

Перспективы грид: грид-компьютинг – следующее поколение распределённого компьютеринга

В чём различие между грид-компьютингом и P2P, CORBA, кластер-компьютингом и DCE

Мэтт Хайнос,
руководитель проекта, грид-маркетинг и стратегия, IBM

Perspectives on grid: Grid computing – Next-generation distributed computing

What's different between grid computing and P2P, CORBA, cluster computing, and DCE?

<http://www-128.ibm.com/developerworks/grid/library/gr-heritage/>

Matt Haynos (mph@us.ibm.com)
Program Director, Grid Marketing and Strategy, IBM
27 Jan 2004

Необходимо отметить два важных фактора, которые за последние несколько лет впечатляюще повысили интерес к концепции грид-компьютинга. Режим экономии побуждает организации бизнеса и промышленности в условиях ограниченного бюджета ИТ более полно использовать имеющиеся у них компьютерные средства и обеспечивать большую гибкость своей ИТ-инфраструктуры для того, чтобы успевать за быстро изменяющейся конъюнктурой рынка путём разумного распределения конечных ресурсов для соответствующих приложений. В этой первой статье из серии публикаций "Perspectives on grid", Мэтт Хаунос проводит направленный анализ подобий и различий между грид-компьютингом и такими технологиями распределённого компьютеринга, как P2P, CORBA, кластер-компьютинг и DCE.

Рост популярности грид-компьютинга за последнее время как к архитектуре распределённого компьютеринга объясняется тем, что такая технология становится наиболее привлекательной для корпоративной ИТ-инфраструктуры. Подходы, предлагаемые грид-компьютингом, используются во многих областях бизнеса и промышленности при удовлетворении критически важных требований таких, например, как:

- грид-компьютинг привлекает финансовые структуры возможностями решения вопросов, связанных с управлением рисками, и в соответствии с предъявляемыми требованиями;
- в автомобильной промышленности решения грид-компьютинга используются для повышения эффективности процессов разработки продукции и расширения сотрудничества;
- нефтяные компании используют грид-технологии для ускорения обнаружения новых месторождений и увеличения числа успешных попыток бурения.

По мере совершенствования грид-компьютинга проявятся новые области его применения.

Главное отличие грид-компьютинга от почти всех, других парадигм распределённого компьютеринга можно сформулировать так: *квинтэссенцией грид-компьютинга является эффективное и оптимальное использование широкого ряда гетерогенных, слабо связанных ресурсов в организацию, обеспечивающую самые совершенные возможности управления рабочей нагрузкой или виртуализацию информации.* (Заметим, что организация может включать множество подразделений, быть пространственно-распределённой и т.д. Термин "организация" здесь используется в абстрактном смысле).

Действительно ли такая характеристика позволяет отличать грид-компьютинг от других моделей распределённого компьютеринга? Мы надеемся в данной статье ответить на этот вопрос, без обсуждения перспектив, рассматривая истоки и эволюцию грид-технологий и конкретно иллюстрируя, чем грид-компьютинг отличается от альтернативных решений распределённого компьютеринга, например таких, как P2P и CORBA. Мы будем сравнивать и сопоставлять концепции грида с наиболее известными решениями распределённого грид-компьютинга. Начнём с обсуждения цели грид-компьютинга.

Зачем грид-компьютинг?

После того, как компании проанализировали свои инвестиции в ИТ-технологии за несколько последних лет, многие из них пришли к заключению, что крайне важно достигнуть более полной отдачи от уже имеющихся средств компьютеринга. Поэтому вопросы использования встали особенно остро; необходимо добиваться большей функциональности в условиях ограниченного ИТ-бюджета.

Кроме этого, особой проблемой является умение *интеллектуально* распределять конечные ресурсы между соответствующими бизнес-приложениями в пространственно распределённом предприятии. Решение такой задачи обеспечивает корпорации значительную гибкость, реализуется ли это в виде возможности перераспределить

ресурсы с учётом рыночной конъюнктуры, или же в виде доступности бизнес-приложений, позволяющих лучше и быстрее обслуживать сложившуюся клиентуру.

Следуя принципам организации производственной линии - в которой большая часть её ресурсов выделяется для выпуска наиболее выгодных продуктов - управление рабочей нагрузкой имеет своей целью выделение ресурсов компьютеринга для наиболее важных приложений. Мы называем это *оптимизацией рабочей нагрузки*. Такой подход представляется привлекательным, хотя может оказаться связанным с необходимостью решения нескольких организационных проблем. Например, как вы решите, в чём состоит наиболее важная работа различных компонент или организаций в предприятии?

