Ордена Ленина

ИНСТИТУТ ПРИКЛАДНОЙ МАТЕМАТИКИ
имени М.В.Келдыша
Российской академии наук

В.Н. Коваленко, Е.И. Коваленко, Д.А. Корягин,
Э.З. Любимский, А.В. Орлов, Е.В. Хухлаев

Структура и проблемы развития программного обеспечения среды распределенных вычислений Грид

Москва
2002 г.

УДК 519.68

В.Н. Коваленко, Е.И. Коваленко, Д.А. Корягин, Э.З. Любимский, А.В. Орлов, Е.В. Хухлаев. **Структура и проблемы развития программного обеспечения среды распределенных вычислений Грид.**

Работа посвящена вопросам создания Грид – глобально распределенной инфраструктуры, которая обеспечивает безопасное и скоординированное разделение ресурсов в рамках виртуальной организации. Рассмотрена послойная организация программного обеспечения Грид, определены нерешенные проблемы и намечены возможные направления дальнейших исследований.

Ключевые слова: Грид, распределенные вычисления, архитектура программного обеспечения.

V.N. Kovalenko, E.I. Kovalenko, D.A. Koryagin, E.Z. Ljubimskii, A.V. Orlov, E.V. Huhlaev. **Grid software: its structure and development problems.** – Preprint, Keldysh Inst. Appl. Mathem., Russian Academy of Science, 2002.

This work is devoted to the “Grid problem” - globally distributed infrastructure which provides the safe and coordinated resource sharing within the framework of the virtual organization. The layered organization of the Grid software is considered, unsolved problems are determined and possible directions for the further researches are scheduled.

Key words and phrases: Grid, distributed computations, software architecture.

Работа выполнена при финансовой поддержке Российского фонда фундаментальных исследований (проект № 02-01-00282)

Содержание

1. Введение

В настоящее время во многих странах мира – США, Великобритании, Франции, Германии, Италии, Польше, Индии, Японии, Корее, Австралии и других – развернуты проекты по созданию программно-телекоммуникационной инфраструктуры, цель которой – обеспечить доступ к разнообразным вычислительным ресурсам независимо от места расположения потребителя. Термин инфраструктура употреблен здесь не случайно – имеет место достаточно полная аналогия с более привычными глобальными ее видами, такими, например, как электрическая, железнодорожная сети или почтовая служба. Неудивительно, что лежащие в основе подхода технологии получили название Грид (Grid) – энергетическая система, только вместо энергии потребитель получает ресурсы обработки данных.

Само развитие тематики Грид представляет собой интересный феномен: фактически сейчас трудно говорить, что есть законченный набор технологий Грид, который можно было бы внедрить в какой-нибудь практической сфере, например, в научных исследованиях – потенциально первой и наиболее очевидной области применения. Однако приведенная выше география проектов показывает, насколько высоко оценивается потенциал Грид: он имеет стратегический характер, и в близкой перспективе Грид должен стать вычислительным инструментарием для развития высоких технологий в различных сферах человеческой деятельности, подобно тому, как подобным инструментарием стали персональный компьютер и Интернет.

2. Концепция Грид в современном понимании

Концепция Грид зародилась в контексте важной, но, как оказалось впоследствии, относительно более узкой проблемы построения сверхмощных вычислительных установок. В середине 80-х годов основными для этой области были суперкомпьютерные технологии, успешность которых во многом определялась стремительным прогрессом микропроцессорной техники. В 1985 году в США была принята – и в последующее десятилетие реализована – общенациональная программа по созданию суперкомпьютерных центров, финансируемая государством через Национальный научный фонд (NSF - National Science Foundation). Полученный опыт оказался не только положительным: появилось понимание, что при высокой цене разработки и производства суперкомпьютеров, построенные архитектуры имеют ограниченную масштабируемость и не успевают за развитием элементной базы. В то же время, проведенные прикладные исследования показали, что для решения ряда насущных и наиболее приоритетных задач методами математического моделирования (прогнозирование природных явлений, обработка данных высокоэнергетических ядерных реакций, эволюция звезд) необходимы вычислительные мощности принципиально нового уровня производительности и быстродействия.

В начале 90-х годов столь же бурно, как и микропроцессоры, стали развиваться телекоммуникационная аппаратура и линии передачи. Идея объединения процессорных технологий с телекоммуникационными дала толчок Метакомпьютингу, вначале как способу соединения суперкомпьютерных центров: термин Метакомпьютинг появился в CASA [1] - проекте одной из экспериментальных гигабитных телекоммуникационных сетей. В статье [2], которая в дальнейшем стала основополагающей, Метакомпьютинг определяется как “использование мощных вычислительных ресурсов, доступных прозрачно посредством телекоммуникационной среды”. В дополнении к условию прозрачности применимы также такие характеристики как бесшовность, масштабируемость и глобальность. Таким образом, в новой парадигме Метакомпьютинга предлагалось полностью скрыть наличие телекоммуникаций и использовать подключенные к сети компьютеры как единый объединенный вычислительный ресурс. Основной акцент в Метакомпьютинге делался на то, что потенциальный пользователь может получить практически неограниченные ресурсы для вычислений и хранения данных. Весь вопрос в том, как подобные распределенные ресурсы запрячь в архитектуру Метакомпьютера.

Начатые в этом направлении работы – в первую очередь по системам Globus [3,4] (совместный проект Аргоннской национальной лаборатории ANL при университете Чикаго и института информатики университета Южной Каролины ISI USC) и Legion [5,6] (университет Вирджинии), а также ряд других - привели к существенному обобщению идеи Метакомпьютинга. Уже на начальном периоде развития было показано, что для программной поддержки распределенной среды необходимо решить широкий круг проблем: связи, безопасности, управления заданиями, доступа к данным, информационного обеспечения. Все эти вопросы имеют прямые аналоги в операционных системах, но должны быть пересмотрены для ненадежной, открытой и распределенной глобальной среды. Более того, архитектура (теперь уже можно говорить о Грид) среды должна быть расширяемой и способствующей наращиванию функциональности при сохранении работоспособности. По-видимому, именно последнее обстоятельство привело к современной трактовке понятия Грид, которое в [7] определяется следующим образом:

> Грид является согласованной, открытой и стандартизованной средой, которая обеспечивает гибкое, безопасное, скоординированное разделение ресурсов в рамках виртуальной организации – то есть динамически формирующейся совокупности независимых пользователей, учреждений и ресурсов.

