

**УМ и СИЛА сходятся:
почему грид и агенты нужны друг другу**

(Brain Meets Brawn: Why Grid and Agents Need Each Other)

СИЛА ЕСТЬ? УМА НЕ НАДО?
Реплика переводчика

Ian Foster
Argonne National Laboratory
& University of Chicago
foster@mcs.anl.gov

Nicolas R. Jennings
Electronics & Comp. Science
University of Southampton
nrj@ecs.soton.ac.uk

Carl Kesselman
Information Sciences Institute
University of Southern California
carl@isi.edu

http://www.semanticgrid.org/documents/003-foster_i_grid.pdf

Резюме

Концепции и механизмы открытых распределённых систем разрабатываются, хотя и с разными доминантами, как в контексте работ по грид-технологиям, так и работ по агентским системам. Исторически сложилось так, что в центре внимания *грид-сообщества* оказалась “*сила*”: то есть инфраструктура, инструментарии и приложения для надёжного и безопасного разделения ресурсов внутри динамичных и географически распределённых виртуальных организаций. В противоположность этому, основные интересы участников *агент-сообщества* определяют вопросы, касающиеся “*ума*”: то есть автономные решатели задач, которые могут гибко оперировать в неопределённых и изменяющихся обстоятельствах. Однако по мере того, как масштабы и амбиции проектов, рассматриваемых в обоих сообществах, возрастают, всё более очевидной становится конвергенция интересов, поскольку агентским системам нужна продуктивная инфраструктура, а грид-системам - механизмы автономного и гибкого управления. Побуждаемые этой конвергенцией интересов, мы рассматриваем состояние дел в обеих областях, возникающие в них проблемы и обсуждаем исследовательские и технологические разработки, которые могут способствовать взаимной поддержке усилий.

1 Введение

В открытых распределённых системах взаимодействие независимых компонент направлено на достижение отдельных и коллективных целей. При этом и компоненты и системы в целом проектируются так, чтобы они могли справиться как с изменением масштаба системы, так и с эволюцией природы её сущностей. Такие системы важны во многих контекстах, от больших научных коллабораций до промышленных организаций и сенсорных сетей.

Разработка подобных открытых распределённых систем ведётся, хотя и с разными доминантами, как в рамках грид-проектов, так и проектов, посвящённых агентским системам. Исторически грид-сообщество [12] сосредоточило своё внимание на том, что мы здесь называем **“мускулатурой”** – (brawn): то есть на интероперабельной инфраструктуре и инструментарии, обеспечивающих безопасное и надёжное разделение ресурсов внутри динамических и географически распределённых виртуальных организаций [14] и выполнение приложений при различных сценариях объединения этих ресурсов. Напротив, внимание специалистов, занимающихся развитием агентских технологий, сфокусировано на **“уме”** – (brain), то есть на развитии концепций, методологий и алгоритмов для автономных решателей задач, которые могут целенаправленно и аргументировано действовать в условиях неопределённой и динамично изменяющейся среды [21]. Ключевым посылом данной работы является то, что от таких агентов часто требуется самим организовать некий коллектив (например, подобный виртуальной организации) и далее взаимодействовать скоординировано. Потребность в поддержке агрегирования, в свою очередь, предопределяет широкое исследование развитых и гибких механизмов для управления такими взаимодействиями.

По мере развития исследований и всё большего внимания к фундаментальным задачам тематики, оба научных сообщества сталкиваются с серьёзными вызовами, как в плане масштабирования, так и применения систем. Причём для этого процесса развития характерно возрастающее перекрытие проблем, решаемых сообществами. Конкретно говоря, возможности современных грид-систем, с точки зрения их характеристик интероперабельности и интерактивности, представляются жёсткими и негибкими, тогда как агентские системы, которые обычно не проектируются в форме сложных масштабируемых распределённых систем, надёжных и безопасных [34]. Тем не менее, технологии каждого сообщества всё больше и больше распространяются в область интересов другого. Так, например, разработчики грид-систем всё чаще уделяют особое внимание вопросам гибкости и устойчивости, а агентские системы становятся более надёжными и масштабируемыми.

В связи с этим представляется целесообразным рассмотреть работы в обеих областях, во-первых, для того, чтобы ознакомить каждое сообщество с тем, что сделано другим, и, во-вторых, выявить возможности взаимного дополнения. Настоящую публикацию мы рассматриваем в качестве первого шага в этом направлении. С этой целью мы сначала рассматриваем состояние дел в областях грид- и агентских систем (Разделы 2 и 3), сравниваем и противопоставляем возможности обоих подходов (Раздел 4), представляем общее видение служебно-ориентированной архитектуры (Раздел 5) и в заключение кратко характеризуем наиболее значительные проблемы в рассматриваемых областях (Раздел 6).

Ограничения по времени и пространству вынудили нас воспользоваться в этой статье только работами, выполняемыми внутри указанных сообществ. Поэтому мы не затрагиваем интересные и связанные с предметом обсуждения работы по открытым распределённым системам, которые можно найти в других областях таких, как робототехника, P2P сети, семантические Web-службы, распределённые системы, искусственный интеллект и автономные системы.

2 Гриды

Гриды нацелены на создание возможности “разделения ресурсов и скоординированного решения проблем в динамичных научных объединениях, называемых виртуальными организациями” [12]. Другими словами, гриды обеспечивают инфраструктуру для разделения объединённых ресурсов, находящихся в доверяемых доменах. Во многом подобно Интернету, на базе которого они создаются, современные гриды определяют протоколы и промежуточное программное обеспечение, посредством которых можно обнаруживать, агрегировать и осваивать ресурсы. Возможности гридов покрывают широкий спектр приложений. Необходимо отметить, что стандартизация протоколов и интерфейсов, используемых при создании систем, является важной составляющей общей программы исследования и развития грид-технологий.

2.1 Технологии

В эволюции грид-технологий просматриваются, по крайней мере, три различимых этапа: ранние ad hoc решения, de-facto стандарты, базирующиеся на инструментальном пакете Globus Toolkit (GT), и появившиеся в настоящее время более формальные стандарты Web-служб, развиваемые в контексте Open Grid Services Architecture (OGSA) [13].