Однако, возможности роста производительности и прибыли, ассоциируемые с тенденцией к оптимизации рабочей нагрузки, остаются слишком значительными, чтобы пренебрегать ими. Идея грид-компьютинга как раз имеет своей целью решение вопросов, связанных с жизненной потребностью организации уметь эффективно использовать и перераспределять свои ИТ-ресурсы. Теперь перейдём к рассмотрению истоков этих подходов и концепций.

Истоки грид-компьютинга

Точно также, как это было с Интернет, академические организации стали первыми разработчиками технологий и архитектур, которые сформировали концептуальную основу грид-компьютинга. Такие инициативы, как Globus Alliance, China Grid и UK e-Science Grid core program стимулировали формирование, развитие подходов грид-технологий и подготовку их к использованию в е-бизнесе.

Грид-концепция родилась в среде научного сообщества, которое испытывало острую потребность в развитых средствах поддержки сотрудничества. Известно, что жизненно важным фактором научно-исследовательского процесса является его механизм распространения знаний – чем более эффективно вы можете обеспечить *не только* разделяемый доступ к большим объёмам информации, *но также* и к ресурсам, которые используются для создания этой информации, тем более высокий уровень качества сотрудничества вы достигнете.

Потребность в распространении знаний испытывает и мир бизнеса. Грид-компьютинг может и здесь найти себе применение, поскольку все большее значение приобретает интеграция бизнес-процессов и транзакций с использованием стандартов Web-служб. Важно, что поскольку адаптация грид-приложений в бизнесе уже происходит, то в стандартах (например, предложенных такими организациями, как Global Grid Forum) по необходимости будут учитываться требования практики бизнеса.

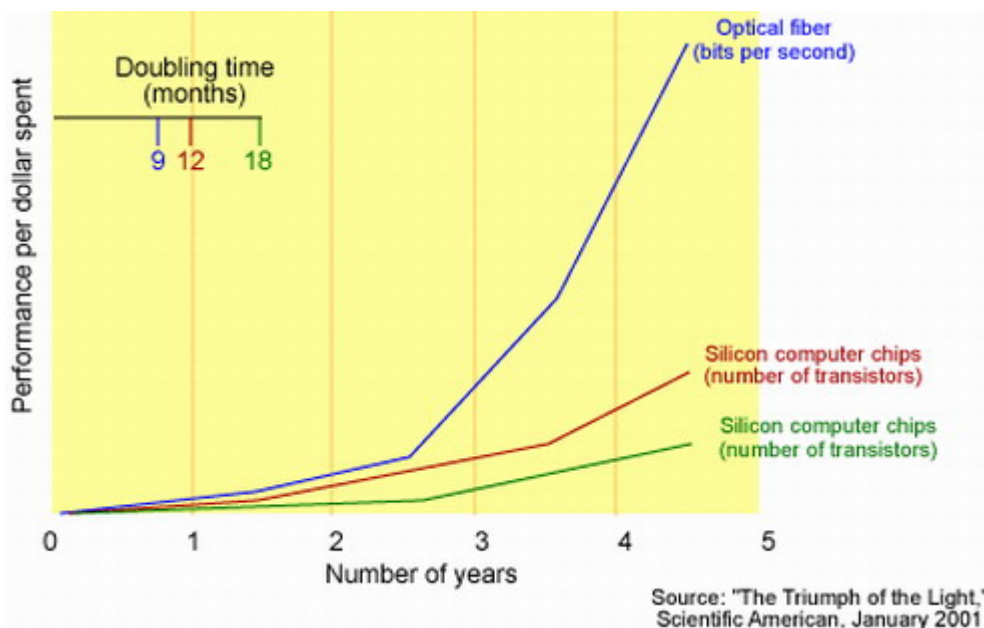
Наряду с этим и грид-компьютинг извлекает пользу из сопоставления первых академических грид-разработок и ранних стандартов с прагматичными реализациями, востребованными бизнесом. Нет оснований полагать, что такая синергия не будет продолжена по мере развития грид-компьютинга.

Грид устраняет критический разрыв

За последние несколько лет проявился существенный разрыв между темпами развития сетевых технологий, возможности которых измеряются в бит/сек, и технологий микропроцессоров, характеризуемых числом транзисторов в интегральной схеме. Это иллюстрируется на Рисунке 1.

