

Объединенный институт ядерных исследований
Лаборатория информационных технологий

На правах рукописи

Кутовский Николай Александрович

РАЗВИТИЕ МЕТОДОВ ПОСТРОЕНИЯ ГРИД-СРЕД И СИСТЕМ ОБЛАЧНЫХ
ВЫЧИСЛЕНИЙ ДЛЯ ЗАДАЧ ФИЗИКИ ВЫСОКИХ ЭНЕРГИЙ

05.13.11 - Математическое и программное обеспечение вычислительных
машин, комплексов и компьютерных сетей

Диссертация
на соискание ученой степени
кандидата физико-математических наук

Научный руководитель:
д.т.н. Кореньков В.В.
Научный консультант:
к.ф.-м.н. Зрелов П.В.

Дубна - 2014

Оглавление

Введение.....	4
Глава 1. Решение исследовательских задач в ФВЭ, связанных с применением облачных и грид-технологий.....	10
1.1. Основные задачи в ФВЭ, связанные с применением ИТ для хранения, передачи, обработки и анализа данных.....	10
1.2. Варианты синтеза облачных и грид-технологий.....	10
1.3. Мировой опыт решения исследовательских задач в ФВЭ, связанных с применением облачных и грид-технологий.....	14
1.4. Требования к комплексам для исследовательских целей.....	29
Глава 2. Методы построения облачных грид-систем и реализация такой системы в ЛИТ ОИЯИ.....	31
2.1. Методы построения облачных грид-систем.....	31
2.2. Описание реализации облачной грид-системы в ЛИТ ОИЯИ.....	32
2.2.1. Полигон на базе ППО ЕМІ.....	35
2.2.2. Полигон ТЗМОН.....	42
2.2.3. Полигон РГС.....	43
2.2.4. Полигон DesktopGrid.....	44
2.2.5. Веб-портал комплекса.....	46
Глава 3. Использование облачной грид-системы ЛИТ ОИЯИ.....	48
3.1. Полигон ЕМІ.....	48
3.1.1. Обучение пользователей.....	48
3.1.2. Обучение системных администраторов.....	51
3.1.3. Обучение разработчиков.....	53
3.1.4. Выполнение обязательств по проекту WLCG.....	53
3.1.5. Адаптация приложений.....	54
3.2. Полигон ТЗМОН.....	54
3.3. Полигон РГС.....	55

3.4. Методы адаптации определённого класса приложений для грид-сред на базе ЕМІ и РГС.....	57
3.4.1. Методы адаптации приложений для грид-среды на базе ППО ЕМІ.....	60
3.4.2. Методы адаптации приложений для грид-среды на базе ППО РГС.....	66
Заключение.....	77
Список терминов.....	78
Список литературы.....	88
Список иллюстративного материала.....	99
Приложение А. Программный код модуля расширения ВИГ РГС для ППП FDS 101	
Приложение Б. Программный код модуля расширения ВИГ РГС для ППП ZondGeoStat.....	112

Введение

Актуальность темы. Создание, сопровождение и развитие инфраструктур для хранения, обработки и анализа данных экспериментов в области физики высоких энергий (ФВЭ) включает решение большого круга задач, связанных с применением в данной области современных информационных технологий (ИТ).

В настоящее время широкое распространение получили грид-технологии и технологии облачных вычислений (облачные технологии). Каждая из них имеет свою специфику, преимущества и область применения.

«Грид (Grid) — согласованная, открытая и стандартизованная среда, которая обеспечивает гибкое, безопасное, скоординированное разделение ресурсов в рамках виртуальной организации» [1]. Другими словами, «грид — географически распределенная инфраструктура, объединяющая множество ресурсов разных типов (процессоры, долговременная и оперативная память, хранилища и базы данных, сети), доступ к которым пользователь может получить из любой точки, независимо от места их расположения. Грид предполагает коллективный разделяемый режим доступа к ресурсам и к связанным с ними услугам в рамках глобально распределенных виртуальных организаций, состоящих из предприятий и отдельных специалистов, совместно использующих общие ресурсы. В каждой виртуальной организации имеется своя собственная политика поведения ее участников, которые должны соблюдать установленные правила. Виртуальная организация может образовываться динамически и иметь ограниченное время существования» [2].

Грид активно применяется для проведения масштабных научных исследований. Он позволяет объединять в единую глобальную инфраструктуру разрозненные ресурсы организаций и совместно их использовать большому количеству пользователей. Например, без применения новых подходов к хранению данных, их обработке и анализу, которые заложены в концепции грид, решение задач экспериментов в области ФВЭ на Большом адронном коллайдере

(БАК) в Европейской организации ядерных исследований (ЦЕРН) было бы невозможным, учитывая объёмы поступающих от детекторов данных (десятки петабайт в год [3]), а также необходимость в их дальнейшей обработке и анализе тысячами учёных из распределённых по всему миру центров.

Грид-технологии успешно используются в области биомедицины и фармацевтики. В качестве примеров можно упомянуть такие проекты как MammoGrid и MammoGrid+ [4], имеющих своей целью качественное улучшение диагностики рака груди у женщин; WISDOM и WISDOM-II [5] — исследование посадочных леганд в поиске лекарства от малярии; neuGRID [6] — создание набора медицинских грид-сервисов для изучения дегенеративных болезней мозга. Активное применение и развитие грид-инструментария в области здравоохранения способствовало созданию международной ассоциации — HealthGrid Association [7]. Неполный список завершившихся и реализуемых проектов по внедрению и использованию грид-технологий в этой области приведён в [8, с. 40].

Кроме того, в Европе был реализован проект «Business experiments in GRID» (BEinGRID) [9] по изучению возможности трансфера грид-технологий из науки в коммерческую сферу, в результате реализации которого в 25 пилотных бизнес-проектах эти технологии были успешно внедрены.

Облачные технологии обеспечивают сетевой доступ к вычислительным, программным и информационным ресурсам (сетям передачи данных, серверам, устройствам хранения, сервисам и приложениям), конфигурируемым в соответствии с оперативными запросами. Они позволяют значительно сократить расходы на ИТ-инфраструктуру, удовлетворять динамически меняющиеся потребности в ресурсах и т. д.

Важную роль и перспективность обоих подходов в области распределённых вычислений и хранению данных подтверждают объёмы финансирования в размере 600 миллионов евро, выделенные Европейской комиссией для реализации 7-й рамочной программы в период с 2007 по 2013 гг. по исследованиям и технологическому развитию [10, с. 1].

Для широкого спектра задач в области ФВЭ является актуальным сокращение времени их выполнения, а также повышение эффективности использования ресурсов. Одним из решений представляется синтез облачных и грид-технологий. Так, повышение эффективности использования компьютерных ресурсов достигается при размещении грид-сервисов на виртуальных машинах в облачной среде (далее в тексте такие системы будут называться облачными грид-системами), а уменьшение времени выполнения — за счёт решения задач на отдельных специализированных комплексах.

Помимо этого, повышение эффективности использования ресурсов позволяет сократить их количество, необходимое для освоения облачных и грид-технологий, тем самым снижая порог вхождения в эти области.

Таким образом, представляется актуальной разработка методов создания многофункциональных гетерогенных комплексов для решения широкого класса задач в области ФВЭ, позволяющих сократить время решения этих задач и повысить эффективность использования ресурсов.

Целью диссертационной работы является развитие методов построения многофункциональных гетерогенных комплексов с использованием облачных и грид-технологий для решения широкого класса задач в области ФВЭ, позволяющих сократить время их решения и повысить эффективность использования ресурсов; проверка данных методов путём реализации подобного комплекса и использования его для решения конкретных задач; разработка методов адаптации приложений для грид-сред.

Для достижения поставленной цели необходимо было решить следующие **задачи**:

1. Провести анализ задач в области ФВЭ, связанных с хранением, обработкой и анализом данных, и вариантов синтеза облачных и грид-технологий для сокращения времени решения этих задач и повышения эффективности использования ресурсов.
2. Развить методы создания комплексов с использованием облачных и грид-

технологий для решения исследовательских задач в области ФВЭ с учётом их специфики и рационального использования компьютерных ресурсов.

3. С помощью развитых в данной работе методов построить облачную грид-систему и применить её для решения конкретных исследовательских задач ФВЭ.
4. Разработать методы адаптации определённого класса приложений для грид-сред на базе ЕМІ и РГС и провести их тестирование.

Методы исследования. В диссертационном исследовании применены современные подходы к организации распределенной обработки информации, в частности, технологии облачных вычислений, грид-технологии и технологии виртуализации.

Научная новизна данной диссертационной работы заключается в следующем:

1. Развита методика построения масштабируемых многофункциональных гетерогенных комплексов (облачных грид-систем), которые позволяют сократить время решения широкого круга задач, связанных с хранением, передачей, обработкой и анализом данных экспериментов БАК, и повысить эффективность использования компьютерных ресурсов.
2. Выполнен анализ и сформулированы основные требования для системы мониторинга ресурсов уровня Tier-3. Программная реализация такой системы увеличила эффективность использования ресурсов, позволив ускорить проведение анализа данных и получение физических результатов эксперимента ATLAS.
3. Разработаны методы адаптации определённого класса приложений для использования их в грид-средах на базе ЕМІ и РГС.

Защищаемые положения.

1. Развита в данной работе методика создания облачных грид-систем для решения исследовательских задач ФВЭ.
2. Облачная грид-система ЛИТ ОИЯИ, реализованная с использованием

предложенных методов.

3. Методы адаптации определённого класса приложений для грид-сред на базе ЕМІ и РГС
4. Ряд адаптированных в эти среды прикладных пакетов для научных исследований в различных областях.

Теоретическая значимость работы определяется развитыми в ней подходами и методами построения многофункциональных гетерогенных комплексов с использованием облачных и грид-технологий для решения широкого класса исследовательских задач в области ФВЭ, а также методами адаптации определённого класса прикладных пакетов для использования их в грид-средах на базе ЕМІ и РГС.

Практическая значимость состоит в том, что развитые в данной работе методы обеспечивают возможность тиражирования подобных комплексов, что позволило построить аналогичные инфраструктуры в институтах стран-участниц ОИЯИ: Центре суперкомпьютерных вычислений Национального технического университета Украины «Киевский политехнический институт», Физико-техническом институте низких температур им. Б.И. Веркина (Украина), Институте теоретической физики им. Н.Н. Боголюбова (Украина), Институте физики Национальной академии наук Азербайджана.

Комплекс ЛИТ ОИЯИ был использован для решения следующих задач:

- создание средств мониторинга сайтов уровня Tier-3 грид-инфраструктуры эксперимента ATLAS;
- разработка методов адаптации приложений для ЕМІ и РГС, а также перенос ряда прикладных пакетов в упомянутые грид-среды;
- написание подсистемы хранения данных для Российской грид-сети;
- разработка функциональных тестов для ряда грид-сервисов;
- обучение облачным и грид-технологиям, в том числе проведение на регулярной основе занятий в образовательных учреждениях.

Использование созданного комплекса для ряда задач подтверждено соответствующими документами.

Апробация работы. Основные положения и результаты работы докладывались и обсуждались на следующих российских и международных научных мероприятиях:

- The International Conference «Distributed Computing and Grid-technologies in Science and Education» 2008, 2010, 2012 (Dubna, Russia);
- The International Conference on Computing in High Energy and Nuclear Physics 2012 (New York City, USA);
- научные конференции молодых учёных и специалистов ОИЯИ 2008, 2010, 2011, 2012, 2013 (Дубна, Россия);
- программно-консультативные комитеты ОИЯИ 2010, 2012, 2013.