OGSA использует стандарты Web-служб, например, такой как Web Services Description Language (WSDL), в качестве базы для служебно-ориентированной архитектуры, внутри которой можно определять и

обнаруживать произвольные службы, а также обращаться к ним через установленные интерфейсы, независимо от того, как эти службы реализованы. Такой подход обеспечивает основу для решения вопросов виртуализации, интероперабельности и композиции.

Грид-сообщество участвовало, а в ряде случаев было инициатором, в разработке спецификаций Web-служб, которые были направлены на решение вопросов, возникающих при развитии грид-технологий. Концепция инфраструктуры WS-Resource Framework (WSRF) устанавливает унифицированные механизмы для определения, проверки и управления состоянием удалённого ресурса, что оказывается жизненно важным во многих ситуациях. Механизмы WSRF приняты в качестве основы при определении средств управления службами. Так, в OASIS (Organization for the Advancement of Structured Information Standards) обсуждается механизм Web Services Distributed Management (WSDM), а спецификация WS-Agreement – в (Global Grid Forum (GGF)). Эти процессы крайне важны для формирования технологий, обеспечивающих создание крупно масштабных и надёжных грид-систем, а также интероперабельных грид-приложений и служб. Другие подобные программы работ нацелены на стандартизацию интерфейсов для данных, компьютеров и других классов ресурсов.

Работа по грид-стандартизации корреспондируется с активной деятельностью сообщества, направленной на создание открытого грид-инструментария. Последняя версия инструментального пакета Globus Toolkit (GT), основанная на использовании Web-служб и установлений WSRF, предоставляет базовое промежуточное программное обеспечение для создания виртуальных организаций и разрешения таких проблем, как спецификация и поддержка многоплановой политики виртуальной организации, обнаружение, настройка (provisioning) и управление службами и ресурсами, объединение, репликация, обнаружение и передача данных. При развёртывании грид-системы в зависимости от доступных ресурсов и планируемых приложений, специфические реализации служб часто разрабатываются и используются в сочетании с другими компонентами, основанными на Globus Toolkit.

Результаты исследований и разработок по грид-технологиям воплотились в спецификациях и технологиях для создания службно-ориентированных архитектур в соответствии с принципами построения надёжных распределённых систем. Механизмы глобального управления, способные надёжно реагировать на отказы и воспринимать изменяющиеся условия среды, стали менее зависимы от особенностей приложений.

2.2 Приложения

В начале большинство грид-приложений разрабатывалось в связи с потребностями научно-исследовательского компьютеринга [6, 10, 19]. Они обеспечивали широкомасштабный распределённый компьютеринг (поддержку объединения компьютеров) [2, 15], интеграцию больших репозитариев данных (гриды данных) [7], коллаборации (научное сотрудничество) [31] и телеинструментарий [23, 26]. В последнее время грид-технологии, как средства решения проблем виртуализации и управления распределёнными системами, серьёзно обсуждаются и в промышленности [13].

Пакет Globus Toolkit практически используется в виртуальных организациях, интегрирующих ресурсы от 20 до 50 сайтов с тысячами единиц ресурсов для обработки и хранения данных, и ожидается, что в недалёком будущем показатели уровня интеграции будут от сотен до тысяч сайтов. Заканчивая данный раздел, мы приведём несколько примеров, чтобы показать области применения и масштабность грид-систем.

Сеть США (US Network), обслуживающая Грид для инженерного моделирования землетрясений (Earthquake Engineering Simulation Grid – NEESgrid), связывает экспериментальные устройства (например, вибростенды), архивы данных, компьютеры и пользователей инженерного сообщества, занимающегося проблематикой землетрясений. Службо-ориентированная архитектура этого Грида определяет стандартные интерфейсы для удалённого использования, мониторинга и дистанционного управления научным инструментарием, а также для публикации, обнаружения и доступа к данным, получаемым с помощью этого инструментария [26]. В NEES grid-экспериментах участвуют более 50 удаленных пользователей и задействованы средства трёх сайтов.

Grid3 [15] связывает 28 сайтов, содержащих кластеры с общей ёмкостью около 3000 процессоров. Эти ресурсы используются научными сообществами физики высоких энергий, астрономии, биологии, химии и информатики при решении больших задач моделирования и численного анализа.

Напротив, инфраструктура Acces Grid [31] нацелена на поддержку межличностных взаимодействий с помощью аудио- и видео-аппаратуры, а также приложений внутри групп сотрудничающих специалистов. В этой инфраструктуре грид-технологии используются для обеспечения безопасности, обнаружения и управления ресурсами.

Инфраструктура Butterfly.net, разработанная на базе пакета Globus Toolkit, обеспечивает настраиваемую среду для проведения многосторонних интерактивных игр, в которых запросы на компьютерные и сетевые ресурсы могут изменяться в очень широком диапазоне, поскольку популярность таких игр всё время растёт [24]. В качестве

другого примера коммерческого использования грид-технологий можно указать на инфраструктуру GlobeXplorer, использующую пакет Globus Toolkit для поддержки задач сбора и обработки данных, поступающих со спутников. [17].

Опыт работы с такими приложениями выявляет проблемы, которые необходимо решить для того, чтобы грид-системы стали бы масштабируемыми для более крупных сообществ, более разнообразных пулов ресурсов и более сложных приложений. Мы рассмотрим эти проблемы в Разделе 6.

3 Агентский компьютеринг

Агент “является инкапсулированной компьютерной системой, которая находится в некоторой исполнительной среде, и может гибко и автономно действовать в этой среде в соответствии с её функциональными требованиями” [33]. Говоря более подробно [21], агента можно определить как сущность:

- ◆ решающую чётко опознаваемую задачу с ясно очерченными граничными условиями и интерфейсами;
- ◆ размещённую (встроенную) в конкретную исполнительную среду, где он через сенсоры получает входные данные относительно состояния этой среды и воздействует на неё через механизмы, называемые эффекторами;
- ◆ спроектированную для исполнения специальной роли – ей указаны частные цели, которые необходимо достигнуть, и она обладает особыми возможностями (службами) решения задач, позволяющими довести выполнение требуемого запроса до конца и без “зависания”;
- ◆ автономную – она управляет как своим внутренним состоянием, так и своим поведением;
- ◆ способную проявлять гибкость при исполнении спроектированной роли; агенту необходимо быть как реактивным (способным своевременно реагировать на изменения, которые происходят в его среде), так и проактивным (способным выгодно выбирать частные цели и проявлять инициативу).