Рисунок 1. Сопоставление темпов развития микропроцессоров, памяти и оптоволоконных сетей.

Источник: "The Triumph of the Light", Scientific American. January 2001.



Здесь показано, что в настоящее время пропускная способность сетей удваивается через каждые девять месяцев, хотя исторически этот рост был значительно медленнее. Закон же Мура утверждает, что удвоение числа транзисторов в кристалле происходит через каждые 18 месяцев. Такая разница в темпах создаёт проблему согласованного использования возможностей технологий.

Если принять как данное, что в настоящее время ускорение в базовых сетевых технологиях значительно выше, чем в технологиях микропроцессоров, то становится очевидной потребность в решениях, позволяющих более продуктивно использовать пропускные возможности сетей в распределённых компьютерных структурах. Такая постановка вопроса изменяет исторически сложившиеся ценовые отношения между сетевыми и микропроцессорными технологиями. Это же касается и технологий массовой памяти, где удвоение возможностей отмечается через каждые 12 месяцев.

Грид-компьютинг, используя технологии связывания распределённых ресурсов в единый виртуальный компьютер, позволяет преодолеть указанный разрыв и ценовые диспропорции. Сочетание такого ресурсоёмкого виртуального компьютера с возможностями создания ускоренных приложений (выполняемых не за недели, а за дни, не за дни, а за часы, не за часы, а за минуты и т.д.) открывает перед бизнесом весьма

соблазнительные перспективы, хотя при этом и потребуются значительные изменения в практике бизнеса телекоммуникаций, особенно в том, что касается цен.

Теперь, ознакомившись с истоками и целями грид-компьютинга, рассмотрим его возможности более конкретно и сделаем это на основе сравнения грида с другими парадигмами распределённого компьютеринга – с кластерами, CORBA, DCE и P2P.

Чем грид отличается от кластера?

Кластер вряд ли можно считать настоящим распределённым компьютерным решением (во всяком случае, пространственно распределённым решением); тем не менее, сопоставить грид-компьютинг с кластер-компьютингом очень полезно, особенно, если учесть, что люди часто смешивают грид-компьютинг с компьютерингом, реализуемым на базе кластера, а они существенно разнятся.

Грид состоит из разнообразных гетерогенных ресурсов. Кластер-компьютинг, главным образом, касается вычислительных ресурсов; грид-компьютинг интегрирует внешнюю память, сетевые средства, компьютеры, программные ресурсы и данные. В кластер, как правило, включаются однотипные компьютеры, и в нём используется одна операционная система; грид же может содержать компьютеры от разных производителей, на которых установлены разные операционные системы. (Программное обеспечение управления рабочей нагрузкой, разработанное фирмами IBM, Platform Computing, DataSynapse и United Devices, может распределять рабочую нагрузку на разных типах компьютеров с различными конфигурациями).

По своей сути грид является динамическим решением. Кластеры же, обычно, содержат статическое число процессоров и ресурсов; в гриде подключение и отключение ресурсов рутинная акция.

Гриду присуща пространственная распределённость через средства локальной, региональной или глобальной сети. Кластеры, как правило, реализуются в виде пула физически связанных компьютеров, который располагается в одном помещении. Гриды могут располагаться (и располагаются) где угодно. Кластерная технология связи предполагает крайне жесткие требования латентности, что накладывает сильные ограничения на взаимное расположение кластеров.

Концепция грида предполагает возможность динамического расширения диапазона масштабируемости. Напротив, физическая близость компонентов кластера и высокие требования к латентности средств коммуникации узлов кластера ограничивают его масштабируемость.

Например, недавно IBM и United Devices вместе с многочисленными партнёрами завершили грид-проект, посвящённый поддержке процесса идентификации перспективных лекарственных соединений для лечения оспы. Грид состоял приблизительно из двух миллионов персональных компьютеров. При использовании традиционных средств на выполнение исследований потребовалось бы нескольких лет, а на гриде этот процесс был завершён за шесть месяцев. Представьте, что было бы, если бы для поддержки исследования в проекте были задействовано 20 миллионов компьютеров. В таком случае работы могли бы быть выполнены в течение нескольких минут.

Кластеры и грид-решения вполне комплементарны; многие грид-реализации среди своих ресурсов содержат кластеры. Действительно, клиент грида может и не знать, что его задание фактически выполняется на некоем удалённом кластере. И хотя между гридами и кластерами существуют различия, они практически взаимосвязаны, поскольку в гридах всегда будет место для кластеров – определённые задачи всегда требуют использования сильно связанных друг с другом процессоров.