Цикл работ «Грид-среда ОИЯИ – элемент Российской и глобальной грид-инфраструктуры», включающий в себя исследования, вошедшие в диссертационную работу, был удостоен Первой премии ОИЯИ за 2013 год. Также эти исследования были поддержаны грантами для молодых учёных и специалистов ОИЯИ в 2010, 2011, 2012, 2013 и 2014 годах, в 2008 г. – стипендией им. Н.Н. Говоруна, в 2012 году удостоены премии конкурса молодёжных работ на конференции молодых учёных и специалистов ОИЯИ.

Публикации. По результатам диссертации опубликовано 29 работ, в том числе 4 [11, 12, 13, 14] — в рецензируемых изданиях¹.

Структура и объём работы.

Диссертация состоит из введения, трёх глав, заключения с основными результатами диссертации, приложений и списка литературы. Общий объём диссертации составляет 125 страниц, включая 14 рисунков и 7 таблиц. Список литературы состоит из 101 наименования на 11 страницах.

¹ Перечень российских рецензируемых научных журналов, в которых должны быть опубликованы основные научные результаты диссертаций на соискание ученых степеней доктора и кандидата наук: <http://vak.ed.gov.ru/87>

Глава 1. Решение исследовательских задач в ФВЭ, связанных с применением облачных и грид-технологий

1.1. Основные задачи в ФВЭ, связанные с применением ИТ для хранения, передачи, обработки и анализа данных

Для хранения, передачи, обработки и анализа данных в ФВЭ применяются современные ИТ – облачные и грид-технологии. При их использовании выделяются следующие основные задачи:

- 1) освоение, связанное с обучением различных категорий специалистов (пользователей, системных администраторов, разработчиков);
- 2) проведение различных исследовательских работ, включая поиск наиболее подходящей технологии, их комбинаций и (или) типа ресурсов для решения конкретной задачи;
- 3) развитие, включающее разработку нового функционала и сервисов, а также адаптацию уже существующего программного обеспечения (ПО) или создание нового для работы в той или иной среде;
- 4) использование, подразумевающее создание и эксплуатацию инфраструктуры.

Для краткости далее в тексте задачи 1-3 из списка выше будут называться «исследовательскими». Аналогично будут называться и комплексы, на которых они решаются, в отличие от производственных грид-систем².

1.2. Варианты синтеза облачных и грид-технологий

Хотя преимущественные области применения облачных и грид-технологий разнятся, для ряда перечисленных выше задач совместное использование обоих подходов позволяет повысить эффективность использования компьютерных ресурсов и скорость решения этих задач.

² Под производственной грид-системой понимается система, которая используется для хранения, обработки и анализа реальных данных (например, инфраструктура WLCG в случае экспериментов на БАК).

Можно выделить несколько вариантов синтеза облачных и грид-технологий (см. рис. 1):

- 1) облачные ресурсы используются в дополнение к грид-ресурсам (например, во время пиковых нагрузок для обеспечения необходимого качества обслуживания);
- 2) все грид-сервисы, включая элементы хранения данных и вычислительные элементы, размещаются на виртуальных машинах в облачной среде;
- 3) все грид-сервисы размещаются на виртуальных машинах в облачной среде и дефицит ресурсов удовлетворяется за счёт облака (т. е. комбинация двух предыдущих вариантов).

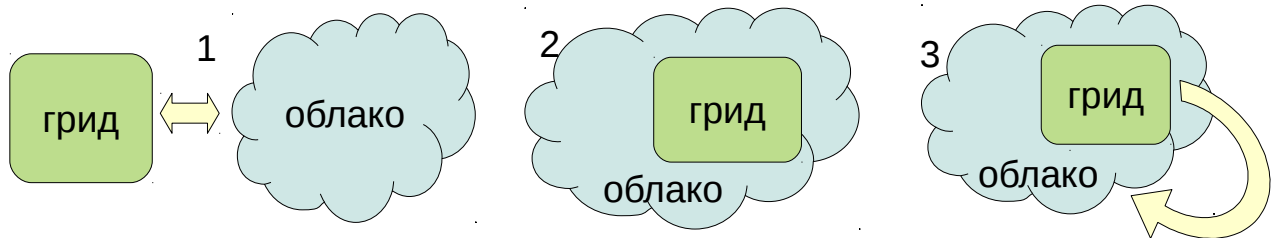


Рисунок 1. Схемы возможных вариантов синтеза облачных и грид-технологий

Соответствие задач вариантам совместного использования облачных и грид-технологий представлено в таблице 1 (знак «+» в ячейке таблицы указывает на возможность получения преимуществ от применения обеих технологий, а знак «-» — на отсутствие такой возможности либо её наличие, но не во всех случаях).

Таблица 1. Соответствие задач возможным вариантам синтеза облачных и грид-технологий

Задача \ Вариант	1	2	3
(1) обучение	-	+	+
(2) исследования	-	+	+
(3) развитие	-	+	+
(4) использование	+	-	-

Первый вариант не повышает эффективность использования компьютерных ресурсов, а позволяет только покрыть их дефицит. Он избыточен для исследовательских задач в силу их низкой ресурсоёмкости. Виртуализация может

приводить к существенным накладным расходам, поэтому размещение высоконагруженных грид-сервисов на виртуальных машинах (варианты 2 и 3) может быть непригодно для задачи (4). Таким образом, повышение эффективности использования компьютерных ресурсов возможно при размещении грид-сервисов в облачных средах, что наилучшим образом подходит для исследовательских задач, требующих существенных мощностей для их одновременного решения.

Если эффективное решение этих задач в области облачных технологий больших трудностей не вызывает, то в случае грид есть некоторые особенности, изложенные ниже.

Во-первых, каждый пользователь для работы в среде грид обязан иметь сертификат, выдаваемый удостоверяющим центром (УЦ). УЦ имеют определённую политику выдачи сертификатов, соответствующую его предназначению. Пользователь может получить сертификат в удостоверяющем центре только при соблюдении определённых условия (например, в случае УЦ RDIG³ претендент должен участвовать в проектах этого консорциума). Более того, особенностью проведения краткосрочных учебных курсов является то, что подписать заявки пользователей на получение сертификата необходимо как можно быстрее, чтобы пользователи могли максимально эффективно использовать время, отведённое для проведения курса, продолжительность которого может быть ограничена несколькими днями. Для этого соответствующие ответственные лица, связанные с удостоверением личности пользователя и выдачей ему сертификата, должны оперативно выполнять предписанные им функции (в идеале – по мере поступления заявок). Существующие УЦ, призванные выдавать сертификаты пользователей и машин глобальных производственных грид-систем, оставляют за собой право выдать сертификат пользователю в течение до нескольких рабочих дней (на практике время, прошедшее с момента подачи заявки до получения

3 Российский консорциум РДИГ (Российский грид для интенсивных операций с данными - Russian Data Intensive Grid, RDIG, www.egee-rdig.ru) организован в сентябре 2003 г. для создания грид инфраструктуры для интенсивных операций с научными данными.

сертификата, может составлять неделю или даже больше), что неприемлемо долго для краткосрочных курсов. В связи с этим для решения исследовательских задач сертификаты необходимо выдавать в специально созданных отдельных удостоверяющих центрах. Однако выданные такими УЦ сертификаты ресурсные центры (РЦ) глобальных грид-систем признавать не обязаны, что делает их ресурсы недоступными для пользователей с сертификатами от таких локальных УЦ.

Во-вторых, все ресурсы глобальных производственных грид-систем используются в рамках так называемых виртуальных организаций (ВО), каждая из которых призвана решать вполне конкретный круг задач в определённой проблемной области. Для доступа к ресурсам грид-системы любой пользователь должен являться членом как минимум одной ВО. Набор поддерживаемых ВО конкретным ресурсным центром является вопросом заинтересованности в поддержке каждой из них организацией, где размещается РЦ (как правило, в данной организации имеется некий коллектив, занимающийся проблематикой, совпадающей с целями и задачами конкретной ВО). Обучение студентов, магистров, аспирантов или даже состоявшихся в своей области специалистов, желающих получить базовые навыки работы в новой для них среде, например, с целью её оценки на предмет возможного будущего использования в качестве инструмента в своей профессиональной деятельности, а также решение других задач, отличных от решаемых на производственных грид-системах, требуют создания отдельной ВО, т.к. в правилах использования ресурсов производственных грид-систем, к которым имеет доступ та или иная ВО, чётко обозначено целевое использование этих самых ресурсов для решения конкретных задач данной виртуальной организацией.

В-третьих, эффективное освоение и развитие грид-технологий предполагает как можно более оперативное выполнение счётных задач на рабочих узлах. Например, ожидание результатов выполнения учебных задач несколько часов, а тем более дней, т.е. время, превышающее продолжительность занятия или курса,

из-за длительного их простаивания в очереди, является неприемлемым, в отличие от производственных грид-систем. Отдельная очередь в производственных РЦ с выделенными рабочими узлами для задач учебной ВО будет снижать эффективность использования самих этих ресурсов, т.к. нагрузка на них от счётных задач во время обучения невысока и запуск самих задач нерегулярен.

В-четвёртых, необходимое время действия сертификатов пользователей или машин, также как и продолжительность членства пользователей в ВО в случае исследовательских задач, может варьироваться от нескольких дней (например, интенсивное обучение) до месяцев (например, семестровые занятия для студентов) или лет (для разработчиков), что является нетипичным для ВО и УЦ производственных грид-систем и вполне может служить ещё одним аргументом в пользу установки отдельных УЦ и сервиса управления виртуальными организациями с одной или несколькими специализированными ВО.

В-пятых, в организации может существовать потребность в решении перечисленных выше непроизводственных задач в нескольких грид-средах одновременно или даже в установке нового ППО, тогда как производственные системы функционируют на одном, реже – на нескольких ППО, среди которых необходимых организации может и не быть, а установка нового ППО на ресурсах производственного РЦ неприемлема.

1.3. Мировой опыт решения исследовательских задач в ФВЭ, связанных с применением облачных и грид-технологий

На данный момент не существует каких-либо комплексов, позволяющих одновременно выполнять все перечисленные выше задачи на разных типах ресурсов и сред. Они решаются на отдельных системах в рамках каждого из проектов самостоятельно.

Так, например, в проектах EGEE, EGEE-II и EGEE-III⁴ существовали отдельные направления.

⁴ В рамках проектов EGEE, EGEE-II и EGEE-III была создана грид-инфраструктура, круглосуточно доступная учёным, независимо от их географического местоположения (<http://en.wikipedia.org/wiki/EGEE>).

В рамках направления SA3 осуществлялось управление процессом создания работоспособного и хорошо документированного ППО, включающее интеграцию пакетов из различных источников, конфигурирование, сертификацию и тестирование [15]. Для выполнения работ по данному направлению был создан отдельный полигон, за поддержание работоспособности каждого сервиса которого отвечала соответствующая команда, занимающаяся разработкой и сопровождением закреплённого за ней сервиса.