Если рассматривать мир приложений с платформы агентских технологий, то скоро становится понятным, что большинство задач требует или связано с использованием множества агентов, например, для представления: децентрализованного характера задачи, множества точек контроля, множества направлений решения, или конкурирующих запросов. Более того, таким агентам приходится интерактивно взаимодействовать, либо для того, чтобы достигать своих индивидуальных целей, либо для того, чтобы управлять зависимостями, которые вытекают из ситуаций,

возникающих в общей исполнительской среде. Таким образом, в любой рассматриваемой системе могут быть как сотрудничающие, так и индивидуалистские агенты, нацеленные, соответственно, на то, чтобы максимизировать продуктивность системы и максимизировать их собственную индивидуальную результативность. Эти интерактивные взаимодействия воплощаются в некую форму семантической интеграции, которая может хорошо отражать корректные взаимосвязи, включает обычные возможности обнаружения и активизации служб, а также и более сложные коллективные взаимодействия, связанные с кооперацией, координацией и переговорами о том, какие услуги выполнены к настоящему времени и с помощью каких агентов.

В большинстве случаев агенты действуют, либо выполняя запросы, поступающие от имени персон (или организаций), либо как часть некоего более мощного решателя проблем. (Здесь следует отметить аналогию с концепцией виртуальной организации). Таким образом, во время взаимодействия агентов имеет место некоторый основной организационный контекст, который определяет отношения между ними. Например, агенты могут быть равными исполнителями в некоей команде, или же один может быть менеджером этой команды. Чтобы охватывать такие связи, агентские системы обычно располагают точными конструкциями для моделирования отношений между организациями или ролей, например таких, как ровня, менеджер или член команды. Во многих случаях эти отношения подвержены непрерывному изменению: социальные взаимодействия подразумевают, что существующие отношения эволюционируют (например, команда, состоящая из равных может выбрать лидера), или же, что создаются новые отношения (например, ряд агентов может образовать виртуальную организацию, чтобы предоставлять особое обслуживание, которое ни один индивидуум не может предложить). Во временном аспекте поле этих отношений также может чрезвычайно варьироваться: от связей, достаточных для того, чтобы одновременно предоставлять конкретные услуги, до постоянных связей.

Независимо от характера процесса общения агентов, есть две особенности, которые качественно отличают взаимодействия агентов от тех, что происходят в других компьютерных моделях. Во-первых, это тенденция к возрастанию сложности взаимодействий между агентами по сравнению с другими контекстами, имеющими, например, дело с представлениями сотрудничества, координации и переговоров. Во-вторых, агенты являются гибкими решателями проблем, оперирующими в исполнительской среде, состояние которой они могут контролировать и обозревать только частично. Поэтому взаимодействия необходимо обрабатывать столь же гибким способом, и, кроме того, агентам нужен программный аппарат, для того, чтобы реализовывать контекстно-зависимые решения о характере и пределах их взаимодействий и для того, чтобы инициировать (или отвечать на) взаимодействия, которые не были

предусмотрены во время проектирования. Однако недостаток такой автономии и гибкости проявляется в том, что при этом трудно гарантировать проведение желаемой глобальной дисциплины. С этой целью часто применяется ряд средств (например, таких, как закрепление навыка, обучение, специальные программы и аппаратура), которые способствуют проверке и поддержке большего порядка.

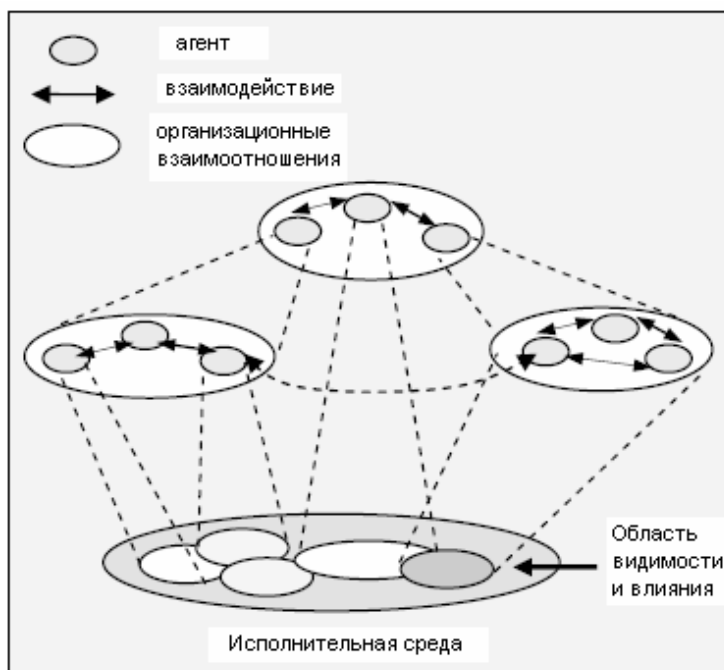


Рис. 1: Каноническое представление агентской системы

Представленная на Рис. 1 схема иллюстрирует, что принятие агентски-ориентированного подхода при проектировании системы связано с декомпозицией проблемы на множество взаимодействующих, автономных компонент, которые преследуют конкретные цели и могут предоставлять конкретные услуги. Ключевыми абстрактными моделями, которые определяют агентски-ориентированное представление замысла системы, являются агенты, взаимодействия и организации. Наконец, явные структуры и механизмы часто используются для описания и обработки сложных и изменяющихся сплетений организационных связей, которые существуют между агентами.