Тем не менее, в связи с ростом эффективности сетевых технологий, проблемы, которые раньше рассматривались только в контексте построения кластеров, будут разрешимы в грид-компьютинге. Понимание этого исключительно важно для установления баланса между естественной масштабируемостью грида и возможностями высокоскоростных межпроцессорных соединений, которые предлагаются кластерами.

Грид или CORBA?

Среди всех парадигм распределённого компьютерного грида CORBA, пожалуй, внешне наиболее совпадает с парадигмой грид-компьютинга. Это объясняется стратегической связью между грид-компьютингом и Web-службами, установленной архитектурой OGSA (Open Grid Services Architecture). Обе парадигмы базируются на концепции ориентированной на службы архитектуры (Service-Oriented Architecture, SOA). CORBA является базовой средой многочисленных критически-важных приложений и продолжает развиваться со времени её создания в 1991 году. Во многих аспектах CORBA оказалась предтечей мира Web (грид)-служб, в котором мы сейчас живём. Она заложила важный фундамент, поскольку несколько лет спустя после её реализации она инициировала разработку механизма удалённого вызова процедуры (Java™ Remote Method Invocation, RMI).

Например, Боинг использует технологию CORBA в своём фирменном приложении DCAC/MRM (Define and Control Airplane Configuration/Manufacturing Resource Management), которое по существу управляет процессами конфигурирования и усовершенствования компонентов коммерческих самолётов (а в авиалайнерах очень много компонентов!). Редактор издания e-Weeks technology Питер Коффи недавно сообщил, что всё проектирование нового океанского лайнера Cunard Queen Mary 2 проведено с помощью CORBA.

Ключевое различие между CORBA и грид-компьютингом выражается в том, что CORBA предполагает объектно-ориентированный подход к созданию приложений (что подтверждается даже её названием), в то время как грид-компьютинг этого не требует. В CORBA каждая сущность является объектом и поддерживает механизмы наследования и полиморфизма. В OGSA также есть некоторые подобия объектной концепции, однако при этом не предполагается объектно-ориентированная реализация архитектуры. В данном случае архитектура ориентирована на аппарат сообщений; объектный же подход является концепцией реализации. Тем не менее, использование языка формального определения, например, WSDL (Web Services Definition Language) согласно спецификациям платформы WSRF (Web Services Resource Framework) означает, что интерфейсы и взаимодействия в грид-компьютинге определяются точно также как в CORBA, что позволяет воспользоваться основными инструментами объектного подхода при проектировании программного обеспечения.

Другое отличие проявляется в том, что грид-компьютинг (OGSA) построен на фундаменте Web-служб. CORBA же интегрируется и взаимодействует с Web-службами. Одна из проблем, связанных с CORBA, состояла в том, что эта технология предполагала слишком много "крайних точек", которые по существу являются всеми компьютерами (клиентами и серверами) включёнными в среду CORBA. Существуют также проблемы, связанные с обеспечением интероперабельности между реализациями CORBA, предлагаемыми различными производителями, с организацией взаимодействия узлов CORBA через Интернет и с механизмом именованной крайних точек. Это означает, что все компьютеры CORBA-пула должны соответствовать определённым требованиям и определённым техническим политикам (предполагается, что все они используют одинаковые протоколы такие, как IDL, IOR и IIOP). Такой подход вполне оправдан, когда речь идёт о создании высоконадёжной, сильно связанной, предварительно компилируемой системы.

Тем не менее, синергия между технологией выполнения работ в среде CORBA и средствами Интернет не проявилась. CORBA побудила создание стандартов Web-служб – многим специалистам понравились концептуальные основы CORBA, и они начали разрабатывать такие стандарты, как XML, WSDL, SOAP и другие. Они усовершенствовали возможности интероперабельности и гибкости, которыми обладала CORBA, создавая для этого Web-службы на фундаменте открытой технологии Интернет, допускающей слабосвязанные и замедленные соединения между службами и их заказчиками. К этим усовершенствованиям OGSA добавила метод так называемой "мягкой реакции на отказ".

Архитектура Web-служб является служебно-ориентированной архитектурой и такой же во многом является CORBA, хотя она и имела другую цель – создание интегрированных систем с достаточно сильными внутренними связями.

В чём отличие от DCE?