Первое, что обращает внимание - речь больше не идет о “**мощных** вычислительных ресурсах” Метакомпьютинга. В качестве процессорных ресурсов рассматриваются теперь, например, рабочие станции и ПК. На самом деле, основные вычислительные мощности сосредоточены вовсе не в

суперкомпьютерном парке. Если организация располагает, скажем, тремя тысячами рабочих мест на базе рабочих станций, то за время их регулярного простоя потерянные циклы составят существенную долю даже терафлопной производительности. Мощные ресурсы - суперкомпьютеры, кластеры, SMP-системы - остаются важным частным случаем. Кроме того, новая трактовка применима к разнообразным типам ресурсов: телекоммуникациям, системам массовой памяти, хранилищам данных, а также измерительным и научным инструментам, например, радиотелескопам.

Выход за рамки высокопроизводительных систем и приложений выявляет реальное содержание Грид: это инфраструктура для поддержки любой глобально распределенной вычислительной деятельности. От инфраструктуры Грид может извлечь пользу множество типов приложений – это электронный бизнес, кооперативное проектирование, исследование данных, системы обработки высокой пропускной способности (High Throughput Computing – HTC), и, конечно, распределенный суперкомпьютинг (то есть Метакомпьютинг). Для многих из этих приложений, в том числе и с большим объемом вычислений, но с “хорошими” свойствами (грубо гранулированных, конвейеризуемых), не требуются высокопроизводительные телекоммуникации как для Метакомпьютинга. Тогда в качестве телекоммуникационной составляющей инфраструктуры Грид может выступать обычный Интернет - неограниченно масштабируемый, всеобъемлющий и повсеместный уже сейчас.

Проиллюстрируем реализацию этого подхода на примере National Grid – проекта правительства Великобритании, курируемого Министерством науки и технологий. В первую очередь он направлен на поддержку кооперативных научных исследований по широкому спектру дисциплин. Кроме того, этот проект служит своеобразным испытательным полигоном для развертывания "e-utility computing", также известного как "e-sourcing" – предоставления компьютерных ресурсов (таких, как пропускная способность, приложения, дисковая память) по Интернет в качестве своеобразной разновидности коммунальных услуг.

Национальный центр сети National Grid расположен в Эдинбурге/Глазго; кроме него будут построены восемь региональных центров в Оксфордском и Кембриджском университетах, в университетах Ньюкасла, Белфаста, Манчестера, Кардиффа, Саутгемптона и в Империал Колледж, Лондон.

В качестве поставщика ключевых технологий и инфраструктуры выбрана компания IBM - она выиграла тендер на создание центра хранения данных в Оксфордском университете, который станет основным в Великобритании источником информации по физике высоких энергий, полученной в ходе экспериментов в лаборатории Ферми (Батавия, штат Иллинойс, США). У IBM есть собственный опыт в области Грид: на основе системы Globus подразделение IBM Research создало собственный Грид – географически распределенный суперкомпьютер, объединяющий исследовательские и

проектно-конструкторские лаборатории IBM в США, Израиле, Швейцарии и Японии.

3. Программная составляющая инфраструктуры Грид

Настоящая работа направлена на анализ современного состояния программной составляющей Грид и определению возможных направлений ее развития. Стоит оговорить, что при этом за рамками остаются другие не менее важные вопросы инфраструктуры – телекоммуникации и вычислительная база Грид.

Можно утверждать, что базовым программным обеспечением Грид и международным стандартом де-факто является на сегодня система Globus. Это, во-первых, признано [8] ведущими компаниями мировой компьютерной индустрии: IBM, Microsoft, Compaq, Cray, SGI, Sun Microsystems, Fujitsu, Hitachi, NEC, Veridian, Entropia, Platform Computing Inc. Во-вторых, Globus реально взят за основу в ведущих проектах Грид: IPG [9], NCSA [10], Gryphyn [11], DataGrid [12]. В-третьих, большая часть новых исследований и разработок в области Грид ориентируется на Globus. В дальнейшем изложении мы будем также исходить из возможностей системы Globus.

4. Базовая архитектура программного обеспечения Грид

Грид – это распределенная среда, и ее функционирование обеспечивается специальной формой программного обеспечения (ПО) – сервисами. Сервисы обладают сетевым интерфейсом, благодаря чему становится возможным удаленное обслуживание клиентов. В отличие от модели “клиент-сервер” в Грид тот или иной набор сервисов устанавливается на каждом ресурсе, хотя традиционное серверное обслуживание также не исключается.

Для кооперативной деятельности множество сервисов должно удовлетворять двум структурным условиям:

- каждый тип сервиса должен иметь стандартный протокол доступа, в соответствии с которым реализуется прикладной интерфейс (API) клиентов. В рамках стандартных протоколов допустимы различные способы реализации сервисного обслуживания;
- множества сервисов на разных ресурсах должны быть согласованными. Это предполагает известную унификацию наборов сервисов на основе тождественности их семантики, а также наличие общих правил, регламентов и организационных соглашений, на которые опирается конфигурирование сервисов.

Успех проекта Globus во многом связан с тремя входящими в его состав сервисами и соответствующими протоколами:

- Протокол доступа к ресурсам и управления (Grid Resource Allocation and Management - GRAM) и сервис Gatekeeper, которые

обеспечивают безопасное, надежное создание удаленных процессов и управление ими [13];

- Сервис Метадиректорий [14] для распределенного сбора данных и информационного обслуживания;
- Сервисы инфраструктуры безопасности (Grid Security Infrastructure - GSI), поддерживающие однократную регистрацию, делегирование полномочий и отображение прав доступа в различные локальные системы [15].

Однако, как стало понятно сейчас, состав сервисов со временем должен стать гораздо богаче. Поэтому ПО Грид должно бы опираться на унифицированную архитектуру сервисов, поддержанную средствами их подключения, регистрации, мониторинга и рядом других. Пока такой архитектуры нет, реализация каждого нового сервиса требует обращения к протоколам самого низкого уровня.