3.1 Технологии

В отличие от грид-компьютинга в агентских технологиях меньше внимания уделяется опознаваемым решениям, которые готовы для использования при разработке приложений. Традиционно, больше внимания

привлекали теории и модели разработки агентов и механизмов их общения, сотрудничества и переговоров. Итоги таких усилий воплотились в ряд алгоритмов, которые могут использоваться как для разработки отдельных агентов, так и управления их взаимодействиями. В центре внимания первого направления были такие алгоритмы и архитектуры, которые давали агенту возможности в условиях неопределённой и непредсказуемой среды планировать эффективный курс действий для достижения цели, адаптировать своё поведение к преобладающим обстоятельствам и достигать эффективного баланса между излишней способностью к ответной реакции (и непрерывным варьированием своих целей так, чтобы процесс решения задачи не завершился безрезультатно) и его чрезмерной приверженности текущему курсу действий (таким, что более важные действия не выполняются своевременно). В контексте второго направления были разработаны алгоритмы, которые агенты могут использовать для эффективного проведения переговоров, формирования команд с оптимальным составом участников и определения степени доверия, которая может быть установлена для конкретного агента на основании характеристик его социальных и организационных отношений.

В настоящее время усилилась тенденция к использованию агентских технологий в качестве важной базы при построении сложных распределённых систем. Несколько агентских инструментальных инфраструктур поддерживают конкретные агентские архитектуры и предлагают библиотеки интерактивных протоколов (например, JACK, JADE, Cougaar и ZEUS), разработаны методологии проектирования программного обеспечения для анализа и создания систем, базирующихся на агентах (Gaia, Tropos и AUMI). Проводятся работы по стандартизации различных механизмов агентских систем, например, таких как межагентские связи (FIPA, KQML). Более того, также как в грид-сообществе, усиливается стремление к использованию Web-служб и технологий семантических Web-служб при создании компьютерной инфраструктуры для агентских систем и растёт понимание важности доверия как центральной проблемы при взаимодействии.

3.2 Приложения

В последнее десятилетие агентская технология использовалась только в отдельных приложениях. Однако за последние несколько лет количество и сфера применения приложения значительно увеличились. В частности, многие большие компании сейчас заинтересованы в развитии приложений, использующих агентские технологии, и распространяемые приложения есть в таких областях как промышленность, электронная коммерция, управление процессами, телекоммуникационные системы, управление трафиком транспорта, сбор и фильтрация информации, управление бизнес-процессами, оборона, развлечения и медицинское обслуживание [25].

4 Мозг и мускулатура

Мы видим, что в основе как агентов, так и гридов есть общая идея, а именно, создание объединений или виртуальных организаций, ориентированных на общую цель или мотив. Однако обсуждаемые здесь сообщества сосредоточились на разных аспектах этой общей идеи. В случае гридов главный интерес посвящён механизмам, с помощью которых сообщества формируются и оперируют. Поэтому мы видим, что значительные усилия посвящены тому, как стандарты сообщества представлены через определённую политику, как политика проводится в жизнь и обеспечивается её исполнение, как члены сообщества идентифицируют друг друга, какие действия внутри сообщества реализованы, как обязательства членов сообщества устанавливаются, отслеживаются и проводятся в жизнь. С другой стороны, наше понимание того, как использовать эти механизмы для создания крупномасштабных систем с устойчивой коллективной дисциплиной менее зрело. Например, широко используемые грид-инструментарии обеспечивают единые механизмы для доступа к данным на различных системах хранения, но не для семантической интеграции таких данных; для доступа к службам и состоянию ресурса, но не для предупреждения, проверки и диагностирования проблем, которые могут возникнуть в связи с изменениями этого состояния; для безопасной аутентификации пользователей и служб, но не для того, чтобы делать вывод о том, можно ли или нет доверять конкретным пользователям или службам выполнению конкретных действий. В пределах этих возможностей гриды – всё же мускулатура, но не мозг.

Агенты сфокусированы на создании сообщества, чувствительного к различным дисциплинам поведения. При этом цель достигается путём использования системных компонент принятия гибких локальных решений, интегрируемых на основе разнообразных общественных взаимодействий и чётких организационных структур. Однако при достижении всей этой гибкости и сложности весьма ограниченное внимание уделялось тому, как эти задачи могут и должны быть решены в реальной распределённой исполнительной среде. Например, агентские инфраструктуры применяют сложные внутренние средства обоснования и вывода решения, но не предлагают поддержки для безопасных взаимодействий или обнаружения служб; кооперативные алгоритмы производят социально оптимальные результаты, но предполагают, что агенты имеют полные знания обо всех результатах при любой потенциальной группировке агентов; и алгоритмы переговоров достигают оптимальных результатов для участвующих агентов, но предполагают, что все части системы известны на начальном этапе переговоров и не будут изменены в течение системной операции. Таким образом, можно сказать, что агенты всего лишь мозг, но не мускулатура.

Ясно, что ни одна технология не идеальна. Для того, чтобы гриды были эффективны, они должны обладать гибкими возможностями принятия решений.

С другой стороны, агентам нужна надёжная платформа распределённого компьютеринга, которая позволяла бы им обнаруживать, получать, объединять и управлять средствами, необходимыми для выполнения их решений. Другими словами, имеются хорошие возможности для использования синергии гридов и агентов.

Один подход к использованию такой синергии может состоять в простом наслаивании технологий, то есть реализации агентских систем над грид-механизмами. Однако более вероятным представляется, что настоящие выгоды интегрированного грид-агент подхода будут достигнуты только путём мелкоструктурного собеседования двух технологий, при этом грид-технологии становятся более агент-подобными, а агентские системы становятся более грид-подобными.

В качестве раннего примера такого тесного связывания отметим разработку, основанную на агентах выбора ресурсов, в которой обучение, базирующееся на закреплении навыков, используется для того, чтобы управлять распределением ресурсов для задач [16]. В этом случае "агент" (то есть логика, применяемая для задачи принятия решения) использует грид-функции для мониторинга состояния, обнаружения ресурсов и представления задачи. Агентская технология в свою очередь обеспечивает важную грид-функцию путём объединения агентов, реализующих такую надёжную дисциплину управления ресурсами, которая в противном случае не достижима. Вторым примером является использование механизмов автоматизации переговоров (особенно для различных форм торгов), при выделении ресурсов в грид-системах [32]. Здесь проектировщики учитывают как эффективность общих подходов достижения цели, так и Vickery протоколы торгов для проблемы выделения ресурсов внутри распределённой системы. Этот пример также показывает, как технологии, известные исследователям агентов, могут быть интегрированы с другими более стандартными компонентами внутри грид-архитектуры.