Как следует из самого названия, DCE (Distributed Computing Environment) не столько архитектура, сколько исполнительная среда, и здесь проявляется самое важное отличие от грид-компьютинга. DCE можно определить, как набор сильно интегрированных технологий, разработанных для того, чтобы упростить проведение распределённого компьютеринга; грид-компьютинг (в интерпретации OGSA) является более сквозной архитектурой, сконструированной для инкапсуляции многих сложных механизмов распределённого компьютеринга.

Так же, как мы это видели при изучении CORBA, в случае DCE отличие проявляется в использовании сильносвязанных и слабосвязанных соединений ресурсов. В набор технологий DCE включены технология безопасности DCE ACLs (Access Control Lists), объектная и компонентная технологии (распределённые объекты DCE), файловая система DFS (Distributed File System) и описание каталогов (DCE registry) – фактически OGSA может работать над многими DCE технологиями.

Например, протоколы безопасности грид-компьютинга, представленные либо в форме GSI (Grid Security Infrastructure), либо в форме соответствующих стандартов Web-служб, могут взаимодействовать с DCE ACLs. Многие грид-приложения могут быть

выполнены для работы с нижележащей DFS (или с её предшественником AFS - Andrew File System). Базовый реестр грид-служб может работать с DCE-реестром.

Хотя большинство этих технологий могут считаться службами, DCE всё же не столько ориентированная на службы архитектура, сколько коллекция технологий. Её возможности поддержки создания приложений в среде SOA ограничены, поскольку DCE, главным образом, предоставляет блоки для создания распределённых приложений, но не обязательно для конструирования ориентированных на службы распределённых приложений.

Другое важное различие между грид-компьютингом и DCE, которое также относится и к CORBA, состоит в том, что архитектурой OGSA предусматриваются три категории служб:

- базовые грид-службы
- грид-службы данных
- грид-службы исполнения программ

CORBA, DCE и Java RMI не уделяют специального внимания данным (за пределами DFS) или службам исполнения программ, поскольку все эти технологии, по существу являются системами, реализующими протокол вызова удалённой процедуры (remote procedure call - RPC). Этот протокол позволяет одной программе, не зная деталей сетевого маршрута, обратиться к службе программы, находящейся на другом компьютере сети. Такая *синхронная* операция предполагает, что программа, запрашивающая услугу, приостанавливается до тех пор, пока удалённая процедура не возвратит результаты, если только при этом не используются *легковесные процессы*, разделяющие одно и то же адресное пространство. Многие из служб, определённых и реализованных как базовые грид-службы (так же, как и WSRF), аналогичны фундаментальным службам как DCE, так и CORBA. Однако в грид-компьютинге службы данных и исполнения программ имеют свою специфику.

И последнее. Различие между грид-компьютингом и CORBA, которое, как мы уже отметили, выражается в их связях со стандартами Web-служб, столь же справедливо по отношению к DCE. И, наконец, большинство усовершенствований в Web-службах обязаны опыту эксплуатации и настройки таких распределённых систем, как DCE и CORBA

И, наконец, о P2P

Приложения, например, такие, как KaZaA – которые постоянно упоминаются в заголовках из-за некоторой неясности с вопросом сохранения авторского права – являются главным источником, который в последнее время привлекает большое внимание к P2P (peer-to-peer) компьютеру. Эта технология сама по себе показывает интересные возможности распределённого компьютеру, хотя многие из них были бы весьма полезны, если бы использовались в грид-среде.

Во-первых, главным признаком P2P-системы является то, что она не имеет компонента централизованного управления; это делает её идеальной для анонимности и предлагает некоторую защиту от прослеживания. Напротив, грид-среды, как правило,

содержат некоторые средства централизованного управления и безопасности (например, в виде управления ресурсами или планирования рабочей нагрузки).

Отсутствие централизации в P2P-среде обуславливает два важных качества:

- P2P-системы, вообще говоря, гораздо больше масштабируемы, чем системы грид-компьютинга. Даже тогда, когда целенаправленно нарушен баланс между средствами управления и распределения, системы грид-компьютинга от природы не столь масштабируемы, как P2P-системы.
- P2P-системы, как правило, более устойчивы по сравнению с грид-системами к отказам в отдельных узлах. Хотя грид-системы способны восстанавливаться намного лучше, чем сильно связанные распределённые системы, они, тем не менее, по сути своей, имеют некоторые ключевые компоненты, которые могут оказаться отдельными точками отказа.

Это означает, что ключевым вопросом построения грид-систем является достижение баланса между децентрализацией и способностью управления – очень неприятная задача.