Эта проблема рассматривается как центральная для проекта Globus. В работе [16] опубликованы предложения, в которых новая архитектура названа OGSA - Open Grid Services Architecture.

При всей важности общеархитектурных вопросов возможности Грид главным образом определяются составом ПО. По сложившемуся представлению [7] программное обеспечение делится на 4 слоя: 1) адаптации ресурсов, 2) связи, 3) доступа к ресурсам, 4) кооперации, к которым мы добавим еще один - 5) координации.

5. Слой адаптации ресурсов

Слой адаптации является той частью ПО Грид, которая работает на ресурсах и представляет их для использования вовне. Под ресурсами понимаются самые разные элементы, используемые в обработке данных: файловая система, многопроцессорная установка, телекоммуникации, пул компьютеров или даже датчик. Первая задача этого слоя – унификация ресурсов и представление их в виде абстрактных типов со стандартизованным множеством операций.

Вторая задача связана с тем, что набор операций, который непосредственно поддерживается базовым обеспечением ресурсов, недостаточен (или неэффективен) для работы в распределенном варианте Грид. Поэтому, во-первых, слой адаптации вводит необходимые дополнительные средства локального управления ресурсами. Для вычислительных комплексов это пакетные системы (PBS, Condor, LSF, Sun Grid Engine и т.д.). Во-вторых, ввиду разнообразия систем управления проводится опять же их унификация.

Необходимо подчеркнуть, что функциональные возможности вышележащих (над данным) слоев в большой степени определяются тем множеством операций, которые реализованы в слое адаптации. Современное состояние локального управления ресурсами оставляет желать много лучшего

как с точки зрения качества реализации, так и с точки зрения богатства набора операций. Хотя средства локального управления формально не входят в номенклатуру ПО Грид, их развитие должно осуществляться параллельно и в тесной связи с остальными вопросами тематики Грид. Далее приводится современный минимум операций для различных типов ресурсов и необходимые расширения.

- Вычислительные ресурсы.

Поддерживаются системами пакетной обработки.
Реализованные операции: 1) запуск/снятие/мониторинг заданий; 2) опрос характеристик оборудования (платформы обрабатывающих узлов, операционные системы), динамического состояния ресурсов (текущей загрузки машин, свободного файлового пространства) и состояния системы управления (характеристики и состояния заданий).
Необходимые расширения:

- Средства выделения ресурсов, в частности, механизм резервирования. На него опирается слой кооперации: без резервирования невозможен запуск параллельных заданий. Один из вариантов резервирования реализован в планировщике Maui [17], который может использоваться вместе с наиболее употребительными в Грид системами пакетной обработки.
- Средства мониторинга оборудования.
- Для эффективного планирования распределения заданий в Грид (см. слой координации) требуются средства контроля за происходящими в системе управления событиями (освобождение ресурсов, запуск/завершение заданий) и средства получения информации, на основе которой может предсказываться ход обработки заданий.

- Ресурсы внешней памяти.

Поддерживаются в основном файловыми системами ОС. Системы массовой памяти снабжаются дополнительными патентованными пакетами ПО от производителей.
Необходимые расширения:

- Высокопроизводительная передача данных, на основе многопоточности [18].
- Оптимизация чтения/записи фрагментов файлов.
- Возобновляемая передача файлов.
- Передача файлов с фильтрацией и редуцированием содержания [19].
- Управление локальными ресурсами, которые используются для передачи данных: оперативной памятью под буферы обмена, шириной сетевой полосы, процессором.
- Опрос состояния: общей емкости, свободного пространства, гарантированной скорости передачи, временной задержки.

- Предварительное резервирование ресурсов и управление квотами памяти.
- Прозрачные интерфейсы для подключения локальных ресурсов к глобальной файловой системе.

- Сетевые ресурсы.

Поддерживаются протоколами слоя связи.
Необходимые расширения:

- Управление сетевым трафиком на основе приоритизации и резервирования.
- Средства опроса характеристик сети и текущей загрузки.

- Каталоги.

В среде Грид каталоги используются для хранения информации о составе, характеристиках и состоянии ресурсов. Практически во всех локальных системах управления ресурсами реализованы частные системы хранения, подчас не имеющие открытых внешних интерфейсов. В качестве унифицированного информационного интерфейса Грид применяется протокол распределенных директорий LDAP [20]. Этот протокол рассчитан на поддержку иерархической модели данных, между тем как в приложениях Грид необходимы каталоги с более развитой информационной структурой.
Необходимые расширения:

- Унифицированные протоколы для поддержки баз данных (БД) различных типов - реляционных и объектных.
- Средства управления схемами БД в условиях быстро растущего числа типов данных.
- Повышение эффективности поиска и обновления информации.
- Поддержка сложных поисковых запросов по нескольким связанным объектам.
- Хранение и обработка массивов однородных данных.

- Хранилища программных кодов.

Поддержка управления программными кодами в операционных системах осуществляется утилитами типа Configure и Make.
Необходимые расширения: для гетерогенной среды Грид нужна система управления зависящими от платформы версиями исходного, объектного и исполняемого кода. На этой основе может быть построена система мобильной, то есть рассчитанной на разные платформы, подготовки заданий и программ. Для локальной среды сетевого кластера с распределенной файловой системой такого рода средство (пакет Metamake) предложено в [21].

6. Слой связи

Слой связи объединяет протоколы коммуникации и безопасности, образуя унифицированную базу сетевых транзакций для всех вышележащих слоев ПО Грид.

Протоколы коммуникации обеспечивают передачу данных, маршрутизацию и именование. В настоящее время эти протоколы основаны на стандартном стеке TCP/IP: транспортный уровень (TCP, UDP), уровень Интернет (IP, ICMP), прикладной уровень (DNS, OSPF, RSVP, и т.д.). В рассматриваемом далее слое кооперации ПО Грид уже возник ряд задач: массовая рассылка сообщений, резервирование пропускной полосы, ранжирование потоков по приоритетам, - для которых, по-видимому, потребуются протоколы нового поколения типа IPv6.