При таком уровне интеграции, несомненно, возникнут новые проблемы, как для агентов, так и для гридов. Однако в результате могут появиться инфраструктуры для конструирования надёжных, крупномасштабных, интеллектуальных распределённых систем, которые качественно и количественно превосходят наилучшие сегодняшние разработки.

5 Надёжные интеллектуальные службо-ориентированные системы

Рассмотрев ключевые особенности концепций гридов и агентских систем, мы теперь проведем сопоставительный анализ с целью выявления их общности и комплементарности.

5.1 Автономные службы

Сутью единой концепции, которая лежит в основе гридов и агентских систем, является понятие *службы*: то есть сущности, которая позволяет *клиенту* путём чётко регламентированного обмена сообщениями получить некую возможность [4]. Внутри третьего поколения гридов взаимодействия служб структурированы через механизмы Web-служб, и поэтому все сущности являются службами. Однако, хотя каждого агента можно рассматривать как некую службу (в том смысле, что он путём обмена сообщениями взаимодействует с другими агентами и своей исполнительной средой), столь же справедливо и утверждение, что не каждая грид-служба обязательно является агентом (в том смысле, что она может и не участвовать в обмене сообщениями, что обнаруживается при гибких автономных действиях).

Поэтому понятие “*автономное действие*” является самым важным при обсуждении вопроса о том, как можно организовать взаимодействие гридов и агентских систем. Чтобы показать, в чём здесь суть вопроса, давайте рассмотрим службу, которая инкапсулирует базу данных.

В среде локальной сети мы могли бы найти версию такой службы, которая отвечает на запросы “читать запись” или “записать запись”. Такая реализация не проявляет автономности действия.

С другой стороны, в более распределённой, административно гетерогенной и подверженной к отказам исполнительной среде такая служба, могла бы проявлять и более сложное поведение. Например, она могла бы реплицировать базу данных, динамически определяя количество создаваемых реплик, используя для этого модели знаний о надёжности и производительности системы. Протоколы распределённых переговоров можно было бы использовать для установления уровня пропускной способности, достижимого для очереди запросов на отдельной реплике, с тем, чтобы оптимизировать общую пропускную способность базы данных. Наконец, алгоритмы распределённого планирования и диспетчеризации можно было бы использовать для того, чтобы отображать очереди на конкретные реплики базы данных, так, чтобы минимизировать время ожидания ответов на запросы пользователей. Во всех этих случаях, надёжная служба базы данных, разработанная для того, чтобы действовать в условиях открытой распределённой системы, будет проявлять гибкую автономность действий, (в том смысле, что на её поведение влияет не только запрос пользователя, но также и другие соображения, включая локальные политики и результаты переговоров с клиентом). Короче говоря, такие службы будут действовать подобно агентам.

5.2 Обогащённые модели служб

Как агентские системы, так и грид-системы состоят из служб, *динамичных* и *имеющих описание, состояния*. Основная модель службы динамична в том смысле, что новые службы могут создаваться и уничтожаться в течение времени жизни системы. Здесь необходим тот вклад, который в создание средств управления временем жизни и именованной модели динамичных служб, внесли грид-технологии [13]. В этой модели и описание отказа службы, и определение семантики масштабируемых распределённых систем, представлены неявно. Напротив, такие вопросы редко принимаются во внимание в агентских системах, но они определённо могут извлечь выгоду из использования таких моделей для представления и управления динамичными службами.

Наличие или отсутствие описания является другой важной характеристикой службы. Служба, имеющая описание состояния (или более или менее эквивалентно *ресурс* [11]) характеризуется внутренним состоянием, которое сохраняется в ходе множества взаимодействий. Часто может оказаться полезным сделать это состояние видимым, так чтобы, например, другая компонента распределённой системы могла определить текущую загрузку сервера, политику управления доступом к серверу и/или схему(ы), поддерживаемую базой данных. Опять же, грид-технологии разрешают этот вопрос, определяя общую модель для представления и получения сведений о состоянии службы [11]. Данная модель включает механизмы для описания “времени жизни” состояния, а также средства задания и проведения в жизнь политики относительно доступа и модификации службы.

Модель с состоянием, используемая в гриде, определяет, как описывать состояние и обращаться к нему, но ничего не говорит о структуре или семантике состояния, с которым приходится иметь дело. Обычная практика состоит в определении состояния в терминах фиксированной схемы или атрибутов. Напротив, агентские системы разрешают проблему семантики, но не предоставляют устойчивой модели состояния. Интегрированный подход может использоваться для публикации богатой семантической информации внутри модели состояния, предлагаемой в гриде, и, таким образом, усовершенствовать интероперабельность возможностей приложений при обнаружении, конфигурировании и управлении службами.

5.3 Переговоры и служебные контракты

Переговоры являются символом ум/мускулатура раскола между современными гридами и агентскими системами. Вообще говоря, нельзя предположить, что при обслуживании пользователю всегда предоставляется конкретная возможность: поставщик может оказаться не в состоянии или не хотеть предоставить услугу предполагаемому пользователю. Поэтому, если

система рассчитана на поддержку разных форм предсказуемого поведения, то ясно, что необходимо предоставлять обязательства (контракты) о готовности обеспечить услугу и характеристики или качество её исполнения.

При наличии способности предоставлять ресурс, обеспечивающий желаемый уровень обслуживания, возникает вопрос, какие же конкретно уровни обслуживания возможны и должны быть получены. Процесс, посредством которого можно ответить на такой вопрос, неизбежно будет представлять собой некоторую форму переговоров, так как автономные сущности, участвующие в этом процессе, должны прийти к взаимно приемлемому соглашению по существу дела. Если переговоры заканчиваются успешно, то есть, оба участника пришли к соглашению, то в результате появится контракт (договор об уровне обслуживания) между поставщиком услуги и её потребителем.