Для обучения было выделено отдельное направление — NA3. В рамках него была разработана целая программа с вводными и обучающими курсами. Согласно [16], вводные курсы покрывали широкий спектр тем от «Что такое Грид» и «Развитие Грид» до «Будущее Грид» и «Что такое электронная наука». Обучающие курсы, как правило, включали практические занятия по работе на инфраструктуре EGEE, для чего использовались портал GENIUS [17] и полигон GILDA [18] (более подробная информация о них приведена ниже). Помимо этого, проводились курсы по разработке приложений для платформы EGEE, а также по установке и администрированию сервисов ППО gLite. Разработанные обучающие модули были доступны для использования всеми партнёрами, участвующими в направлении NA3. Важными аспектами данного направления являлись следующие:

- поддержание учебных материалов в актуальном состоянии по мере развития инфраструктуры, появления новых сервисов и областей её применения;
- обеспечение в учебной программе материалов с примерами из реальной жизни, а также охват широкого набора тем для разных целевых групп: от новичков до экспертов в области грид-технологий.

Веб-портал GENIUS позволял работать с инфраструктурой EGEE с помощью графического интерфейса, тем самым существенно облегчая пользователям выполнение операций, связанных с отправкой задач на счёт, отслеживанием их статуса, получением результатов, загрузкой и сохранением данных.

GILDA (Grid Infn Laboratory for Dissemination Activities) — виртуальная лаборатория для демонстрации возможностей грид-технологий и их популяризации, которая состоит из следующих компонент:

- полигон GILDA — набор грид-сайтов и сервисов (система управления загрузкой, информационная система, файловый каталог, средства мониторинга, вычислительные элементы и элементы хранения данных), установленных в разных организациях-участниках этого проекта по всему миру и функционирующих на базе ППО INFN Grid (полностью совместимого с gLite);
- Grid демонстратор — модифицированная версия веб-портала GENIUS, с которого пользователи полигона GILDA могли запускать на этот полигон задачи, использующие предопределённый набор приложений;
- удостоверяющий центр (УЦ) GILDA — полнофункциональный УЦ, который использовался для выдачи сертификатов с четырнадцатидневным сроком действия любому желающему познакомиться с грид-технологиями на базе полигона GILDA;
- виртуальная организация GILDA — ВО, членство в которой предоставлялось всем желающим получить первый опыт работы грид на базе полигона GILDA;
- Grid репетитор — основанный на GENIUS веб-портал, который используется только во время практических занятий;
- GILDA P-GRADE портал — среда для разработки и выполнения, основанная на базе ПО P-GRADE команды MTA SZTAKI;
- интерфейс пользователя GILDA — интерфейс пользователя на базе gLite, доступный по протоколу SSH через веб-браузер;
- система мониторинга — набор утилит для мониторинга статуса полигона GILDA и доступности его сервисов;
- электронные списки рассылки.

Веб-портал GENIUS, как и инфраструктура GILDA позволяли проводить обучение только одной категории — пользователей. Для обучения разработчиков и системных администраторов каждый из партнёров-участников направления SA3 должен был самостоятельно создавать соответствующую инфраструктуру. Более того, четырнадцатидневный срок действия сертификатов пользователей, выдаваемый УЦ GILDA, не позволял проводить, например, семестровые занятия для студентов вузов.

Преемником проекта EGEE-III стал проект European Grid Infrastructure — EGI [19]. В нём поменялась организационная структура, которая стала представлять собой «федерацию из более 350 ресурсных центров, координируемых некоммерческим фондом EGI.eu, созданным для управления инфраструктурой от лица её участников — Национальных грид-инициатив (National Grid Initiatives, NGIs) и европейских международных исследовательских организаций» [20]. Согласно [21], по отношению к EGI.eu поставщики технологий, ресурсов, виртуальные исследовательские сообщества, организации по выработке международных директив и стандартов, а также координирующие и финансирующие проекты выступают в роли внешних партнёров (см. рис. 2).

Решение задач, связанных с освоением грид-технологий, полностью перешло на уровень национальных грид-инициатив, а задачи развития этих технологий, внедрения новых подходов и функционала — к внешним партнёрам. Вот что по этому поводу говорится на соответствующей странице портала EGI относительно поставщиков технологий [22]: «EGI не занимается разработкой ПО, установленного на грид-инфраструктуре, — все обновления и новые программы создаются в других местах в партнёрстве с независимыми поставщиками технологий. Поставщики технологий — это организации или проекты, сотрудничающие с EGI с целью разработки и предоставления ПО для использования в производственной инфраструктуре или сообществами её пользователей. Как правило, поставщики технологий берут на себя обязательства предоставлять ПО, удовлетворяющее одному или более требованиям

пользователей, описанных в технической дорожной карте EGI».

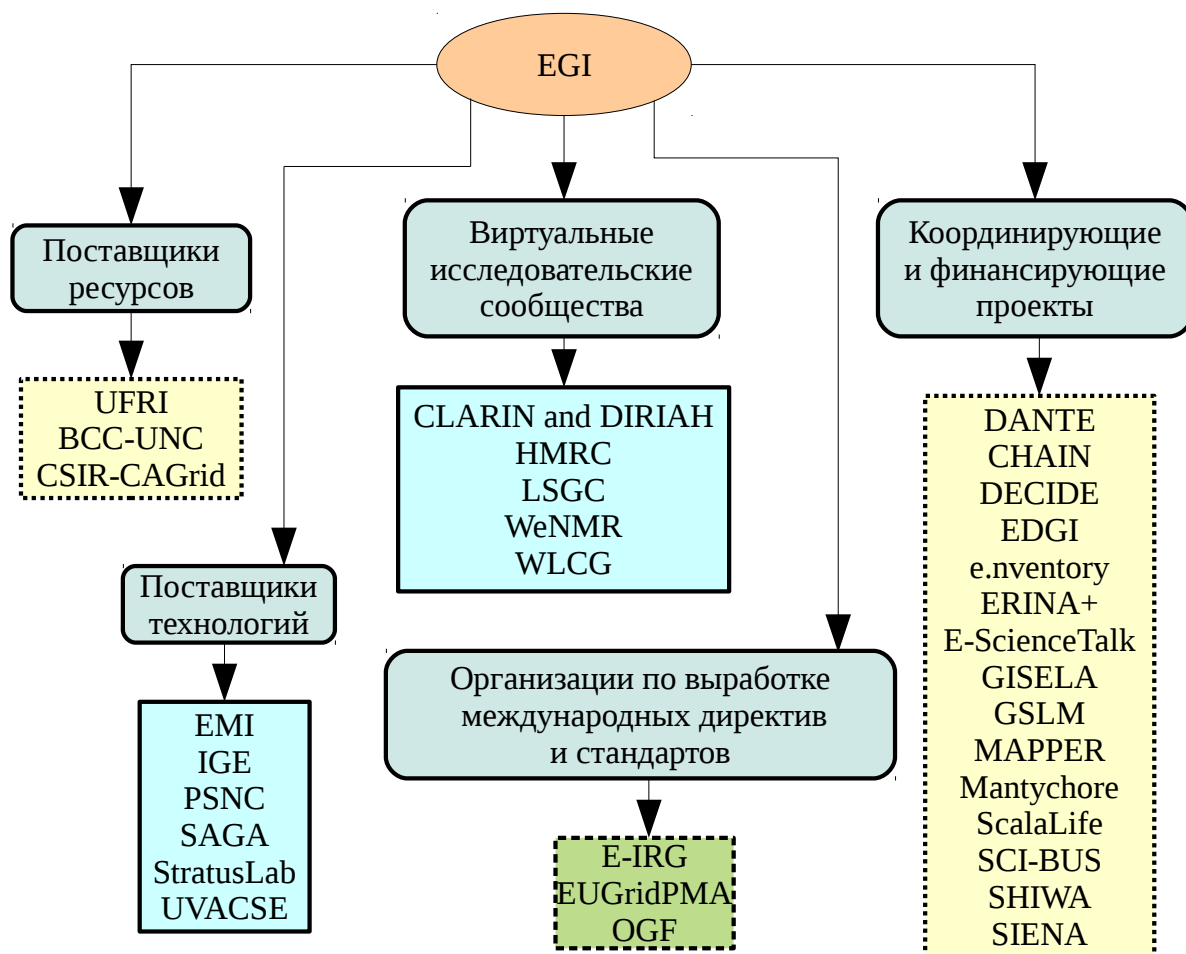


Рисунок 2. Схема взаимосвязей между организациями и проектами, являющимися участниками EGI

Согласно [21], на данный момент такими поставщиками технологий для EGI являются следующие.

European Middleware Initiative (EMI) [23, 24] — коллаборация четырёх основных поставщиков европейского ППО: ARC [25], dCache [26], gLite [27] и UNICORE [28]. Основным предназначением данного объединения является предоставление единого набора компонент ППО для установки на инфраструктуре EGI, выступающей частью унифицированного дистрибутива ППО (Unified Middleware Distribution, UMD [29]). Помимо этого, задачами EMI являются расширение интероперабельности и интеграции между грид-системами и другими вычислительными инфраструктурами, повышение надёжности и управляемости сервисов, а также установление стойкой модели сопровождения и

развития ППО, гарантирующую её соответствие требованиям научного сообщества.

Initiative for Globus in Europe (IGE) [30, 31] — это проект, «предоставляющий европейским исследователям программные средства для совместного использования вычислительных мощностей, баз данных и других онлайн инструментов. IGE является членом альянса Globus (Globus Alliance) — коллаборации, разрабатывающей Globus Toolkit [32], набор программных компонент с открытым исходным кодом, призванных решать наиболее сложные проблемы совместного использования распределённых ресурсов. С помощью данного сотрудничества IGE обеспечивает поддержку европейской компьютерной инфраструктуры и её пользователей, выступая центральной точкой взаимодействия в Европе и одновременно с этим усиливая влияние европейских разработчиков и пользователей на разработку Globus. IGE координирует работы в Европе в области Globus, а также помогает в развитии, адаптации под нужды европейских партнёров, организационно-техническом обеспечении, поддержке и сопровождении Globus Toolkit». Согласно [31], это предполагает следующие направления деятельности:

- непрерывное определение качества ПО,
- функционирование репозитория для европейских разработчиков Globus,
- сотрудничество с организациями по разработке стандартов,
- участие в соответствующих тематических конференциях,
- обучение, распространение и разработка документации.

Ещё одним поставщиком технологий является Познаньский суперкомпьютерный и сетевой центр (Poznan Super Computing and Networking Center, PSNC [33]). В соответствии с [34], участие этой организации в проекте EGI заключается в интеграции ППО QosCosGrid⁵ (QCG) в европейскую грид-инфраструктуру, что расширит возможности последней за счёт внедрения

5 QosCosGrid — система с возможностью расширенного управления задачами и ресурсами, предоставляющая конечным пользователям сравнимую с суперкомпьютерами производительность и архитектуру (<http://www.qoscosgrid.org/trac/qcg>).

механизмов расширенного резервирования и совместного использования гетерогенных ресурсов, тем самым предоставляя возможности для разработки пользовательских интерфейсов с расширенным функционалом.

Помимо этого, сотрудничество с PSNC предполагает работы, связанные с интеграцией в функционирующую грид-инфраструктуру программных компонент для мониторинга, учёта потреблённых ресурсов, сервисов предоставления информации и сопровождения.