Протоколы безопасности, составляющие Инфраструктуру безопасности Грид (GSI), надстроены над коммуникационными. Решаются задачи аутентификации, защиты сообщений и авторизации. Реализация протоколов безопасности выполнена в виде расширения протокола TLS (Transport Layer Security) [22] и основана на криптографических алгоритмах и технологии открытых ключей. Для идентификации пользователей и ресурсов используются сертификаты стандарта X.509 [23]. Управление авторизацией осуществляется посредством интерфейса Generic Authorization and Access (GAA) [24]. Этот интерфейс позволяет интегрировать в инфраструктуру Грид различные локальные политики безопасности, основанные, например, на текстовых паролях или системе Kerberos [25].

Протоколы безопасности удовлетворяют следующим необходимым требованиям:

- однократная регистрация пользователя в Грид;
- делегирование полномочий программам и сервисам, выполняющимся от имени пользователя.

Необходимые расширения:

- Для поддержки GSI требуется программно-организационная инфраструктура управления сертификатами: иерархия сертификационных центров выдачи, обновления и отзыва сертификатов (самая тяжелая проблема), однако соответствующего стандартизованного ПО нет.
- Имеется существенный недостаток в современном способе авторизации. Пользователь должен быть зарегистрирован в операционной системе каждого доступного ему компьютера и прописан в специальном конфигурационном файле (Gridmap-file) ресурса. Для открытой и масштабной Грид этот способ представляется неудовлетворительным и требует развития.

7. Слой доступа к ресурсам.

Слой доступа, надстроенный над слоем связи, определяет ряд протоколов и программных интерфейсов (API), которые делают возможным использование ресурсов Грид повсеместно. С помощью средств этого слоя производится поиск ресурсов, а также дистанционная инициация, мониторинг и управление операциями. В отличие от кооперативного, слой доступа ограничен возможностью работы с индивидуальными ресурсами – без какого-либо учета глобального состояния Грид.

В слое реализованы два типа протоколов: информационные и управляющие.

Два **информационных** протокола [14] базируются на LDAP (Lightweight Directory Access Protocol). Сервисы первого из них - GRIP (Grid Resource Information Protocol) устанавливаются на каждом ресурсе и собирают данные о его характеристиках (конфигурация, платформа) и состоянии (текущей нагрузке). Информационная модель GRIP расширяема и позволяет, в принципе, представлять произвольные данные. Распределенная модель поддерживается вторым протоколом - регистрации ресурсов GRRP (Grid Resource Registration Protocol). Посредством GRRP сведения о наличии и местоположении GRIP сообщаются серверу GIIS (Grid Index Information Server), на который впоследствии подкачиваются данные со всех зарегистрированных серверов GRIP.

Направления развития:

Недостатки реализованного подхода обусловлены тем, что базовый протокол LDAP ориентирован на работу с медленно меняющейся и слабо структурированной информацией. Так, язык запросов LDAP не может дать результат при необходимости вычислений на двух разных объектах в информационной схеме, или выражаясь на реляционном языке, когда нужна операция объединения (join). В рамках схемы GRIP – GRRP протокол GRIP должен:

- поддерживать информационную модель с иерархией типов данных и возможностями их связывания;
- допускать использование в качестве каталогов реляционных баз данных;
- предусматривать язык запросов, соответствующий потребностям слоев кооперации и координации, поддерживающий, в частности, составные запросы.

Поясним последний пункт на примере запроса по поиску ресурсов при запуске задания. Сейчас языковый запрос (полностью унаследованный от LDAP) позволяет записать требования к ресурсам в виде пар отношений <имя ресурса> - <минимальное допустимое значение>. Во-первых, должно быть стандартизовано множество типов ресурсов (по-видимому, оно должно быть расширяемым). Во-вторых, язык должен позволять описывать временную последовательность использования ресурсов (доставка файлов - вычисления – сетевой обмен). В-третьих, должны быть обеспечены способы описания параллельных заданий, для которых нужна совокупность ресурсов.

Нуждается в совершенствовании и модель распределенного хранения GRRP – сейчас реализована лишь полная интеграция локальных информационных баз на индексный сервер, между тем как LDAP поддерживает распределенный поиск без физической интеграции данных.

Не решен вопрос наполнения информационных баз. В соответствии с архитектурой наполнение (и обновление) должно происходить автоматически программами-поставщиками статусной информации. Реально эти программы (относящиеся к слою адаптации ресурсов) разработаны только для вычислительных ресурсов, причем, не вполне ясно, те ли это данные - можно ли по ним выбирать ресурсы.

Управляющие протоколы позволяют удаленно выполнять операции на ресурсе, такие, например, как запуск процесса или передача файла.

- Для вычислительных ресурсов реализован протокол GRAM (Grid Resource Access and Management), базирующийся на протоколе HTTP. Он позволяет:
  - запустить/снять задание, создать для него программную среду,
  - получить информацию о статусе задания (ожидание, выполнение),
  - доставить выходной поток данных выполняющейся программы удаленному пользователю.
- Для ресурсов внешней памяти. В стадии реализации находится протокол передачи файлов GridFTP [26]. Он является серьезным шагом вперед в сравнении с предшествовавшими средствами, обеспечивая безопасность в соответствии со слоем связи, доступ к частям файлов, параллельную многопотоковую передачу, отдельные каналы для управления и передачи данных, возобновляемость.

Необходимые расширения:

- Расширение номенклатуры управляемых ресурсов.
- Введение механизмов для расширения состава функций, в частности требуются функции выделения ресурсов и их предварительного резервирования.
- Перестройка протоколов в соответствии с общей архитектурой сервисов OGSA.

8. Слой кооперации

Кооперативный слой, строится над слоем удаленного доступа, но, в отличие от последнего, позволяет взаимодействовать не с индивидуальным ресурсом, а с их совокупностью. На этом уровне Грид рассматривается уже как организованная среда. К этому уровню можно отнести следующее ПО.

- *Сервис директорий* (GIIS) содержит информацию обо всех ресурсах Грид. Таким образом, для поиска ресурсов с нужными свойствами достаточно направить запрос в одну точку, а не опрашивать каждый ресурс по отдельности. Способ наполнения сервера GIIS, описанный в предыдущем разделе, заключается в периодическом обновлении информации путем

опроса зарегистрированных по протоколу GRIP сервисов ресурсов. Последняя версия (MDS-2) информационной архитектуры Globus рассмотрена в работе [14].

