

Использование ресурсов европейской грид-инфраструктуры в рамках платформы Everest

О. В. Сухорослов, С. Ю. Волков

Институт проблем передачи информации им. А. А. Харкевича Российской академии наук,
127051, г. Москва, Большой Каретный переулок, д. 19, стр. 1

E-mail: sukhoroslov@iitp.ru

В работе рассматривается интеграция платформы Everest с европейской грид-инфраструктурой EGI. Everest - облачная платформа, реализующая публикацию, выполнение и композицию вычислительных приложений в распределенной среде. Платформа позволяет пользователям подключать к ней внешние вычислительные ресурсы различных типов и использовать данные ресурсы для запуска приложений, опубликованных на платформе. Интеграция с EGI позволила пользователям Everest удобным образом выполнять запуск расчетов на ресурсах грида через веб-браузер. В отличие от традиционных грид-порталов, рассматриваемый подход поддерживает новые сценарии использования, такие как комбинированное использование ресурсов грида и ресурсов других типов, композицию приложений с помощью REST API, а также автоматизацию проведения многовариантных расчетов.

Ключевые слова: распределенные вычисления, грид, облачная платформа, интеграция ресурсов, многовариантные расчеты.

Работа выполнена при поддержке Российского научного фонда (проект No 16-11-10352).

© 2016 Олег Викторович Сухорослов, Сергей Юрьевич Волков

Введение

European Grid Infrastructure (EGI)¹ является крупнейшей грид-инфраструктурой, объединяющий ресурсы около 300 вычислительных центров по всему миру для поддержки исследований в различных областях науки. Российские организации-участники EGI поддерживают национальный сегмент данной инфраструктуры в рамках консорциума RDIG (Russian Data Intensive Grid). Суммарный объем вычислительных ресурсов EGI составляет около 500 тысяч процессорных ядер. Доступ к ресурсам EGI регламентируется на основе участия пользователей грид в около 200 виртуальных организациях (ВО), посвященных различным областям прикладных исследований.

Базовым способом доступа пользователей к EGI является использование интерфейса командной строки (т.н. User Interface, UI), включающего набор команд для запуска заданий, отслеживания их статуса, работы с данными, получения информации о ресурсах и т.п. Однако данный способ является слишком низкоуровневым и неудобным для многих пользователей, особенно не имеющих опыта работы в командной строке. Альтернативным и более удобным способом работы с гридом является использование грид-порталов [Thomas, et al., 2005], реализующих предметно-ориентированные веб-интерфейсы для запуска в гриде определенного рода расчетов, пакетов и приложений. Как правило, данные порталы реализуются на базе UI или программных интерфейсов грида.

Несмотря на свое удобство, существующие грид-порталы зачастую не в полной мере удовлетворяют современным потребностям исследователей. В частности, они не позволяют задействовать для расчетов весь спектр ресурсов, доступных пользователю, куда, помимо ресурсов грида, относятся одиночные серверы, локальные кластеры, суперкомпьютеры, облачные сервисы. Также многие грид-порталы не предоставляют возможность публикации и композиции приложений пользователями, что важно с точки зрения поддержки автоматизации сложных составных расчетов (workflow). Кроме того, не всегда присутствуют удобные средства для автоматизации типовых расчетов, таких как многовариантные расчеты.

В данной работе рассматривается реализация доступа к ресурсам EGI в рамках платформы Everest. Лежащий в основе данной платформы подход позволяет в полной мере удовлетворить описанным выше требованиям.

Платформа Everest

Everest [Sukhoroslov, Volkov, Afanasiev, 2015] - веб-платформа для поддержки распределенных вычислений, разрабатываемая в ИППИ РАН. Платформа предоставляет пользователям готовые средства для публикации и совместного использования вычислительных приложений в виде сервисов. Everest также осуществляет выполнение приложений на внешних вычислительных ресурсах, подключенных пользователями. В отличие от традиционных систем распределенных вычислений, Everest реализует модель облачных вычислений Platform as a Service (PaaS), предоставляя доступ к своей функциональности через удаленные интерфейсы: пользовательский веб-интерфейс и программный интерфейс (REST API). Один экземпляр платформы может обслуживать много пользователей, позволяя им публиковать приложения, запускать их и делиться ими с другими пользователями без необходимости установки дополнительного ПО на свои компьютеры. Платформа публично доступна в режиме онлайн² для всех заинтересованных пользователей.