Проект Simple API for Grid Applications (SAGA) [35, 36] является ещё одним поставщиком технологий для EGI. SAGA — это проект по созданию стандартизированного интерфейса программирования приложений (ИПП) для разработки распределённого прикладного ПО с целью его запуска в облачных и грид-средах. Сообщество SAGA состоит из разработчиков и конечных пользователей, которые совместно работают по следующим направлениям:

- создание программных абстракций высокого уровня, основанных на SAGA;
- создание и сопровождение документации со спецификациями по SAGA API;
- реализация SAGA API и его привязка к ППО;
- установка и сопровождение реализаций SAGA на множестве распределённых вычислительных инфраструктур.
- Сотрудничество SAGA и EGI происходит по таким направлениям, как
- предоставление надёжных, ориентированных на пользователя сервисов научным сообществам;
- выработка и соблюдение соглашений о качестве предоставляемых услуг для обеспечения поддержки третьего уровня на происшествия и запросы;
- ускорение разработки стандартов в рамках производственных грид-инфраструктур;
- распространение результатов сотрудничества в пределах сферы полномочий каждого из проектов;
- обмен идеями и сотрудничество по выработке моделей устойчивого развития;

- сотрудничество в развитии связей с бизнес-сообществом.

Проект StratusLab [37, 38] также является поставщиком технологий для EGI, цель которого — использование виртуализации и облачных технологий в грид-инфраструктурах. В зону ответственности StratusLab входит интеграция, развитие и поддержание работы самодостаточной облачной инфраструктуры на базе ПО с открытым исходным кодом для использования существующими и новыми грид-сайтами, а также разработка компонент для координирования размещения виртуальных машин и управления их репозиториями.

Последним в списке поставщиков технологий для EGI значится University of Virginia Alliance for Computational Science and Engineering (UVACSE) — одно из подразделений университета Вирджинии. Направления сотрудничества аналогичны оным с SAGA.

Соответственно, задачи, аналогичные выполняемым в рамках направления SA3 проектов EGEE, теперь решаются каждым из поставщиков технологий самостоятельно на собственных полигонах, включая интеграцию своих наработок в общий стек ППО.

Что касается обучения грид-технологиям в рамках EGI, то партнёры занимаются этим самостоятельно, а EGI предоставляет общий сервис, называемый EGI Training Marketplace. Согласно [39], этот сервис призван «координировать учебные курсы между сообществами, проектами и национальными командами». Training Marketplace позволяет «преподавателям размещать информацию о проводимых ими учебных мероприятиях и материалы курсов, также как и использующим EGI исследователям находить, получать доступ и комментировать учебные материалы и мероприятия, удовлетворяющие их потребностям».

Таким образом, в силу децентрализации организационно-административной модели управления проектом EGI, задачи обучения пользователей, которые раньше успешно выполнялись на инфраструктуре GILDA, каждой национальной инициативе в области грид приходится решать самостоятельно, создавая для

этого аналогичные учебные полигоны, на что далеко не у всех NGI хватает ресурсов. В качестве примера одного из успешных мероприятий по обучению пользователей в Европе можно привести GridKa School [40] — международную летнюю школу в области грид-технологий, проводимую в Штейнбукском суперкомпьютерном центре Технологического института Карсруэ и реализуемую при поддержке немецкой национальной грид-инициативы (NGI_DE).

Что касается решения задачи по адаптации существующего прикладного ПО в грид-среду, то согласно [41], для этих целей созданы соответствующие команды экспертов в рамках национальных грид-инициатив: «эксперты в рамках NGI доступны для оказания пользователям помощи и интеграции новых приложений в производственную инфраструктуру. Это подразумевает анализ существующего приложения, анализ ожиданий пользователя и его нужд, определение сценария портирования, написание кода, тестирование и в конечном итоге написание документации для приложения. Команда NGI также помогает в организации общего доступа к гридифицированному приложению для других пользователей сообщества EGI с помощью базы данных приложений».

В рамках проекта Open Science Grid (OSG) [42] задача проведения различных исследовательских работ в области грид, оценки и интеграции передовых разработок, а также внедрение нового функционала и сервисов для грид-инфраструктуры этого проекта решается подобно тому, как это было организовано в EGEE: для выполнения работ по каждой составляющей этой задачи сформированы отдельные команды и используется набор полигонов (Integration Testbed — ITB, Validation Testbed — VTB) [43]. Также, согласно информации в разделе часто задаваемых вопросов портала этого проекта [44], конечным пользователям предоставляется помощь в адаптации их прикладного ПО для использования на инфраструктуре OSG.

Обучение в OSG представлено школами, семинарами и онлайн курсами для самостоятельного освоения материала. В качестве целей этих мероприятий указано «преподавание основ распределённых вычислений и проведение

практических занятий по работе в грид-среде для исследователей и студентов в различных областях, включая естественные, прикладные, общественные, технические науки и информатику. Работая с ведущими экспертами в области грид-технологий, участники приобретают необходимые навыки для проведения и выполнения научного анализа в грид-средах посредством сочетания лекций, дискуссий и практических упражнений на масштабных аппаратных и программных ресурсах» [45].

Для проведения практических занятий создана отдельная виртуальная организация (ВО) OSGEDU [46]. Слушатели подобных учебных мероприятий должны быть зарегистрированы в этой ВО для доступа к грид-ресурсам и выполнения упражнений.

Не удалось найти информации о том, на каких ресурсах проводятся подобные мероприятия в рамках OSG для обучения пользователей: используется ли отдельный полигон или для ВО OSGEDU предоставляется часть ресурсов производственной грид-инфраструктуры OSG.

Также оказались безуспешными попытки получить актуальную информацию из открытых источников, что из себя представляет процесс обучения системных администраторов и разработчиков.

Осваивать грид-технологий пользователи могут также с помощью портала WS-PGRADE [47]. Он является веб-порталом системы gUSE [48] — grid and cloud User Support Environment — и позволяет выполнять разработку и запуск распределённых приложений, исполняемых на вычислительных ресурсах различных распределённых инфраструктур, включая кластеры, производственные грид-инфраструктуры (ARC, gLite, Globus, UNICORE), грид-среды на базе ПО BOINC с использованием персональных компьютеров и облачные платформы на основе Google App Engine.

Зарегистрированным пользователям и разработчикам приложений доступ к portalу WS-PGRADE предоставляется с помощью обычных веб-браузеров. Для разработки новых цепочек взаимодействия приложений и загрузки их в

репозиторий gUSE разработчикам приложений доступен расширенный функционал (граф, абстрактная последовательность, шаблон, приложение и проект). Конечным пользователем WS-PGRADE предоставляет полный доступ к параметризации и исполнению приложений, скачанных из репозитория gUSE.

Данный портал хотя и позволяет решать задачу обучения пользователей и разработчиков, а также адаптировать или создавать приложения для достаточно широкого набора типов распределённых вычислительных сред, однако она непригодна для обучения системных администраторов, а также разработки для отсутствующих в этой комплексе GRID-сред или распределённых систем хранения данных.

Что касается решения перечисленных в начале этой главы задач в странах СНГ, то на протяжении последнего десятилетия было реализовано несколько проектов и государственных программ, так или иначе связанных с использованием GRID-технологий в данном регионе.

Так например, для полномасштабного участия России в проекте EGEE был создан консорциум РДИГ (RDIG — Russian Data Intensive Grid) [49]. В рамках данного консорциума Россия участвовала, в том числе и в направлениях JRA, SA3 и NA3, тем самым успешно решая задачи освоения и развития GRID-технологий. Однако с окончанием проекта EGEE в РФ не была создана национальная GRID-инициатива, в рамках которой было бы возможным продолжение работ по данным направлениям.

В качестве главной цели программы союзного государства России и Беларуси «СКИФ-GRID» значится «освоение и адаптация передовых наукоемких технологий на перспективных суперкомпьютерных платформах, оптимизация суперкомпьютерных конфигураций семейства «СКИФ», ориентированных на построение на их основе GRID-компьютерных сетей, т. е. создание новой технологической базы для обеспечения динамики роста экономического и оборонного потенциала России и Беларуси» [50]. Решение задачи отладки и тестирования ППО и распределенных приложений в рамках этой программы было

выделено в отдельный проект — «СКИФ-Полигон», в качестве цели которого указано «создание СКИФ-Полигона — территориально-распределенной вычислительной системы (ГРИД-системы), объединяющей вычислительные установки участников программы «СКИФ-ГРИД», которая могла бы служить полигоном для отладки и тестирования программного обеспечения промежуточного уровня и распределенных приложений. В территориально-распределенной вычислительной системе СКИФ-Полигон будут реализовываться такие возможности, как:

- решение вычислительно сложных задач, недоступных для решения при помощи отдельных суперкомпьютерных установок;
- повышение эффективности использования дорогостоящих суперкомпьютерных установок за счет выравнивания нагрузки (обмена вычислительными заданиями);
- развитие новых технологий высокопроизводительных вычислений и обработки данных при помощи географически распределенных вычислительных ресурсов» [51].

Подготовка кадров упоминается как одна из составляющих четвертого направления работ программы «СКИФ-ГРИД»: «Пилотные системы: реализация прикладных систем в перспективных областях применения создаваемых вычислительных установок, решение актуальных задач на суперкомпьютерах и GRID-системах, усилия по подготовке и переподготовке кадров в области суперкомпьютерных и GRID-технологий». Также для решения задачи подготовки кадров и привлечения пользователей в рамках программы «СКИФ-ГРИД» была объявлена инновационная инициатива «СКИФ-ГРИД» образованию и науке. Согласно её пресс-релизу [52], участники данной инициативы смогут «[...] получить доступ и внедрить самые современные средства и методики обучения как ИТ-специалистов, так и специалистов в самых разных областях человеческой деятельности, способных эффективно использовать всю мощь суперкомпьютерных технологий», а также «организовать процесс подготовки

своих специалистов и студентов, нацеленный на решение самых сложных задач на передовых рубежах научно-технического прогресса, предоставить возможность занять ключевые позиции в новой, модернизированной науке и экономике России [...]». Возможно, именно следствием этой инициативы является достаточно большое количество вузов среди участников этой программы. Согласно интервью научного руководителя суперкомпьютерных программ «СКИФ» и «СКИФ-ГРИД» от России, директора Института программных систем им. А.К. Айламазяна РАН Сергея Абрамова, «в программе «СКИФ» было отдельное мероприятие, в 2000–2004 годах – подготовка и переподготовка кадров. Оно включало в себя написание учебников, разработку методических материалов и курсов, электронных учебников, предоставление их в открытом доступе (это же за бюджетные деньги сделано, отдавали без коммерции) и проведение летних школ по подготовке и переподготовке кадров. Туда приезжали и студенты, и аспиранты, и молодые ученые. Сегодня в программе «СКИФ-ГРИД» участвует большое количество вузов. И везде ведется серьезная работа по подготовке кадров [...]» [53]. Однако в открытых источниках не удалось найти какую-либо информацию о формате учебных мероприятий, их тематике (имеет ли место обучение грид-технологиям либо преподаётся только работа на кластерах и суперкомпьютерах), а также используемых ресурсах для практических занятий.