- *Сервис коаллокации*. Применяется для локализации и заказа глобально распределенного комплекса ресурсов. Этот сервис нужен для запуска больших параллельных заданий. Попытка его реализации была предпринята в системе Globus (комплекс Duroc), но заложенные в нем принципы оказались не работоспособными. Пока это открытая тема, решение которой упирается в два вопроса. Первое, нужна поддержка резервирования в слоях адаптации ресурсов и доступа к ресурсам. Второе, нужно расширение языка запросов, на основе которого можно бы было выбирать оптимальное множество ресурсов с точки зрения эффективности выполнения распределенного приложения.
- *Сервис брокеров*. Слой доступа к ресурсам содержит сервисы для запуска заданий, проверки статуса, доставки выходных данных. От сервиса брокеров ожидаются дополнительные возможности:
  - Для поиска ресурсов брокер использует один или несколько серверов GIIS. При поиске учитывается доступность ресурса для данного пользователя.
  - Обеспечивается надежный запуск заданий: если задание прервалось по не зависящим от него причинам (сбой машины или сети), оно запускается заново.
  - Ведется протокол запуска заданий, доступный как владельцу задания, так и администраторам ресурсов.
  - Производится доставка файлов на исполнительный ресурс, при этом учитывается репликация файлов.
  - Производится выделение и резервирование ресурсов.

  Работы по брокерам были начаты в смежных с Грид областях. Упомянем AppLeS [27], Condor-G [28], Nimrod-G [29] и DRM [30], однако каждый из этих брокеров направлен на решение отдельного вопроса, например, AppLeS - на оптимизацию использования ресурсов в отдельном задании, а, кроме того, все они слабо интегрированы в среду Грид. В нашем проекте [31] и системе GRB (Grid Resource Broker) [32] проекта DataGrid решается большинство из перечисленных выше вопросов, но уже на базе сервисов Грид.

  Брокеры - то есть агенты, посредничающие между заданием и ресурсами – рассчитаны на поиск свободных ресурсов. В ситуации, когда ресурсы Грид загружены, полезность брокеров имеет ограниченный характер – дополнительно требуется поддержка очередей и динамическое распределение заданий по освобождающимся ресурсам, что составляет функции диспетчеров (Scheduler), рассматриваемых в следующем разделе. Тем не менее, все компоненты брокеров сохраняют ценность и используются в диспетчерах.

- *Сервис мониторинга и диагностики*. Функционирование распределенных систем типа Грид опирается на разнообразные данные о состоянии компонентов, которые затем используются в различных задачах: обнаружения сбоев, анализа производительности, распределения загрузки и т.п. Информационные системы общего назначения (базы данных и сервисы директорий) плохо подходят для распределенного мониторинга, ввиду природы самих данных. Статусные данные мониторинга имеют ограниченное и, как правило, короткое время жизни (после чего они становится недостоверными). Поэтому частота их обновлений должна быть высокой, в то время как обычные БД оптимизируются на запросы, а не на обновления. В информационной системе мониторинга должна обеспечиваться низкая задержка при передаче от точки получения данных к точке, где они хранятся. В свою очередь, принимающая сторона должна выдерживать высокую скорость приема, обусловленную частыми обновлениями.

  Архитектура с такими свойствами предложена в [33]. Суть предложения заключается в том, чтобы разделить сбор данных и операции поиска. Данные мониторинга хранятся распределенно – там же где и производятся. Так как суммарный объем данных очень большой, задержки при поиске по всему информационному массиву будут непредсказуемы. Поэтому предлагается адресовать поисковые запросы “реестру матаданных”, который представляет собой индекс распределенного хранения и позволяет определить источник требуемых данных. Далее запрос переадресуется в место хранения и там производится уже более узкий поиск. Имеется реализация этого подхода (R-GMA [34]).
  Таким образом, представление об информационной службе Грид существенно уточняется: рассматривается комплексная задача производства (программного), хранения и извлечения данных. Однако ряд вопросов остаются нерешенными:
  - Нуждается в развитии содержательная сторона мониторинга – приложений, которые работают с собираемой информацией, практически нет.
  - С появлением приложений должна сложиться понятийная база мониторинга, для чего требуется стандартизация типов хранимых данных.
  - Должны быть развиты методы построения реестров метаданных, которые сейчас ограничены индексацией по месту положения объектов в иерархической схеме базы данных LDAP.

- *Сервис репликации* поддерживает управление файлами большого объема. Репликация является одним из основных способов увеличения скорости работы с файлами и одновременно уменьшения нагрузки на сеть. Сервис репликации отвечает за порождение реплик, отслеживает их размещение (с

помощью каталога реплик) и предоставляет “лучшую” конкретным пользователям (причем, пользователь знает только имя файла). Разработанный прототип сервиса репликации Grid Data Management Pilot (GDMP) [35] уже реально используется для решения задач в области физики высоких энергий. Развитие в сторону интеграции с сервисами аутентификации и передачи данных Грид намечено в работе [36]. В то же время, управление репликами остается достаточно сложной открытой проблемой:

- При размещении и выборе реплик должна учитываться производительность сети, соединяющей пользователя и местонахождение индивидуальной реплики.
- Репликация должна взаимодействовать с сервисами планирования размещения заданий.
- Наиболее трудными представляются вопросы коллективной работы с репликами и обеспечением их идентичности.

Сервис репликации представляется одним из первых шагов на пути создания глобальной файловой системы для Грид с такими свойствами как единое пространство именования, независимость доступа от местоположения файлов, прозрачность выполнения файловых операций. В этой связи представляет интерес работа [37].