Порядок ведения переговоров может быть весьма многообразным; есть миллион различных протоколов и исследователи агентов немало потрудились над тем, чтобы определить какие протоколы соответствуют каким обстоятельствам [9]. В этом контексте переговоры регулируются операционной политикой как поставщика услуг, так и потребителя. А именно, условия политики должны рассматриваться в различных аспектах, например, таких, как текущая загрузка, идентичность и репутация потребителя, а также его платёжеспособность.

Использование переговоров в качестве средства формулирования контракта на обслуживание представляет исключительный интерес как для агентского сообщества [22], так и для грид-сообщества [8]. Один обещающий подход внутри гридов состоит в том, чтобы представить договор в виде утверждения о согласованной политике и определить надёжные расширяемые протоколы для изменения и согласования условий политики. Создание этих договоров с учётом модели отказов Байзатини может оказаться сложным. После проектирования протоколов ведения переговоров, следующий шаг состоит в том, чтобы определить стратегию, которой должны следовать системные компоненты, чтобы достигать своих политических целей. Стратегии могут варьироваться от очень простой (например, от агента, предлагающего правильное определённую стоимость услуги) до сложной (способности обсуждать о других участниках и их возможных стратегиях).

5.4 Управление виртуальной организацией

Общие средства взаимодействия как грид-систем, так и агентских систем встречаются, когда несколько агентов объединяются для формирования новой виртуальной организации. Такие виртуальные организации можно рассматривать как некую форму динамичной композиции служб: ряд изначально различных сущностей, договорившись о некоторых взаимно приемлемых рабочих соглашениях, собираются вместе для того, чтобы

сформировать новую сущность которая предлагает новое обслуживание. В таких случаях одна из ключевых проблем для участвующих агентов состоит в том, чтобы выяснить, кто ещё должен быть вовлечён в коалицию и как должны быть распределены различные роли и ответственности. Опять же такая деятельность обычно связана с необходимостью проведения переговоров между участниками для того, чтобы определить взаимно приемлемые соглашения относительно разделения труда и обязательств.

Динамическое создание также поднимает проблему обнаружения службы. Опыт грид-сообщества показывает, что это обнаружение должно осуществляться не просто путём использования типа службы, а скорее должно исходить из понимания возможностей службы (то есть семантики). Хотя грид-технологии обеспечивают средства для описания и группирования служб, эти высокоуровневые средства “сватовства” и обнаружения всё же не являются общеупотребительной частью грид-инфраструктуры. К счастью, в этом плане исследователями и разработчиками агентов получены значительные результаты, и их освоение способствовало бы усовершенствованию процессов управления виртуальными организациями. Эта интеграция может оказать влияние как на представление состояния, так и на то, как службы организованы.

5.5 Аутентификация, доверие и политика

Как отмечалось в разделе 5.2, ассоциация идентичности с динамически создаваемыми службами постоянно является существенной проблемой грид-инфраструктуры. Общий подход к этой проблеме состоит в том, чтобы отображать идентичности в глобальном пространстве имён и затем применять делегирование, как средство построения объединённых пространств имён для динамически создаваемых сущностей. В последнее время внимание фокусируется на применении обогащённых операторах политики и создании сообщества, основанного на утверждениях авторизации и полномочий [27].

Также фундаментальным для создания коллаборации и сообщества, а также для построения над вышеупомянутыми понятиями аутентификации являются понятия доверия. Эффективное управление доверием и политикой внутри сообщества, подобного виртуальной организации, требует гибких автономных механизмов, способных учитывать, в ходе организации сообществ, не только семантику положений политики, но также и способность договариваться об условиях политики и управлять ограниченным делегированием прав.

Как и в случае других аспектов агентов и гридов, мы надеемся увидеть адаптацию агентских алгоритмов и технологий, как они инкорпорируют спецификации политики и требований в свои базовые операции, и мы ожидаем увидеть, как грид-алгоритмы будут использовать широкие возможности некоторых моделей агентов доверия и репутации, которые уже разработаны

[28]. Мы также ожидаем, что различные типы положений политики, определяемые по мере их распространения и применения, будут более полно интегрированы в гриды. Например, механизмы аутентификации, основанные на репутации, которые служат агентским реализациям, представляются многообещающими в условиях грид-среды.

6 Десять научно-исследовательских проблем

В заключение мы в общих чертах охарактеризуем (в произвольном порядке) десять областей, в которых необходимо провести исследования для того, чтобы реализовать интегрированный агент-грид подход к созданию открытых распределённых систем.

Службо – ориентированная архитектура. Конвергенция агентских и грид-концепций будет ускорена, если мы сможем определить архитектуру интеграции служб, обеспечивающую надёжный фундамент для автономных форм деятельности. Эта архитектура должна установить

- ◆ базовый набор интерфейсов и дисциплин поведения, которые поддерживают динамические службы, имеющие описание состояния,
- ◆ комплект высокоуровневых интерфейсов и служб, для шифрования важных переговоров, мониторинга и управления шаблонами. Определение соответствующего набора таких архитектурных элементов само по себе является важной целью исследований и, кроме того, может облегчить создание, многократное использование и композицию интероперабельных компонент.

Истинность переговоров и управление. Работа всех распределённых систем, за исключением очень простых, связана с взаимодействием сущностей (служб), которым нет полного доверия. Поэтому решения об авторизации часто необходимо принимать в условиях, когда нет устойчивых взаимоотношений доверия. Промежуточное программное обеспечение гридов решает проблему безопасной аутентификации, но не значительно более трудные проблемы установления, мониторинга и управления доверием в условиях динамичной, открытой, многовалентной системы. Нужны новые средства для выражения и обоснования доверия. Механизмы репутации [29] и способность интегрировать утверждения многих полномочных источников (“А говорит, что М может выполнить X, но В не согласен с этим”) могут оказаться важными во многих контекстах, наряду с идентичностью и/или предыдущей деятельностью сущности, требующей какого-то действия или подтверждения некоторого факта, столь же важного как другие показатели, например, такие, как местонахождение или платёжеспособность. Вопросы доверия также связаны с проблемой интеграции данных, в том смысле, что наша убеждённость в достоверности “данных”, предоставляемых сущностью, может

зависеть от нашего доверия этой сущности, так, например, наше доверие утверждению “А говорит, что М зелёный” зависит от наших знаний о прошлом А..