¹ <https://www.egi.eu/>

² <https://everest.distcomp.org/>

Everest поддерживает публикацию и выполнение следующих типов приложений. Существующие приложения с интерфейсом командной строки могут быть быстро опубликованы с использованием универсального шаблона прямо через веб-интерфейс платформы. Интерфейс прикладного программирования (API) платформы позволяет создавать сложносоставные приложения, включающие в себя запуски нескольких задач или других размещенных на Everest приложений в рамках произвольных расчетных схем. Данные приложения затем также могут быть опубликованы в виде сервисов. Кроме того, в рамках платформы реализован универсальный сервис для автоматизации проведения многовариантных расчетов [Volkov, Sukhoroslov, 2015], состоящих из большого числа вычислительных задач.

Вместо использования собственной вычислительной инфраструктуры, Everest выполняет запуск приложений на внешних ресурсах, подключенных к платформе пользователями. При этом пользователь может произвольным образом комбинировать имеющиеся в его распоряжении ресурсы для проведения расчетов. Первоначально была реализована интеграция платформы с одиночными ресурсами, такими как серверы, вычислительные кластеры и суперкомпьютеры. Метод интеграции основан на использовании специально разработанной программы, т.н. агента. Агент выполняется на стороне ресурса и играет роль посредника между платформой и ресурсом. Основными функциями агента являются управление выполнением задач на локальном ресурсе (включая загрузку входных данных, запуск задачи, мониторинг ее статуса и передачу полученных результатов), а также предоставление информации о характеристиках и текущем состоянии ресурса. Описанная в главе 3 интеграция Everest с инфраструктурой EGI сделала возможным использование ресурсов грида (как по отдельности, так и в сочетании с другими ресурсами) для запуска приложений в рамках платформы.

Интеграция платформы Everest с грид-инфраструктурой EGI

Интеграция Everest с EGI заметно отличается от интеграции с одиночными ресурсами в силу следующих дополнительных проблем. Во-первых, поскольку сейчас многие пользователи грида не пользуются напрямую UI, то не всегда возможно или удобно запускать агента на машине с UI аналогично интеграции с кластерами. Во-вторых, поскольку доступ к ресурсам EGI регулируется на основе членства пользователя в ВО, то для запуска задания в гриде платформе требуется действующий прокси-сертификат с атрибутами соответствующей ВО. В-третьих, для грида характерны высокие и зачастую непредсказуемые задержки при планировании заданий, например некоторые задания могут подолгу стоять в очередях отдельных ресурсов.

Исходя из описанных особенностей, для интеграции с EGI был использован следующий подход (Рис. 1). Ранее разработанные агенты динамически отправляются в виде заданий в грид через сконфигурированный экземпляр UI, развернутый на серверной стороне платформы. Будучи запущенным на рабочем узле грида агент подключается к Everest и приступает к выполнению назначенных ему платформой вычислительных задач на локальном узле. При этом взаимодействие агента и платформы не отличается от случая одиночного ресурса. Однако агентов теперь запускает платформа, а не пользователь, и с одним ресурсом в Everest теперь может быть связано сразу несколько агентов. Кроме того, в случае отсутствия новых задач, агент автоматически завершает свое выполнение, чтобы освободить занимаемый ресурс грида. Описанный подход основан на известной стратегии *pilot jobs* [Sfiligoi, et al., 2009], используемой для устранения задержек при планировании заданий в грид. В данном случае в качестве пилотных заданий используются имеющиеся агенты.

Подключение ресурсов EGI к платформе реализовано на уровне отдельных ВО. Пользователь Everest может подключить к платформе ресурсы определенной ВО, передав платформе действующий прокси-сертификат. Прокси-сертификат генерируется пользователем на основе его персонального сертификата при помощи стандартной утилиты `grid-proxy-init` (входит в состав UI) или предоставляемого кроссплатформенного приложения на Java. Перед запуском заданий в гриде Everest с помощью UI формирует расширенный прокси-сертификат с VOMS-

расширениями в соответствии с конфигурацией используемой ВО. Каждая подключенная ВО представлена в Everest в виде отдельного ресурса, который может быть использован для запуска приложений точно так же, как одиночные ресурсы. Отметим, что одной ВО, подключенной к платформе двумя различными пользователями, в рамках Everest будут соответствовать два независимых ресурса. По истечении срока действия прокси-сертификата соответствующий ресурс становится недоступным до тех пор, пока пользователь не обновит сертификат.