В качестве ещё одного российского проекта, использующего грид-технологии, можно назвать ГридННС, который был реализован в рамках федеральной целевой программы «Развитие инфраструктуры наноиндустрии в Российской Федерации на 2008 - 2010 годы» (мероприятие 2.1. - I очередь, лот 4). Цель проекта — «обеспечение географически распределенных научных и инженерных коллективов-участников национальной нанотехнологической сети (ННС) возможностью эффективного удаленного использования информационной, коммуникационной и вычислительной инфраструктуры. Национальная нанотехнологическая сеть – совокупность предприятий различных организационно-правовых форм, обеспечивающих и осуществляющих

скоординированную кооперативную деятельность по разработке и коммерциализации нанотехнологий, включая проведение фундаментальных и прикладных исследований, подготовку кадров, развитие инфраструктуры nanoиндустрии, организацию производства и непосредственное производство нанотехнологической продукции» [54]. В результате реализации данного проекта в России была создана грид-инфраструктура, объединяющая вычислительные ресурсы нескольких организаций и работающая на базе стека ППО, включающего сервисы Globus Toolkit и собственные разработки. Исследовательские работы, адаптация сторонних программных компонент и создание собственных, как и портирование прикладного ПО для работы в этой грид-среде выполнялись на отдельном полигоне. Однако в рамках данного проекта не проводилась какая-либо образовательная деятельность и (пере)подготовка кадров.

Похоже положение дел наблюдалось в проекте создания российской грид-сети для высокопроизводительных вычислений (РГС). РГС — «это территориально-распределенная телекоммуникационная автоматизированная система общегосударственного уровня для обеспечения доступа к ресурсам российских суперкомпьютерных центров при организации распределенных высокопроизводительных вычислений» [55]. Данная система «предназначена для предоставления научно-техническим и промышленным коллективам возможности удаленного выполнения высокопроизводительных вычислений и обработки данных на широком спектре географически разнесенных вычислительных ресурсов, от уникальных суперкомпьютеров до средних и малых суперкомпьютерных кластеров и отдельных серверов, за счет эффективного, прозрачного и безопасного доступа к сервисам и ресурсам грид-сети, а также за счет использования выделенных каналов высокоскоростного обмена данными» [там же].

Разработка отдельных программных компонент ППО, также как и адаптация прикладного ПО выполнялась несколькими организациями-участниками этого проекта и на отдельном полигоне. Однако, как и в ГридННС, в рамках данного

проекта не проводилась какая-либо образовательная деятельность и (пере)подготовка кадров.

Участие других государств СНГ в проектах в области грид-технологий, как правило, ограничивается участием в EGI (например, Украина и Беларусь), совместных с Россией программах (Беларусь в «СКИФ-ГРИД»), либо же они ещё находятся на этапе освоения этих технологий, в том числе при активной помощи от Объединенного института ядерных исследований, странами-участницами или ассоциированными членами которого они являются.

Ситуация с решением исследовательских задач в области облачных технологий отличается от ситуации в области грид, т. к. преимущественными пользователями в первом случае выступают коммерческие компании, что главным образом и определяет взаимоотношения между всеми участниками в этой области: разработчиками, поставщиками и потребителями. Компании могут играть как одновременно обе первые две роли, так и каждую по отдельности. Например, Microsoft является разработчиком своего гипервизора Hyper-V [56] и на базе его специальным образом модифицированной версии предлагает своё облачное решение Windows Azure [57], а аналогичные продукты таких крупных игроков на этом рынке, как Amazon и Google, построены на гипервизорах сторонних разработчиков – Xen [58] и KVM [59] соответственно.

Существуют также программные платформы для создания локальных облачных инфраструктур, лицензия которых позволяет бесплатное использование этого ПО. В качестве примеров можно привести OpenNebula [60], OpenStack [61]. Учебные курсы по использованию этих продуктов проводят коммерческие компании, как например, C12G Labs [62] в случае OpenNebula или Rackspace [63] по OpenStack.

Разработку и обучение компании выполняют на собственных полигонах и своим потребителям предлагают широкий набор обучающих курсов [64, 65, 66].

1.4. Требования к комплексам для исследовательских целей

Таким образом, для эффективного с точки зрения скорости достижения результата и использования компьютерных ресурсов решения исследовательских задач в области ФВЭ, связанных с применением облачных и грид-технологий, необходимы отдельные специализированные комплексы, которые должны удовлетворять следующим требованиям.

- 1) Возможность выполнения работ на различных типах ресурсов: суперкомпьютерах, кластерах, гетерогенных средах на базе персональных компьютеров, облачных средах, системах распределённого хранения данных.
- 2) Возможность одновременного решения исследовательских задач на различных программных платформах с возможностью расширения их набора.
- 3) Рациональное использование компьютерных ресурсов.
- 4) Возможность задавать срок доступа к ресурсам комплекса.
- 5) Набор сервисов каждой из грид-сред должен обеспечивать достаточную функциональность для решения задач инфраструктуры и не являться избыточным.
- 6) Комплекс должен быть автономным, т. е. независимым от каких-либо сервисов производственных грид-систем (например, от сервисов управления виртуальными организациями, сервисов по учёту потребленных ресурсов и т. д.). Такая автономность позволит задействовать подобные комплексы для максимально широкого круга задач за счёт отсутствия необходимости соблюдения многих формальных процедур, которые обязательны в случае производственных грид-систем и которые для указанных целей являются необоснованно избыточными и могут только уменьшить эффективность использования таких комплексов, а также сузить спектр решаемых на них задач.
- 7) Объём усилий для создания и поддержания работоспособности комплекса

должен быть соразмерен выполняемым на нём задачам и требуемой функциональности, а также обеспечивать необходимое и достаточное качество обслуживания.

На основании изложенного в первой главе можно сделать следующие выводы. В результате проведённого анализа выделено несколько возможных вариантов синтеза облачных и грид-технологий. Размещение грид-сервисов на виртуальных машинах в облачной среде позволяет повысить эффективность использования компьютерных ресурсов. Данная конфигурация наилучшим образом подходит для исследовательских задач, решение части из которых на производственных грид-системах невозможно, а остальных — менее эффективно с точки зрения скорости достижения результата и использования компьютерных ресурсов по сравнению с их решением на специализированных комплексах. Подобные комплексы должны удовлетворять сформулированным в этой главе требованиям.

Глава 2. Методы построения облачных грид-систем и реализация такой системы в ЛИТ ОИЯИ

2.1. Методы построения облачных грид-систем

Развитые в данной работе методы создания облачных грид-систем заключаются в следующем:

- 1) построение локальной облачной платформы с поддержкой виртуального разделения ресурсов на уровне операционной системы (последующие шаги 2-4 выполняются на виртуальных машинах облачной платформы);
- 2) создание локального УЦ для выдачи сертификатов пользователей и машин;
- 3) инсталляция сервиса управления виртуальными организациями с одной или несколькими отдельными ВО;
- 4) установка сервисов грид-полигонов на базе используемых ППО.

В качестве дополнительных аргументов в пользу применения технологии виртуализации можно привести следующие:

- 1) за счёт повышения эффективности использования компьютерных ресурсов виртуализация способствует снижению энергопотребления и уменьшению нагрузки на системы охлаждения т.е. имеет место один из аспектов так называемого «эко-компьютинга» — направления по минимизации негативного влияния на окружающую среду от применения информационных технологий, активно набирающее популярность в США и развитых странах Европы и поддерживаемое ведущими компаниями в области ИТ;
- 2) применение виртуальных машин в контексте грид-системы существенно упрощает её администрирование, в том числе предоставляя такие возможности, как установка виртуальных машин с грид-сервисами из заранее сконфигурированных шаблонов, «прозрачная» для пользователей миграция виртуальной машины с одного физического сервера на другой,

создание резервных копий и т. д.;

- 3) размещение виртуальных машин в облачной среде автоматизирует их распределение по серверам, т. е. упрощает работу администратора, а также даёт возможность дополнительно увеличить эффективность использования компьютерных ресурсов за счёт наличия планировщика и функции миграции виртуальных машин с одного физического сервера на другой.

При решении исследовательских задач может потребоваться одновременное наличие полигонов, функционирующих на базе разных ППО. Повысить эффективность использования компьютерных ресурсов можно также за счёт совместного использования некоторых сервисов различными средами. Например, сертификаты машин и пользователей для разных полигонов могут выдаваться в одном и том же удостоверяющем центре; перенастраивая среду окружения на интерфейсе пользователя, каждый пользователь может работать с соответствующим набором команд для нужной ему среды, т. е. использовать всего лишь одну машину для пользователей всех установленных ППО; сервис управления виртуальными организациями может быть задействован для авторизации пользователей в тех системах, службы которых поддерживают работу с ним.

2.2. Описание реализации облачной грид-системы в ЛИТ ОИЯИ

Для удовлетворения потребности в решении исследовательских задач в ЛИТ ОИЯИ была создана облачная грид-система [11, 13] с помощью описанных выше методов. В качестве программных платформ грид-сред используется несколько ППО, полигоны на базе которых по мере исчезновения необходимости в них удаляются либо создаются новые. Ниже приведено описание всего комплекса в целом и составляющих его полигонов.

Схематично структура комплекса изображена на рис. 3.

Выбор ПО для построения локальной облачной платформы осуществлялся в соответствии со следующими критериями:

- 1) поддержка необходимого типа виртуализации или наличие относительно простой возможности его интеграции;
- 2) совместимость с операционной системой (ОС) Linux и возможность запуска виртуальных машин с этой ОС, т. к. сервисы наиболее распространённых ППО грид работают под ней;
- 3) наличие понятной и доступной документации;
- 4) требуемая функциональность и качество программного продукта;
- 5) лицензия, допускающая модификацию и бесплатное использование;
- 6) поддержка со стороны разработчиков при необходимости модификации ПО.

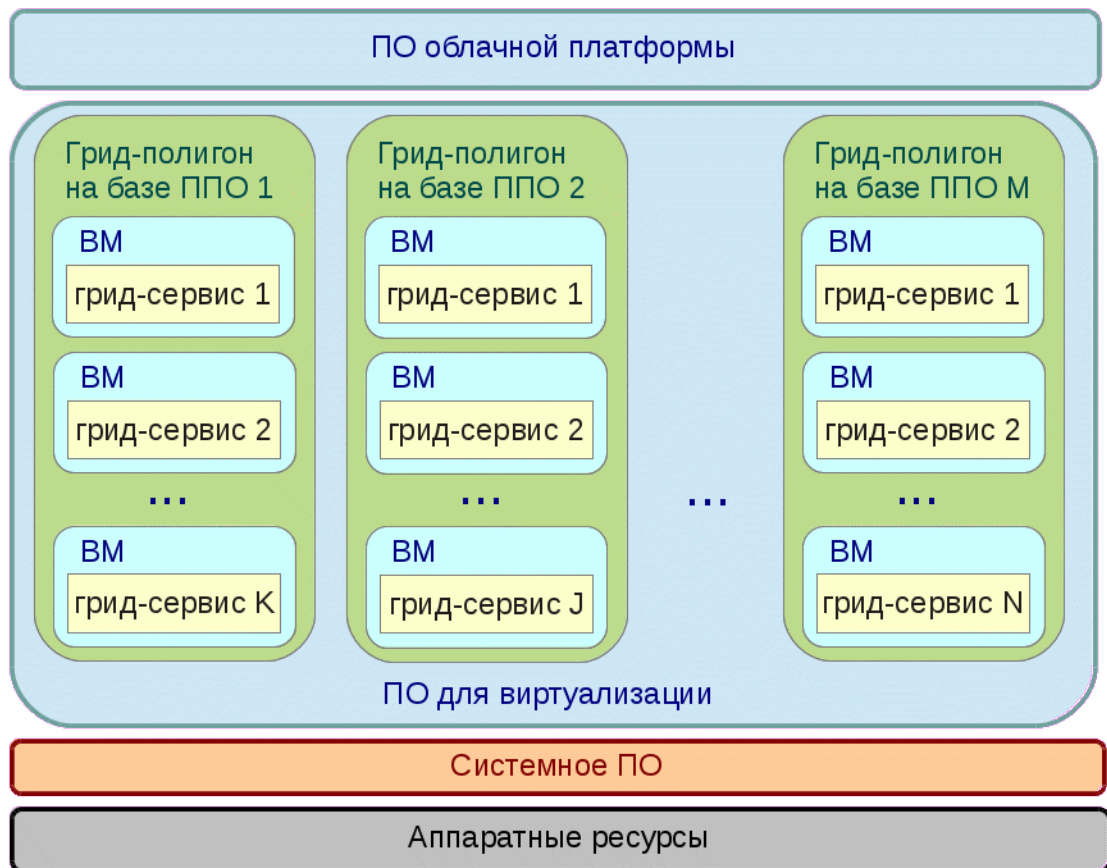


Рисунок 3. Схема структуры комплекса для исследовательских задач

Выбор ПО для виртуализации осуществлялся в соответствии со следующими критериями:

- 1) виртуализация на уровне операционной системы (т. к. эффективность использования физических серверов зависит в том числе и от накладных

расходов на виртуализацию, а из существующих на данный момент реализаций наиболее производительной, т. е. с минимальной нагрузкой на сервер, считается именно такая);

- 2) совместимость с ОС Linux;
- 3) возможность балансировки нагрузки на сервер посредством «живой» (т. е. без ощутимых для пользователей перерывов в работе) миграции виртуальных машин между серверами;
- 4) наличие консольных средств управления виртуальными машинами;
- 5) понятная и подробная документация;
- 6) необходимая функциональность и качество программного продукта;
- 7) лицензия, разрешающая бесплатное использование.

В результате проведённого анализа для виртуального разделения ресурсов был выбран программный продукт OpenVZ [67], а для создания облачных платформ – OpenNebula. Хотя ПО OpenNebula изначально не имело поддержки OpenVZ, но автором данной диссертации была написана первая версия драйвера [68] для упомянутой реализации виртуального разделения ресурсов, которая в дальнейшем была переработана коллегами из Центра суперкомпьютерных вычислений Национального технического университета Украины «Киевский политехнический институт» и актуализирована совместно с коллегами из ЛИТ ОИЯИ для более поздних версий OpenNebula [69].

На рис. 4 представлена общая схема облачной грид-системы ЛИТ ОИЯИ для исследовательских задач. Область с цифрой «1» и надписью «ЕМІ» обозначает полигон на базе одноимённого ППО и включает в себя фигуры с закруглёнными углами, символизирующие грид-сайты, входящие в состав этого полигона и размещённые в институтах стран-участниц ОИЯИ. Область с цифрой «2» и надписью «облако» ЛИТ ОИЯИ» содержит изображения полигонов на базе разных ППО, размещённые в облачной среде ЛИТ ОИЯИ.

Набор компонент данного комплекса непостоянен. Он меняется в зависимости от выполняемых на нём задач: при необходимости разворачиваются

новые полигоны или отдельные сервисы, а по окончании работ ненужные компоненты удаляются.

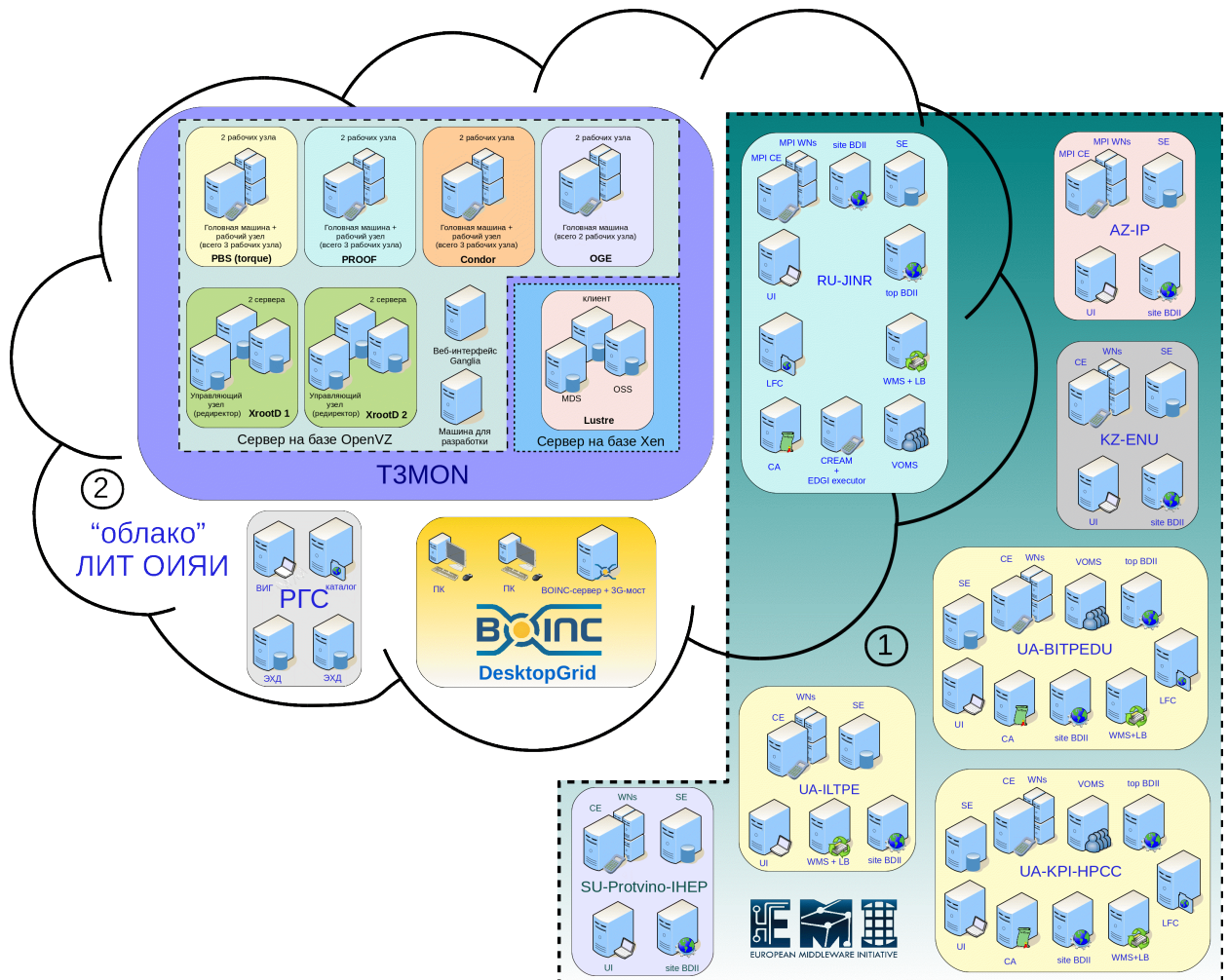


Рисунок 4. Общая схема облачной грид-системы ЛИТ ОИЯИ для исследовательских задач
Ниже приводится описание основных составляющих этого комплекса.

2.2.1. Полигон на базе ППО ЕМІ

Создание полигона на базе ППО ЕМІ выполнялось с учётом специфики исследовательских задач. Отличительные принципы построения и работы данного полигона [70, 71, 72] по сравнению с производственной грид-системой на базе этого же ППО (например, WLCG) изложены ниже, а также сведены в таблицу 2.

Во-первых, деятельность национальных и региональных удостоверяющих центров инфраструктуры WLCG санкционируется соответствующим органом управления политиками (Policy Management Authorities, PMAs), общие

рекомендации и правила для которых, в свою очередь, устанавливаются специальным органом – International Grid Trust Federation, IGTF [73]. Для решения исследовательских задач нет необходимости в такой иерархии и системе взаимоотношений. Достаточно, чтобы один УЦ обслуживал одну или несколько организаций-участников, а сами УЦ имели одинаковый ранг и признавали сертификаты друг друга.

Во-вторых, т. к. каждая из ВО производственной грид-системы проводит собственные курсы для пользователей с целью обучения их работе со специфическим для данной виртуальной организации ПО, то необходимость обучения таких целевых групп на исследовательской среде ЕМІ отпадает, а значит и отсутствует потребность в сервисах VOBOX, которые в инфраструктуре WLCG выполняют функции шлюза для ПО виртуальных организаций.

В-третьих, общее количество грид-сайтов производственной инфраструктуры WLCG на данный момент составляет более 300. Их список формируется автоматически на основе информации из базы данных операционного центра Грид (Grid Operations Center Database, GOCDB) и считывается информационными сервисами верхнего уровня (top-level Berkeley Database Information Index, tBDII). В исследовательском комплексе файл с подобным списком можно создавать и редактировать вручную, т. к. количество грид-сайтов полигона ЕМІ по сравнению с производственной грид-системой намного меньше (до десятка), а усилия по установке, настройке и поддержанию работоспособности сервиса GOCDB для такого их количества нецелесообразно.

В-четвёртых, в силу равного статуса всех организаций-участников этого распределённого полигона, предполагающего бесплатное использование ими данного комплекса для исследовательских целей, простоты и низкой ресурсоёмкости запускаемых задач, а также малого объёма передаваемых и хранимых данных, нет необходимости в системе учёта использования ресурсов, представленной в производственной инфраструктуре сервисами APEL.

В-пятых, для отслеживания текущего состояния сервисов среды достаточно

упрощённой системы мониторинга, основанной на данных из информационной системы и их визуализации. Наличие разнообразных подобных вспомогательных служб в производственной инфраструктуре (например, SAM, GridView, GStat) обусловлено важностью обеспечения намного более высокого качества обслуживания в том числе за счёт бесперебойной работы максимального количества ресурсов, что для полигона представляется избыточным.

В-шестых, небольшое (от двух-трёх до десятка) количество рабочих узлов на каждом из грид-сайтов полигона вполне достаточно для выполнения возлагаемых на него функций, т.к. количество одновременно запускаемых счётных задач невелико, они достаточно простые и не требуют больших вычислительных ресурсов, а также выполняются относительно быстро.

В-седьмых, небольшой объём передаваемых и хранимых данных (подавляющее большинство тестовых файлов не превышает нескольких мегабайт) и отсутствие необходимости в передаче файлов с очень высокой надёжностью позволяет отказаться от такого сервиса, как File Transfer Service (FTS), а также ограничить ёмкость элементов хранения данных до нескольких десятков гигабайт.

В-восьмых, отсутствие высоких требований к производительности грид-сервисов, включая вычислительные элементы и хранилища данных, а также проведённые эксперименты, подтверждающие полноценное их функционирование на виртуальных машинах, позволяют создавать грид-полигоны на базе ППО ЕМІ в виртуальных средах.