Управление системой и поиск неисправностей. Грид-технологии облегчают безопасный, надёжный и единообразный доступ к большому числу ресурсов. Тем не менее, скоординированное управление этими ресурсами требует новых абстракций, механизмов, и стандартов для квази-автоматизированного (“автономного” [20]) управления таким ансамблем – несмотря на множество, возможно конкурирующих, целей различных участников ансамбля и сложного сценария отказов. С этим тесно связана проблема поиска неисправностей, то есть выявления, диагностирования и, в конечном счёте, адекватной реакции на непредсказуемое поведение отдельных компонент распределённой системы, или фактически системы в целом. Это требование будет мотивировать разработку надёжных и безопасных механизмов регистрации и аудита. Регистрация, обнаружение, мониторинг, управление доступными точками регистрации и разработка средств для выявления и устранения неисправностей (например, перегрузки или мошенничества), остаются открытыми проблемами. Нам также необходимы продвижения в анализе и толковании (например, путём визуализации) крупно масштабных распределённых систем.

Переговоры. Мы уже подробно обсуждали процедуру переговоров; здесь мы просто отметим, что в этой жизненно важной области ещё остались открытыми значительные проблемы.

Композиция служб. Реализация конкретного требования пользователя или виртуальной организации может потребовать динамичной композиции множества служб. Технологии Web-служб определяют соглашения относительно описания интерфейсов служб и последовательности выполняемых действий, а инфраструктура WSRF предоставляет механизмы для слежения за состоянием службы и организации наборов служб. Однако, нам нужны гораздо более мощные технологии для описания, обнаружения, композиции, мониторинга, управления и адаптации таких наборов служб.

Формирование виртуальной организации и управление. Хотя понятие виртуальной организации представляется интуитивным и естественным, мы до сих пор не располагаем чёткими определениями относительно того, что вводит в действие виртуальную организацию, или ясно очерченными процедурами решения вопросов о том, когда новая виртуальная организация должна быть сформирована, кто в ней должен участвовать, в чём суть их деятельности, когда виртуальная организация может быть изменена, и когда она неизбежно будет расформирована.

Предсказуемость системы. Хотя открытые распределённые системы по своей сути и непредсказуемы, всё же важно обеспечить гарантии, касающиеся производительности системы (например, живучести, сохранности свойств или стохастических границ производительности). Однако такие гарантии требуют глубокого понимания состояний, возникающих в сложных системах.

Человеко-компьютерное взаимодействие. Многие виртуальные организации будут представлять собой гибриды, в которых решение некоторых задач проводится людьми, а некоторых - программами. Эти компоненты должны взаимодействовать в бесшовной манере. Для того, чтобы освоить в таких гибридных системах разнообразные общественные взаимодействия, нужны новые модели совместной деятельности.

Оценка. Для содержательного сравнения новых подходов и технологий необходимо определить соответствующие эталонные тесты, круг остро проблемных приложений и создать исполнительную среду, в которой можно провести реалистичную оценку. Возможно, что одним наиболее эффективным средством продвижения агент-грид интеграции может оказаться определение весьма привлекательных проблемных приложений. Задачи таких приложений должны затребовать как мускулатуру грида, так мозги агентов и определить строгую систему показателей, которая может быть использована при проведении разработок в обеих областях. Возможные проблемные приложения могли бы охватывать распределённый мониторинг и управление крупномасштабными гридами, а также надёжность и долгоживущую функциональность агентских приложений.

Оценка может проводиться в условиях как моделируемой, так и реальной исполнительной среды. Значительный прогресс достигнут в системах моделирования как агентов, так и гридов (например, [30]). Практически используемые инфраструктуры, такие, как Grid3 [15], TeraGrid [5] и NEESgrid [26], а также полигоны, подобные PlanetLab[1], потенциально доступны как экспериментальные платформы для оценки объединённых систем, например, в контексте только что упомянутых остро проблемных приложений.

Семантическая интеграция. Открытые распределённые системы привлекают множество взаимодействующих участников к участию в торгах и поставках услуг. При этом трудно достигнуть полной согласованности взаимодействий, поскольку различные сущности в любой открытой системе используют разные информационные модели. Поэтому необходимы продвижения в таких взаимосвязанных областях, как создание тематических онтологий, схемное посредничество и семантическое посредничество [3]. Опять же, вопросы надёжности и стоимости здесь играют жизненно важную роль.

Благодарности

Работа первого автора была частично поддержана в рамках подпрограммы Отделения математики, информатики и научных вычислений, проводимой по контракту W-31-109-Eng-38 Управлением передовых исследований в научном компьютеринге Министерства энергетики США. Второй автор благодарит за поддержку проекта EPSRC “Virtual organizations for e-Science” (GR/S62710/01).