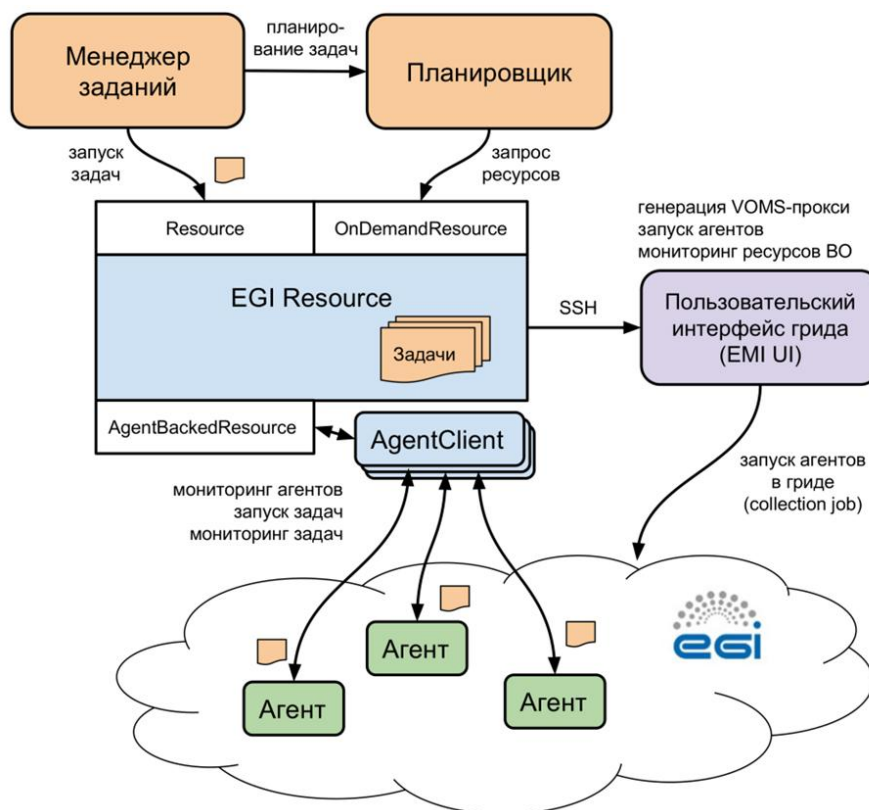


Рис. 1. Схема интеграции Everest с грид-инфраструктурой EGI

Число агентов, выполняемых в гриде, для каждого связанного с EGI ресурса контролируется планировщиком Everest. В случае, если в конце цикла планирования остались нераспределенные задачи, связанные с данным ресурсом, то планировщик запрашивает выделение соответствующего числа новых агентов. Агенты отправляются в грид в виде одного пакетного задания (collection job) с тем, чтобы минимизировать накладные расходы на отправку. В качестве дополнительной оптимизации для каждого агента запрашивается сразу несколько ядер на рабочем узле грида. Для борьбы с задержками при планировании заданий в гриде платформа автоматически повторяет отправку агентов в случае, если какие-то агенты не были запущены в течение заданного времени. Максимальный объем ресурсов, выделяемых в гриде, определяется лимитом на число агентов, задаваемым пользователем. Освобождение ресурсов грида реализуется путем автоматической остановки агентов, которые бездействовали в течение пяти минут.

Пример использования

В качестве примера использования EGI в рамках Everest рассмотрим запуски реальных многовариантных расчетов на пулах из одиночных ресурсов (сервер с 12 ядрами и кластер с 216 ядрами) и виртуальных организаций EGI. Первый расчет состоял из 1000 вычислительных задач, выполняющих виртуальный скрининг молекул с помощью Autodock Vina. Для данного

расчета использовалась ВО BIOMED, посвященная исследованиям в области биомедицины. Второй расчет состоял из 670 задач, выполняющих табуляцию сложной многомерной функции для решения обратной задачи геофизики. В данном расчете использовалась ВО ESR, посвященная исследованиями в области наук о Земле. Оба расчета запускались через универсальный сервис Parameter Sweep [Volkov, Sukhoroslov, 2015], доступный в Everest.

На Рис. 2 изображены графики числа выполнявшихся задач на используемых ресурсах в течение данных расчетов. Из графиков видно, что платформа смогла полностью загрузить одиночные ресурсы и задействовать дополнительные ресурсы из грида. Периодические спады загрузки кластера связаны с завершением задач волнами и задержками при мониторинге и планировании задач на уровнях планировщика кластера и Everest. Ступенчатый вид графика загрузки ВО связан с задержками планирования в гриде и особенностями механизма выделения ресурсов, описанными в главе 3. Кроме того, во время первого расчета был достигнут заданный лимит на объем ресурсов, выделенных в гриде (100 ядер).