В-девятых, задачи пользователей исследовательской грид-среды довольно простые и выполняются относительно быстро, что позволяет каждому пользователю за время занятия запустить по несколько разнотипных задач и получить их результат. Продолжительность каждого занятия (от полутора часов в случае семинарских занятий студентов до 8 часов для курсов с более интенсивным обучением) меньше срока действия прокси-сертификата, с использованием которого задача была отправлена на счёт (по умолчанию, оно равно 12 часам). Данные обстоятельства позволяют отказаться от установки на

полигоне и использования сервиса MyProxy. В исключительных случаях, когда время счёта задачи не ограничено временем занятия (например, при тестировании разработчиком своего грид-приложения или сервиса), то проблема истечения срока действия прокси-сертификата пользователя до завершения задачи решается путём генерации прокси-сертификата с бóльшим временем действия, ограниченным лишь сроком действия сертификата атрибутов пользователя, заданным в конфигурации VOMS-сервера.

В-десятых, организация и обеспечение работоспособности полнофункциональных аналогов организационно-административных структур WLCG, подобных Regional operation center (ROC), для исследовательских полигонов представляются ресурсоёмкими и избыточными. Все организационно-консультационные вопросы, а также техническая поддержка администраторов ресурсных центров вполне может осуществляться одним человеком с использованием современных средств общения (электронной почты, сервисов обмена мгновенными сообщениями, программ для голосового общения и т.д.).

Таблица 2. Различия между производственными и исследовательскими грид-системами

Сервис	Производная грид-система	Исслед. грид-система	Пояснение значений для исследовательской грид-системы
УЦ	иерархия	одноранговые	локальные одноранговые УЦ
VOBOX	присутствует	отсутствует	обучение пользователей самими ВО
GOCDB	присутствует	отсутствует	малое кол-во грид-сайтов
APEL	присутствует	отсутствует	равный статус всех организаций-участников, бесплатное пользование ресурсами, простые нересурсоёмкие задачи
SAM,	присутствует	отсутствует	упрощённый мониторинг на

Сервис	Производная грид- система	Исслед. грид- система	Пояснение значений для исследовательской грид- системы
GridView, GStat			основе данных из информационной системы
Рабочие узлы	много	мало	кол-во одновременно выполняемых счётных задач невелико, быстрые нересурсоёмкие задачи
FTS	присутствует	отсутствует	небольшой объём передаваемых данных, отсутствие необходимости в передаче файлов с высокой надёжностью
SE	большого объёма	малого объёма	малый объём хранимых данных
MyProxy	присутствует	отсутствует	короткое время выполнения задач
ROC	присутствует	отсутствует	выполнение функций 1 человеком

Данный полигон изначально состоял из двух грид-сайтов, размещённых только на ресурсах ЛИТ ОИЯИ. В дальнейшем в него были интегрированы грид-сайты организаций стран-участниц ОИЯИ. Состав участников этого сегмента комплекса носит динамический характер: добавляются новые грид-сайты и прекращают своё участие (по различным причинам) интегрированные ранее.

Схема распределённого полигона на базе ППО ЕМІ в состоянии на момент написания текста диссертации представлена на рис. 5, а список названий организаций и размещённых в них грид-сервисов – в таблице 3.

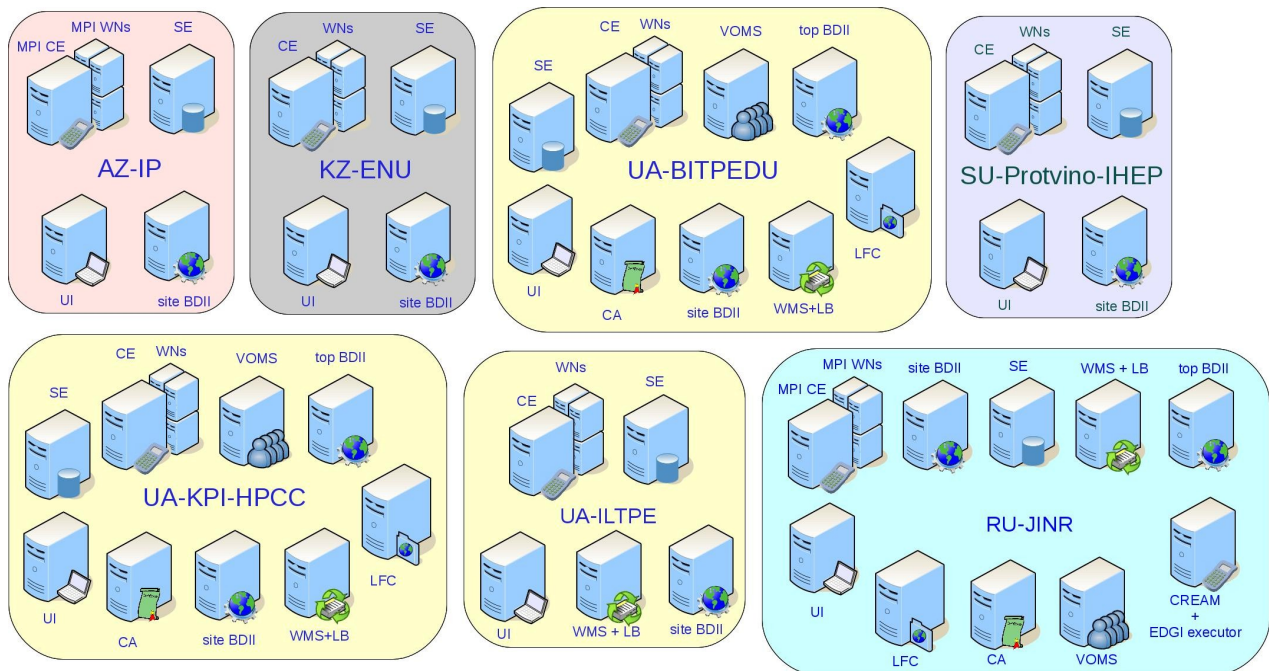


Рисунок 5. Схема распределённого полигона на базе ППО ЕМИ

Таблица 3. Список названий организаций и размещённых в них грид-сервисов распределённого полигона на базе ППО ЕМИ

Название грид-сайта	Название организации	Сервисы грид-сайта
RU-JINR	Объединённый институт ядерных исследований (г. Дубна, Московская область, Россия)	интерфейс пользователя (User Interface, UI), вычислительный элемент типа CREAM (MPI CE CREAM) с тремя рабочими узлами и с поддержкой счёта параллельных задач (MPI Worker Nodes, MPI WNs), хранилище данных типа DPM (Disk Pool Manager Storage Element, DPM SE), файловый каталог типа LCG (LCG File Catalogue, LFC), система управления загрузкой (Workload Management System, WMS), сервис сбора и хранения информации о задачах и их статусе

Название грид-сайта	Название организации	Сервисы грид-сайта
		(Logging&Bookkeeping Service, LB), информационный сервис о ресурсах грид-сайта (site BDII, sBDII), информационный сервис о грид-сайтах (top BDII, tBDII), сервис управления виртуальными организациями (Virtual organizations management service, VOMS)
SU-Protvino- ИИЕР	Институт физики высоких энергий (г. Протвино, Московская область, Россия)	UI, LCG-CE + 2 WNs, dCache SE, WMS, LB, sBDII;
UA- BITPEDU	Институт теоретической физики им. Н.Н. Боголюбова (г. Киев, Украина)	UI, CREAM + 24 WNs, DPM SE, LFC, WMS, LB, sBDII, tBDII;
UA-KPI- HPCC	Национальный технический университет Украины «Киевский политехнический институт» (г. Киев, Украина)	UI, LCG-CE + 8 WNs, DPM SE, LFC, WMS, LB, sBDII, tBDII.
KZ-ENU	Евразийский национальный университет им. Л.Н. Гумелёва (г. Астана, Казахстан)	UI, CREAM + 4 WNs, sBDII, DPM SE

Название грид-сайта	Название организации	Сервисы грид-сайта
UA-ILTPЕ	Физико-технический институт низких температур им. Б.И. Веркина Национальной академии наук Украины (г. Харьков, Украина)	UI, CREAM + 8WNs, sBDII, DPM SE, WMS+LB
AZ-IP	Институт физики Национальной академии наук Азербайджана (г. Баку, Азербайджан)	UI, CREAM + 4WNs, sBDII, DPM SE

На грид-сайтах данного полигона представлены как обычные кластеры, так и кластеры с поддержкой счёта параллельных задач (для имитации суперкомпьютеров). Также был установлен отдельный вычислительный элемент на базе CREAM с компонентой «EDGI executor» [74], которая позволяет выполнять задания, отправленные из среды EMI, на персональных компьютерах, используемых в качестве вычислительных ресурсов полигона DesktopGrid (см. п. 2.2.4).

2.2.2. Полигон T3MON

В связи с использованием локальных ресурсов организаций для повышения эффективности проведения конечного анализа данных с Большого адронного коллайдера (БАК) эксперимента ATLAS появилась задача мониторинга этих ресурсов (так называемых ресурсов уровня Tier-3 эксперимента ATLAS), решение которой стало целью проекта «T3MON» [75].

В архитектуре системы их мониторинга условно можно выделить две компоненты: средства локального мониторинга, устанавливаемые на каждый Tier-

3 сайт, и глобального мониторинга.

В качестве основы для построения системы локального мониторинга выбран программный продукт Ganglia. Для разработки модулей к этой системе на исследовательском грид-комплексе был создан полигон [12], в рамках которого были развёрнуты кластера на базе наиболее используемых на ATLAS Tier-3 сайтах средств управления локальными вычислительными ресурсами (PROOF, PBS, Condor, OGE/SGE) и хранения данных (XRootD, Lustre).

Компоненты этого полигона представлены на рис. 6 и описаны в таблице 4.