Литература

1. Bavier, A., Bowman, M., Chun, B., Culler, D., Karlin, S., Muir, S., Peterson, L., Roscoe, T., Spalink, T. and Wawrzoniak, M., Operating System Support for Planetary-Scale Network Services. *Networking Systems Design and Implementation (NSDI)*, 2004.
2. Beiriger, J., Johnson, W., Bivens, H., Humphreys, S. and Rhea, R., Constructing the ASCI Grid. *9th IEEE International Symposium on High Performance Distributed Computing*, 2000, IEEE Computer Society Press.
3. Berners-Lee, T., Hendler, J. and Lassila, O. The Semantic Web. *Scientific American*, 284 (5). 34-43. 2001.
4. Booth, D., Haas, H., McCabe, F., Newcomer, E., Champion, M., Ferris, C. and Orchard, D. Web Services Architecture. W3C, Working Draft <http://www.w3.org/TR/2003/WD-wsarch-20030808/>, 2003.
5. Catlett, C. The TeraGrid: A Primer, 2002. www.teragrid.org.
6. Catlett, C. and Smarr, L. Metacomputing. *Communications of the ACM*, 35 (6). 44-52. 1992.
7. Chervenak, A., Foster, I., Kesselman, C., Salisbury, C. and Tuecke, S. The Data Grid: Towards an Architecture for the Distributed Management and Analysis of Large Scientific Data Sets. *J. Network and Computer Applications*, 23 (3). 187-200. 2000.
8. Czajkowski, K., Foster, I. and Kesselman, C. Resource and Service Management. *The Grid: Blueprint for a New Computing Infrastructure (2nd Edition)*, 2004.
9. Dash, R.K., Parkes, D.C. and Jennings, N.R. Computational Mechanism Design: A Call to Arms. *IEEE Intelligent Systems*, 18 (6). 40-47. 2003.
10. DeFanti, T., Foster, I., Papka, M., Stevens, R. and Kuhfuss, T. Overview of the I-WAY: Wide Area Visual Supercomputing. *International Journal of Supercomputer Applications*, 10 (2). 123-130. 1996.
11. Foster, I., Frey, J., Graham, S., Tuecke, S., Czajkowski, K., Ferguson, D., Leymann, F., Nally, M., Storey, T. and Weerawaranna, S. Modeling Stateful Resources with Web Services. Globus Alliance, 2004.
12. Foster, I. and Kesselman, C. (eds.). *The Grid: Blueprint for a New Computing Infrastructure (2nd Edition)*. Morgan Kaufmann, 2004.

13. Foster, I., Kesselman, C., Nick, J.M. and Tuecke, S. Grid Services for Distributed Systems Integration. *IEEE Computer*, 35 (6). 37-46. 2002.
14. Foster, I., Kesselman, C. and Tuecke, S. The Anatomy of the Grid: Enabling Scalable Virtual Organizations. *International Journal of Supercomputer Applications*, 15 (3). 200-222. 2001.
15. Foster, I. and others, The Grid2003 Production Grid: Principles and Practice. *IEEE International Symposium on High Performance Distributed Computing*, 2004, IEEE Computer Science Press.
16. Galstyan, A., Czajkowski, K. and Lerman, K., Resource Allocation in the Grid Using Reinforcement Learning. *International Conference on Autonomous Agents and Multiagent Systems*, 2004.
17. Gentsch, W. Enterprise Resource Management: Applications in Research and Industry. *The Grid: Blueprint for a New Computing Infrastructure (2nd Edition)*, Morgan Kaufmann, 2004.
18. Goble, C.A., De Roure, D., Shadbolt, N.R. and Fernandes, A. Enhancing Services and Applications with Knowledge and Semantics. *The Grid: Blueprint for a New Computing Infrastructure (2nd Edition)*, Morgan-Kaufmann, 2004.
19. Grimshaw, A., Weissman, J., West, E. and E. Lyot, J. Metasystems: An Approach Combining Parallel Processing and Heterogeneous Distributed Computing Systems. *Journal of Parallel and Distributed Computing*, 21 (3). 257-270. 1994.
20. Horn, P. The IBM Vision for Autonomic Computing. IBM, 2001.
www.research.ibm.com/autonomic/manifesto.
21. Jennings, N.R. An agent-based approach for building complex software systems. *Communications of the ACM*, 44 (4). 35-41. 2001.
22. Jennings, N.R., Faratin, P., Lomuscio, A.R., Parsons, S., Sierra, C. and Wooldridge, M. Automated negotiation: prospects, methods and challenges. *Intl. J. of Group Decision and Negotiation*, 10 (2). 199-215. 2001.
23. Johnston, W. Realtime Widely Distributed Instrumentation Systems. Foster, I. and Kesselman, C. eds. *The Grid: Blueprint for a New Computing Infrastructure*, Morgan Kaufmann, 1999, 75-103.
24. Levine, D. and Wirt, M. Interactivity with Scalability: Infrastructure for Multiplayer Games. *The Grid: Blueprint for a New Computing Infrastructure (2nd Edition)*, Morgan Kaufmann, 2004.
25. Luck, M., McBurney, P. and Preist, C. Agent technology: Enabling Next Generation Computing. *AgentLink*. 2003.
26. Pearlman, L., Kesselman, C., Gullapalli, S., Spencer, B.F., Futrelle, J., Ricker, K., Foster, I., Hubbard, P. and Severance, C., Distributed Hybrid Earthquake Engineering Experiments: Experiences with a Ground-Shaking Grid Application. *13th IEEE International Symposium on High Performance Distributed Computing*, 2004, NEESgrid.
27. Pearlman, L., Welch, V., Foster, I., Kesselman, C. and Tuecke, S., A Community Authorization Service for Group Collaboration. *IEEE 3rd International Workshop on Policies for Distributed Systems and Networks*, 2002.

28. Ramchurn, S.D., Huynh, D. and Jennings, N.R. Trust in Multiagent Systems. *The Knowledge Engineering Review*. 2004.
29. Resnick, P., Zeckhauser, R., Friedman, E. and Kuwabara, K. Reputation Systems. *Communications of the ACM*, 43 (12). 45-48. 2000.
30. Song, H., Liu, X., Jakobsen, D., Bhagwan, R., Zhang, X., K., Taura and Chien, A., The MicroGrid: A Scientific Tool for Modeling Computational Grids. *SC 2000*, 2000, IEEE Computer Society Press.
31. Stevens, R. Group-Oriented Collaboration: The Access Grid Collaboration System. *The Grid: Blueprint for a New Computing Infrastructure (2nd Edition)*, Morgan Kaufmann, 2004.
32. Wolski, R., Brevik, J., Plank, J. and Bryan, T. Grid Resource Allocation and Control Using Computational Economies. Berman, F., Fox, G. and Hey, T. eds. *Grid Computing: Making the Global Infrastructure a Reality*, Wiley and Sons, 2003, 747-772.
33. Wooldridge, M. Agent-based software engineering. *IEE Proc. Software Engineering*, 144. 26-37. 1997.
34. Wooldridge, M. and Jennings, N.R. Software Engineering with Agents: Pitfalls and Pratfalls. *IEEE Internet Computing*, 3 (3). 20-27. 1999.