного времени и унаследовали некоторые устаревшие подходы к архитектуре и принципам работы грид-систем.

From:

<https://theory.npi.msu.ru/> - **THEORY**

Permanent link:

<https://theory.npi.msu.ru/doku.php/dcomp/fcp/rest/emirest>

Last update: **26/01/2017 12:08**