- *Сервис управления прикладным ПО* должен позволить выполнять задания повсеместно в Грид, независимо от типа вычислительных средств или операционной системы (то есть от платформы). Реальный путь для достижения этого – применение в прикладных приложениях стандартных средств программирования. При этом условии мобильность приложений может быть обеспечена на уровне исходных текстов программ, однако сервис управления должен уметь работать с исходными, объектными кодами и различными версиями приложений. Такое решение предложено в работе [21], однако, его применимость ограничена локальными системами. Расширение на глобальную среду Грид упирается в задачу создания глобальной файловой системы с прозрачным доступом к файлам.
- *Сервис авторизации*. До сих пор из всех вопросов безопасности Грид удовлетворительно решен вопрос аутентификации. Как результат следующего этапа – авторизации, задание, запущенное от имени пользователя, должно получить определенный набор ресурсов. Ключевой вопрос авторизации – создание таких средств для спецификации и проведения политики предоставления ресурсов, которые бы удовлетворяли требованиям: 1) минимизации личных контактов для получения доступа к ресурсам и 2) минимизации администрирования. В рамках же существующих технологий для работы на каждом ресурсе Грид необходимо обратиться к его владельцу для регистрации и создания соответствующего профиля. В работе [38] описывается архитектура централизованного и масштабируемого сервиса Community Authorization

Service (CAS). Тем не менее, удовлетворительного в практическом плане решения по-прежнему нет. Необходимы средства:

- спецификации прав пользователя, например, квот внешней памяти;
- динамического выделения ресурсов с учетом конкретных параметров задания, но в рамках прав данного пользователя;
- динамической регистрации пользователя в локальных системах без участия администратора.

- *Инфраструктура сертификации*. Безопасность Грид основана на сертификатах стандарта X509, законность которых удостоверяет Центр сертификации (Certification Authority – CA) своей подписью. Технология выдачи сертификатов содержит неформальный момент, связанный с идентификацией личности владельца и верификации его атрибутов, что предполагает личный контакт. Такая технология может работать в Грид лишь при условии распределения функции верификации и отнесения ее на уровень ответственных организаций. В таком варианте необходима поддержанная программно технология надежной передачи запросов на выдачу сертификатов между уровнями, а также стандартные регламенты и протоколы взаимодействия. Работы по выработке стандартов сервисов Центра сертификации ведутся в рамках проекта DataGrid.
- *Сервис учета и платежей*. Успех Грид будет во многом зависеть от того, удастся ли преодолеть естественное предубеждение владельцев ресурсов перед необходимостью открывать их для доступа посторонним. В числе прочего, организация функционирования Грид должна создавать стимулы для предоставления ресурсов и гарантировать их справедливое распределение. Все это будет возможно только на базе персонифицированного учета использования ресурсов и контроля лимитов. Уровень конфиденциальности и защиты данных, обеспечиваемый сервисами безопасности Грид, не уступает банковским технологиям, а потому достаточен для реализации надежной платежной системы. Ее основами могут служить сервисы протоколирования и учета в системе GRB [32] и CAS [38], а экономические модели для Грид рассмотрены в [39].

9. Слой координации

Кооперативный слой завершает превращение распределенных ресурсов в единую операционную среду с общими регламентами, стандартными протоколами и интеграционными сервисами. Однако можно утверждать, что Грид все-таки работать не сможет и останавливаться на кооперативном слое нельзя. Причина в том, что в любой момент времени общий объем ресурсов будет меньше потребностей, причем достаточно, чтобы не хватало какого-нибудь одного типа ресурсов, например сетевого. Для реальной организации

работы Грид необходимо распределять ресурсы не только по пространству, но и по времени, и это функция слоя координации.

Программное обеспечение этого слоя состоит из сервисов планирования, которые собирают ресурсные запросы пользователей, поддерживают очереди запросов, определяют порядок (расписание) их удовлетворения и в соответвии с расписанием выполняют соответствующие задания.

Планирование должно основываться на общих (для виртуальной организации) принципах и соглашениях по распределению ресурсов. По-видимому, следует исходить из интегральных по некоторому периоду фиксированных долей (квот) для пользователей из общего объема ресурсов. Соблюдение квот должно обеспечиваться планированием.

Как показывает анализ, сколько-нибудь эффективный алгоритм планирования невозможно построить, если известно только текущее состояние ресурсов. Нужен, по крайней мере, механизм, дающий оценку ближайшего времени получения заданного набора ресурсов. Более широкие возможности планирования открываются, если локальная система управления ресурсом умеет моделировать последовательность распределения ресурсов для множества запросов.

Создание слоя координации требует не только решения внутренних, довольно трудных задач – алгоритмов планирования в глобально распределенной среде, но и существенного расширения всех рассмотренных выше слоев Грид.

- В слое связи необходимо ввести в действие новое поколение протоколов, обладающих способностью программно регулировать сетевой трафик с помощью приоритизации сообщений и резервирования пропускной полосы.
- В слоях адаптации ресурсов, доступа и кооперации необходимы:
  - локальные системы управления, поддерживающие функции резервирования и моделирования распределения ресурсов;
  - новые типы интерфейсов с ресурсами, реализуемые на основе общей архитектуры сервисов OGSA;
  - расширение номенклатуры поддерживаемых каталогов на реляционные и объектные модели;
  - развитие информационных сервисов в направлении мониторинга событий в локальных системах управления с возможностями оперативного реагирования;
  - разработка принципов взаиморасчетов и моделей платежных систем.

Работ, которые можно отнести к слою координации, не много. В области диспетчеризации заданий в глобальной среде можно отметить систему Silver [40], однако статус ее реализации неизвестен и, кроме того, она не встроена в программную инфраструктуру Грид. Наша собственная разработка [41,42], выполнена в рамках средств системы Globus, но пока

используемые алгоритмы планирования весьма примитивны и ведется работа по их существенному улучшению.

10. Российский опыт Грид

В ИПМ работы по Грид были начаты в 1998 году. Тогда это называлось Метакомпьютингом, но было понятно, что потенциал нового направления далеко не ограничен сверхпроизводительными вычислениями, особенно с учетом того положения дел с обеспеченностью компьютерными мощностями и телекоммуникациями, которое мы имеем в России. Различные формы использования новых технологий, их связь с коммерческим программным обеспечением были рассмотрены в работах [43, 41].

За прошедшие годы проведены исследования по кластерным системам [44], применению в них распределенных файловых систем [45] и информационной службе системы Globus [46,47]. Наши программные разработки – это средство подготовки мобильных программ в кластерной среде [21] и диспетчер для Грид [42].