Кроме того, хотя важной характеристикой грид-компьютинга является то, что ресурсы динамичны, в P2P-системах ресурсы намного динамичнее по своей природе и, вообще говоря, более скоротечны, чем ресурсы грид-компьютинга. Главной целью как P2P-систем, так и систем грид-компьютинга является утилизация распределённых ресурсов. Имея доступ к множеству компьютерных ресурсов, такие системы будут пытаться использовать эти ресурсы настолько, насколько это возможно.

Наконец, ещё одно различие между этими двумя системами выражается в стандартах – недостаток общих стандартов в мире P2P контрастирует с множеством стандартов в сфере грид-компьютинга. И благодаря таким сообществам, как Global Grid Forum, грид-компьютинг располагает механизмом для уточнения существующих стандартов и созданием новых.

Можно предположить, что, взаимно обогащаясь, грид- и P2P-системы в конечном счёте сольются, особенно тогда, когда грид-системы достигнут стадии "межгридовского развития", в которой они, по существу, станут реализацией общественных услуг (public utilities).

О значимости мира данных

Мы изучили компоненты грид-компьютинга, рассмотрели их истоки, отметили его важность для бизнеса, приложений, основанных на Web-службах, и провели направленный анализ подобий и различий между грид-компьютингом и четырьмя основными типами систем распределённого компьютерного ресурса.

Почти каждая крупная организация располагает огромным, широко распределённым неиспользуемым потенциалом компьютерного ресурса. Виртуализация – движущая сила грид-компьютинга, может помочь задействовать неиспользуемые возможности, а IBM имеет богатую историю разработки виртуальной памяти, виртуальных хранилищ и

технологии виртуальных процессоров. Но не только в создании этой технологии для потребителей.

Разработанная на базе инструментального комплекса Globus система intraGrid фирмы IBM является исследовательским пилотным гридом, который позволяет IBM усилить её собственные глобальные активы для целей исследования. Кроме того, она даёт грид-разработчикам компании возможность понять реальный мир проблем и сложностей управления грид-инфраструктурой на уровне предприятия. IBM также использует множество гридов, имеющихся у компании, связывая их с Центрами Проектирования (Design Centers) для е-бизнеса, что позволяет IBM управлять этими Центрами как единым целым.

В следующей статье мы рассмотрим стратегические направления технологий для многих важных компонент грида. Грид-компьютинг, интегрируя дополнительные технологии, эволюционирует в направлении сред компьютеринга по запросу и адаптивного компьютеринга, воплощающих почти автоматизированное пространство, для промышленности и бизнеса.

Ссылки

- The "[History of CORBA](#)" содержит подробный отчет обо всех обновлениях спецификации CORBA, начиная с версии V1.0 в октябре 1991 года.
- "[On Death, Taxes, and the Convergence of Peer-to-Peer and Grid Computing](#)" сравнивает и противопоставляет P2P и грид-компьютинг, предлагая дорожную карту их возможной конвергенции.
- "[A visual tour of Open Grid Services Architecture](#)" объясняет цели и компоненты Открытой архитектуры грид-служб (OGSA), фундамента грид-компьютинга.
- "[Is Web services the reincarnation of CORBA?](#)" подробно обсуждаются различия между Web-службами и инструментарием CORBA, предлагается подход к оценке значимости Web-служб в распределенном мире компьютеринга.
- [CORBA](#) приводит список историй успеха во многих областях промышленности.
- [The Globus Alliance](#) является поставщиком широко используемого инструментального пакета Globus и одним из лидеров, определяющих основные операторы грида
- The [China Grid](#) это объединенный проект IBM и Китайского министерства образования, использующий технологию грида для предоставления университетам всей страны возможности участвовать в исследовательских, научных и образовательных проектах. Является одной из самых больших реализаций грид-компьютинга в мире. (Сайт China Grid выполнен на китайском языке; по ссылке [news item](#) можно ознакомиться с проектом на английском.)

- The [Global Grid Forum](#) это общедоступный форум более чем 5000 исследователей и практиков, разрабатывающих и продвигающих грид-технологии и лучшие примеры практического применения
- This [press release](#) объясняет роль грида в проектах IBM, United Devices и Accelrys по оказанию помощи глобальным исследовательским усилиям в разработке новых лекарств для борьбы с вирусом оспы.
- Visit the developerWorks [Grid computing zone](#) Посетите семинар разработчиков [Grid computing zone](#) для получения информации об использовании технологий грид-компьютинга