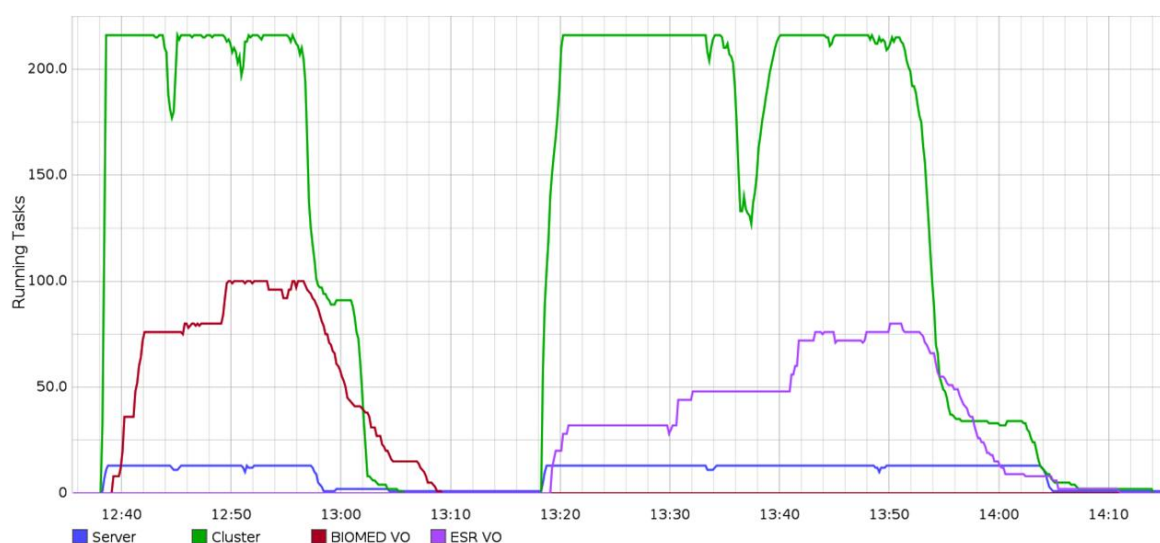


Рис. 2. Число выполнявшихся задач на различных ресурсах в течение расчетов

Для оценки эффекта от использования пула ресурсов при помощи Everest в сравнении с использованием одного ресурса, данные расчеты также были выполнены на одиночном сервере. Использование Everest позволило уменьшить время первого расчета с 9 часов до 30 минут, а второго - с 14,5 часов до 1 часа.

Список литературы

- Thomas M.P. et al.* Grid portal architectures for scientific applications // *Journal of Physics: Conference Series*. — IOP Publishing, 2005. — Vol. 16, No. 1. — P. 596.
- Sukhoroslov O., Volkov S., Afanasiev A.* A Web-Based Platform for Publication and Distributed Execution of Computing Applications // *14th International Symposium on Parallel and Distributed Computing (ISPDC)*. — IEEE, 2015. — P. 175–184.
- Volkov S., Sukhoroslov O.* A Generic Web Service for Running Parameter Sweep Experiments in Distributed Computing Environment // *Procedia Computer Science*. — 2015. — Vol. 66. — P. 477–486.
- Sfiligoi I. et al.* The pilot way to grid resources using glideinWMS // *Computer Science and Information Engineering, 2009 WRI World Congress on*. — IEEE, 2009. — Vol. 2. — P. 428–432.

Using EGI Resources with Everest Platform

O. V. Sukhoroslov, S. Yu. Volkov

Institute for Information Transmission Problems of the Russian Academy of Science, Kharkevich Institute,
19, build.1, Bolshoy Karetny per., Moscow, 127051, Russia

E-mail: sukhoroslov@iitp.ru

The report discusses integration of Everest platform with European Grid Infrastructure (EGI). Everest is a cloud platform supporting publication, execution and composition of computing applications in a distributed environment. The platform allows users to attach different types of external computing resources and use these resources in order to run applications published on Everest. The integration with EGI enabled Everest users to seamlessly run applications on grid resources via a web browser. In contrast to classic grid portals, the presented solution supports additional use cases such as combined use of EGI and other resources, composition of applications via REST API and running of parameter sweep experiments.

Keywords: distributed computing, grid, cloud platform, resource integration, parameter sweep.

This work was supported by the Russian Science Foundation (project No. 16-11-10352).

© 2016 Oleg V. Sukhoroslov, Sergey Yu. Volkov