В практическом плане был создан стенд Грид, распределенный на две площадки (на Миусской площади и в районе метро Калужская) института. На каждой площадке работает сетевой кластер из рабочих станций. В качестве систем управления кластерами используется свободно распространяемая система OpenPBS [48], хотя имеется опыт применения и других систем – Condor [49] и DQS [50]. Связь между ресурсами осуществляется по Интернет, - правда этот вариант годится лишь для экспериментального режима, - а весь удаленный доступ управляется системой Globus.

Новые перспективы открылись в связи с инициативой ряда институтов физики высоких энергий (НИИЯФ МГУ, ОИЯИ, ИТЭФ, ИФВЭ) по созданию российского сегмента европейской инфраструктуры DataGrid. В сжатые сроки была проведена установка и конфигурирование необходимого программного обеспечения всех членов кооперации. В НИИЯФ МГУ открыт региональный Сертификационный Центр. Там же работает интегральный информационный сервер (GIIS), в который поступают данные с серверов организаций. В свою очередь, сервер НИИЯФ подключен к общему серверу DataGrid, и таким образом, реализована трехуровневая структура распределенной информационной базы.

В настоящее время на этой инфраструктуре производится тестирование и определение производственных характеристик (устойчивость, производительность, сетевые нагрузки) упомянутого выше Метадиспетчера [42]. В дальнейшем планируется его использование для балансировки вычислительной загрузки кластеров институтов.

11. Заключение

Родившаяся в рамках относительно узкой постановки высокопроизводительных вычислений Метакомпьютинга, идея соединения

вычислительных технологий с телекоммуникационными оказалась удивительно богатой и плодотворной. В результате нескольких лет исследований и пилотных проектов, география которых существенно расширилась и продолжает расширяться, сформировалось представление об основных принципах построения Грид: архитектуре, составе программного обеспечения и необходимых функциях. По-видимому, каких-то новых существенных предложений по номенклатуре ПО ожидать не стоит. Тем не менее, остается очень много теоретических проблем в рамках имеющихся постановок и еще большее поле для разработок и реализаций. Нужно признать, что пока окончательно не сложились даже базовые слои, и это сдерживает прогресс в слоях кооперации и координации, а именно они дают возможность практического использования Грид.

Литература

[1]. *Lyster P., Bergman L., Li P., Stanfill D., Crippe B., Blom R., Pardo C., Okaya D.*, CASA Gigabit Supercomputing Network: CALCRUST three-dimensional real-time multi-dataset rendering, Proceedings of Supercomputing ’92
[2]. *Catlett, C. and Smarr, L.* Metacomputing. Communications of the ACM, 35 (6). 44--52.1992.
[3]. http://www.globus.org
[4]. *Ian Foster, Carl Kesselman*, Globus: A Metacomputing Infrastructure Toolkit, International Journal of Supercomputer Applications, 11(2): 115-128, 1997.
[5]. http://legion.virginia.edu/
[6]. *Grimshaw A., Wulf W. et al*., The Legion Vision of a Worldwide Virtual Computer. Communications of the ACM, vol. 40(1), January 1997.
[7]. *Foster I., Kesselman C., Tuecke S*. The Anatomy of the Grid: Enabling Scalable Virtual Organizations. International Journal of High Performance Computing Applications, 15 (3). 200-222. 2001
. www.globus.org/research/papers/anatomy.pdf.
[8]. http://www.globus.org/developer/news/20011112a.html
[9]. http://www.ipg.nasa.gov/
[10]. http://www.ncsa.uiuc.edu/alliance/alliance/NationalComputational/ PrototypingTheFuture.html
[11]. *E.Deelman, I.Foster, C.Kesselman, M.*Livny. Griphyn data grid reference architecture. Draft, Jan 2001.
[12]. *G.Cancio, CERN, S.M. Fisher, RAL, T.Folkes, RAL, F.Giacomini, INFN-CNAF, W.Hoschek, CERN, B.L.Tierney, LBL/CERN*. The DataGrid Architecture. Version 1. March 7, 2001.
[13]. *Czajkowski, K., Foster I., Karonis N., Kesselman C., Martin S., Smith W. and Tuecke S.* A Resource Management Architecture for Metacomputing Systems. In 4th Workshop on Job Scheduling Strategies for Parallel Processing, Springer-Verlag, 1998, 62-82.

[14]. *Czajkowski, K., Fitzgerald, S., Foster, I. and Kesselman, C.* Grid Information Services for Distributed Resource Sharing, Proc. 10th IEEE International Symposium on High-Performance Distributed Computing (HPDC-10), IEEE Press, 2001. http://www.globus.org/research/papers/MDS-2.pdf
[15]. *Foster I., Kesselman C., Tsudik G. and Tuecke S*. A Security Architecture for Computational Grids. In ACM Conference on Computers and Security, 1998, 83-91.
[16]. *Ian Foster, Carl Kesselman, Jeffrey M. Nick, Steven Tuecke*. The Physiology of the Grid: An Open Grid Services Architecture for Distributed Systems Integration. http://www.globus.org/research/papers/ogsa.pdf
[17]. http://www.supercluster.org/maui
[18]. *Tierney, B., Johnston, W., Lee, J. and Hoo, G*. Performance Analysis in High-Speed Wide Area IP over ATM Networks: Top-to-Bottom End-to-End Monitoring. IEEE Networking, 1996.
[19]. *Beynon, M., Ferreira, R., Kurc, T., Sussman, A. and Saltz, J*., DataCutter: Middleware for Filtering Very Large Scientific Datasets on Archival Storage Systems. In Proc. 8th Goddard Conference on Mass Storage Systems and Technologies/17th IEEE Symposium on Mass Storage Systems, 2000, 119-133.
[20]. http://www.ietf.org/rfc/rfc1777.txt
[21]. *Хухлаев Е.В.* “Metamake – средство подготовки программ в сетевой гетерогенной среде”. Препринт ИПМ РАН, № 28, стр. 1-32, Москва, 1999
[22]. *Dierks, T. and Allen, C.* The TLS Protocol Version 1.0, IETF, RFC 2246, 1999. http://www.ietf.org/rfc/rfc2246.txt.
[23]. http://www.ietf.org/rfc/rfc2459
[24]. "Generic Authorization and Access control API ” (GAA API). IETF Draft. http://ghost.isi.edu/info/gss_api.html)
[25]. *Steiner, J., Neuman, B.C. and Schiller, J.*, Kerberos: An Authentication System for Open Network Systems. In Proc. Usenix Conference, 1988, 191-202.
[26]. *Allcock, B., Bester, J., Bresnahan, J., Chervenak, A.L., Foster, I., Kesselman, C., Meder, S., Nefedova,V., Quesnel,D. and Tuecke, S*. Secure, Efficient Data Transport and Replica Management for High-Performance Data-Intensive Computing. In Mass Storage Conference, 2001.
[27]. *Berman, F., Wolski, R., Figueira, S., Schopf, J. and Shao, G.* Application-Level Scheduling on Distributed Heterogeneous Networks. In Proc. Supercomputing '96, 1996.
[28]. *Frey, J., Foster, I., Livny, M., Tannenbaum, T. and Tuecke, S*. Condor-G: A Computation Management Agent for Multi-Institutional Grids, University of Wisconsin Madison, 2001.
[29]. *Abramson, D., Sosic, R., Giddy, J. and Hall, B*. Nimrod: A Tool for Performing Parameterized Simulations Using Distributed Workstations. In Proc. 4th IEEE Symp. On High Performance Distributed Computing, 1995.
[30]. *Beiriger, J., Johnson, W., Bivens, H., Humphreys, S. and Rhea, R.*, Constructing the ASCI Grid. In Proc. 9th IEEE Symposium on High Performance Distributed Computing, 2000, IEEE Press.

[31]. *С.А.Богданов, В.Н.Коваленко, Е.В.Хухлаев, О.Н.Шорин*, “Метадиспетчер: реализация средствами метакомпьютерной системы Globus”. Препринт ИПМ РАН, № 30, стр. 1-23, Москва, 2001
[32]. *S. Cavalieri and S. Monforte*. Resource Broker Architecture and APIs. University of Catania – Faculty of Engineering Department of Computer Science and Telecommunications Engineering (DIIT), June 2001. http://server11.infn.it/workload-grid/docs/20010613-RBArch-2.pdf
[33]. *Brian Tierney, Ruth Aydt, Dan Gunter, Warren Smith, Valerie Taylor, Rich Wolski, and Martin Swany*. A grid monitoring architecture. Technical Report GWD-Perf-16-2, GGF, 2001. http://www-didc.lbl.gov/GGF-PERF/GMA-WG/papers/GWD-GP-16-2.pdf
[34]. R-GMA - Relational Information Monitoring and Management System. User Guide Version 2.1.1, The WP3 relational team, October 15, 2001
[35]. *H. Stockinger, A. Samar, B. Allcock, I. Foster, K. Holtman, B. Tierney*. File and Object Replication in Data Grids. Proceedings of the Tenth International Symposium on High Performance Distributed Computing (HPDC-10), IEEE Press, August 2001. http://www.globus.org/research/papers/gdmp_hpdc_final_version.pdf
[36]. *B. Allcock, J. Bester, J. Bresnahan, A. Chervenak, I. Foster, C. Kesselman, S. Meder, V. Nefedova, D. Quesnel, S. Tuecke*. Secure, Efficient Data Transport and Replica Management for High-Performance Data-Intensive Computing. IEEE Mass Storage Conference, 2001. http://www.globus.org/research/papers/msc01.pdf
[37]. http://www.gridpp.ac.uk/slashgrid/
[38]. *L. Pearlman, V. Welch, I. Foster, C. Kesselman, S. Tuecke.* A Community Authorization Service for Group Collaboration. Submitted to IEEE 3rd International Workshop on Policies for Distributed Systems and Networks, 2001. http://www.globus.org/research/papers/CAS_2002_Submitted.pdf
[39]. *Rajkumar Buyya and Sudharshan Vazhkudai*, Compute Power Market: Towards a Market-Oriented Grid, The First IEEE/ACM International Symposium on Cluster Computing and the Grid (CCGrid 2001), Brisbane, Australia, May 15-18, 2001. http://www.buyya.com/papers/cpm.pdf
[40]. *Quinn Snell, Mark Clement, David Jackson, Chad Gregory*. The Performance Impact of Advance Reservation Meta-scheduling. Computer Science Department Brigham Young University Provo, Utah 84602-6576, 2000, http://supercluster.org/research/papers/ipdps2000.pdf
[41]. *Коваленко В.Н., Коваленко Е.И., Корягин Д.А, Любимский Э.З., Хухлаев Е.В.,* Управление заданиями в распределенной вычислительной среде. Открытые системы, № 5-6 (2001), стр. 22-28, http://www.osp.ru/os/2001/05-06/022.htm
[42]. *С.А.Богданов, В.Н.Коваленко, Е.В.Хухлаев, О.Н.Шорин*, “Метадиспетчер: реализация средствами метакомпьютерной системы Globus”. Препринт ИПМ РАН, № 30, стр. 1-23, Москва, 2001
[43]. *Коваленко В.Н., Корягин Д.А*. Вычислительная инфраструктура будущего. Открытые системы, № 11-12 (1999), стр. 45-52, http://www.osp.ru/os/1999/11-12/045.htm

[44]. *Коваленко В.Н., Коваленко Е.И.* Пакетная обработка заданий в компьютерных сетях. Открытые системы, № 7-8 (2000), стр. 1-19
[45]. *Коваленко В.Н.* Проблемы сетевых файловых систем. Открытые системы, №3 (1999), стр. 9-15, http://www.osp.ru/os/1999/03/03.htm
[46]. *М.К. Валиев, Е.Л. Китаев, М.И.Слепенков*. « Служба директорий LDAP как инструментальное средство для создания распределенных информационных систем». Препринт ИПМ РАН, № 23, стр. 1-22, Москва, 2000
[47]. *М.К.Валиев, Е.Л.Китаев, М.И.Слепенков*, “Использование службы директорий LDAP для представления метаинформации в глобальных вычислительных системах”, Препринт ИПМ РАН, № 29, стр. 1-27, Москва, 2000
[48]. http://www.openpbs.com
[49]. http://www.cs.wisc.edu/condor
[50]. http://www.scri.fsu.edu/~pasko/dqs.html