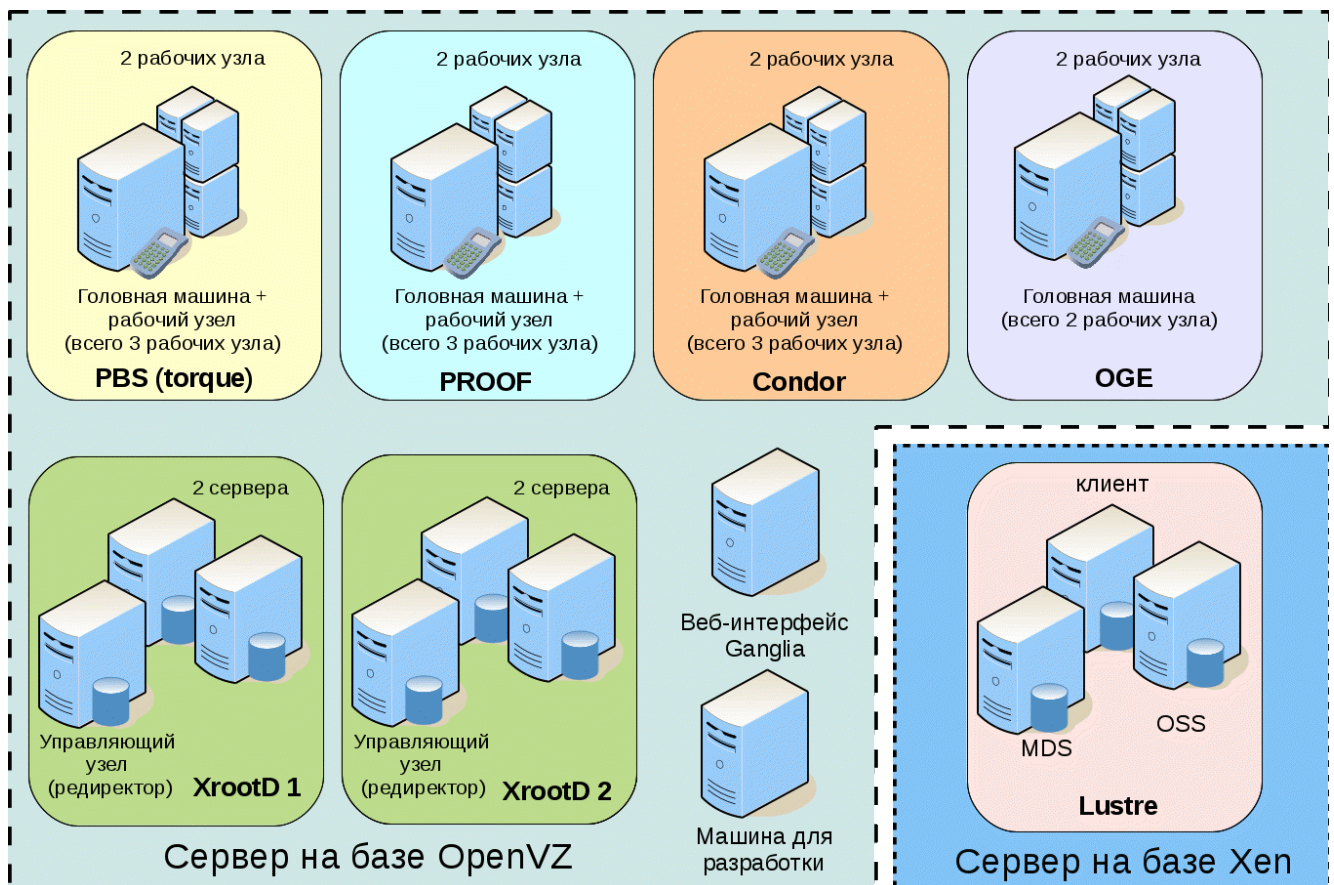


Рисунок 6. Схема компонент полигона ТЗМОН

2.2.3. Полигон РГС

Для выполнения обязательств ЛИТ ОИЯИ в рамках участия в проекте создания Российской грид-сети (РГС) для высокопроизводительных вычислений на исследовательском комплексе был установлен следующий набор сервисов:

- 1) веб-интерфейс пользователя для разработки проблемно-ориентированных интерфейсов (ПОИ) в виде расширений к нему;

- 2) элементы хранения данных на базе GridFTP;
- 3) виртуальная машина для разработки подсистемы хранения данных (ПХД) РГС.

Таблица 4. Компоненты полигона ТЗМОН

Название кластера	Компоненты
PBS	Головная машина (ГМ) кластера, на которой также установлены рабочий узел (РУ), система мониторинга Ganglia с компонентами gmond, gmetad и веб-интерфейсом, а также её расширение Job Monarch [76], два рабочих узла с установленной на каждой из них компоненте gmond
PROOF	ГМ и 2 рабочих узла с gmond на каждом
Condor	ГМ, на которой также установлен РУ и gmond, два РУ с gmond
OGE	ГМ с gmond, а также 2 РУ с gmond, на одном из которых установлен ещё и клиент
XRootD 1	Управляющий узел (редиректор) с компонентой gmond, 2 сервера с gmond
XRootD 2	Управляющий узел (редиректор) с компонентой gmond, 2 сервера с gmond
Lustre	Сервер метаданных с gmond и компонентой t3mon-site-lustre для передачи информации в систему Ganglia, 2 сервера хранения объектов с gmond, на одном из которых установлен ещё и клиент

Схематичное изображение этих же компонент представлено на рис. 7.

2.2.4. Полигон DesktopGrid

Одним из видов распределённых вычислительных ресурсов являются системы, в которых в качестве счётных узлов используются персональные компьютеры (такие среды ещё называют «DesktopGrid» [77]). Подобный тип ресурсов используется в большом количестве проектов для решения достаточно широкого круга задач [78, 79, 80].

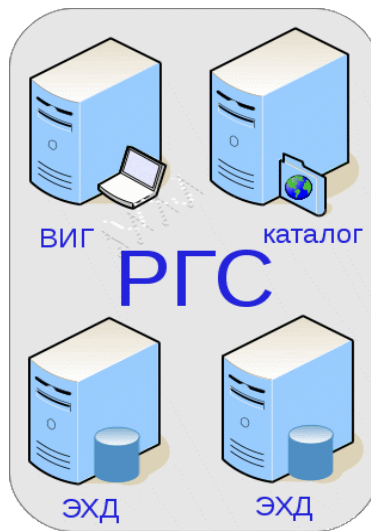


Рисунок 7. Схема компонент полигона РГС

Для апробации самой технологии, а также последующей адаптации в эту среду востребованных приложений на исследовательском комплексе был создан отдельный полигон. Из существующего ПО, позволяющего создавать подобные инфраструктуры, был выбран продукт VOINC, как один из наиболее стабильных и распространённых. Программная компонента 3G-мост [81], разработка института MTA SZTAKI, позволила интегрировать данный полигон с полигоном на базе EMI, что дало возможность использовать одни и те же средства для запуска задач в обеих средах. Этот сегмент включает в себя следующие составляющие (см. рис. 8):

- 1) VOINC-сервер с 3G-мостом;
- 2) виртуальная машина с предустановленным VOINC-клиентом;
- 3) вычислительный элемент CREAM, являющийся также частью полигона EMI, с программной компонентой «EDGI executor».

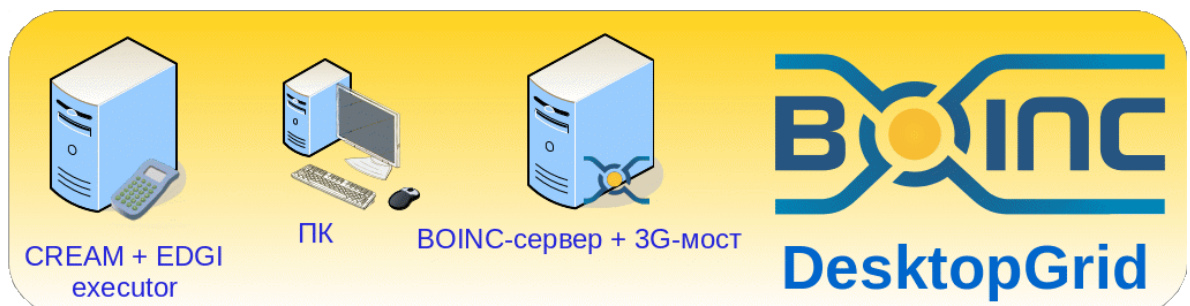


Рисунок 8. Схема компонент полигона DesktopGrid

2.2.5. Веб-портал комплекса

Для предоставления актуальной информации об исследовательском комплексе (новости, текущее состояние, проводимые работы, завершённые и выполняемые на ней задачи) на одной из виртуальных машин в облаке был развёрнут веб-портал (см. рис. 9), доступный по следующему URL: <http://gridedu.jinr.ru>.



Рисунок 9. Снимки экрана веб-портала облачной грид-системы

Он содержит описание распределённого полигона на базе ЕМІ и показывает на карте местонахождение организаций с грид-сайтами, входящими в его состав. На данном портале размещены инструкции по интеграции грид-сайтов организаций в общую среду на базе данного ППО, руководства для системных администраторов по установке сервисов ЕМІ на виртуальные машины OpenVZ и по созданию вычислительных элементов на базе ППО ЕМІ с поддержкой счёта параллельных задач. Помимо этого, размещена информация об ответственных за функционирование комплекса.

Таким образом, разработаны методы создания специализированных

комплексов с учётом сформулированных в первой главе требований. Эти методы позволяют повысить эффективность использования компьютерных ресурсов и сократить время решения исследовательских задач за счёт размещения грид-сервисов на виртуальных машинах в облачной среде и применения виртуализации на уровне операционной системы. На этой методологической основе в ЛИТ ОИЯИ был создан исследовательский комплекс.

Глава 3. Использование облачной грид-системы ЛИТ ОИЯИ

Созданная в ЛИТ ОИЯИ облачная грид-система была использована для решения различных задач в рамках выполнения обязательств Института как в различных международных и национальных проектах, так и перед его странами-участницами.

Ниже приводится описание задач, для которых использовался каждый из полигонов облачной грид-системы.

3.1. Полигон ЕМІ

Одной из наиболее востребованных и интенсивно используемых компонент комплекса является полигон на базе ППО ЕМІ. Он был успешно задействован и применяется для следующих задач.

3.1.1. Обучение пользователей

Обучение пользователей работе в среде ЕМІ заключается в проведении учебных курсов для коллег из ОИЯИ и его стран-участниц, семестровых практических занятий для студентов университета «Дубна» и Учебно-научного центра ОИЯИ, а также участников различных международных школ и практик.

Учебные курсы для пользователей варьируются по продолжительности, формату проведения и содержанию:

- краткосрочные (несколько полных дней интенсивных занятий с преподавателем);
- среднесрочные (от недели до месяца: часть дня отводится для обучения с преподавателем, а вторая часть — для самостоятельной работы);
- долгосрочные (больше месяца в виде семестровых занятий в образовательных учреждениях либо в форме консультаций посредством электронной почты, служб обмена мгновенными сообщениями или средств голосового общения).

Программа курса включает в себя следующие разделы:

- 1) введение в грид-технологии;
- 2) безопасность в Грид, получение сертификата пользователя;
- 3) информационная система;
- 4) управление задачами;
- 5) управление данными.

Степень детализации их содержания варьируется в зависимости от продолжительности курса. Для доступа к учебным материалам был разработан веб-портал с URL <http://grid-web.jinr.ru>.

Для выполнения практических упражнений в данной грид-среде каждый участник учебного мероприятия должен иметь учётную запись на специальном сервисе – интерфейсе пользователя (User Interface, UI), сертификат пользователя и являться членом виртуальной организации edu, которая поддерживается на всех грид-сайтах этого полигона. После получения обучаемыми учётных записей на интерфейсе пользователя они генерируют запрос на сертификат, который подписывается локальным УЦ. Далее пользователи должны запросить членство в ВО edu путём конвертирования выданного сертификата из формата x509 в формат PKCS12, импорта его в браузер, заполнением и отправкой соответствующей формы запроса на странице ВО. Администратор ВО уведомляется электронным письмом о поступлении новых запросов на членство. В случае одобрения или отказа пользователи получают соответствующее уведомление. По окончании курса членство пользователей в ВО edu аннулируется, а сертификаты отзываются УЦ.

Пользователи взаимодействуют с сервисами данной грид-среды посредством выполнения команд в командной строке на UI. Вход под своей учётной записью на UI осуществляется по протоколу ssh.

Список проведённых на данном полигоне курсов для пользователей приведён в таблице 5. Организация или мероприятие, город и страна, которые представляли обучаемые, указаны в соответствующих столбцах, расшифровка аббревиатур организаций приведена после таблицы, символ «*» означает

несколько городов или стран.

Таблица 5. Список проведённых курсов по обучению пользователей работе в средах gLite и EMI

Организация/мероприятие	Город	Страна	Даты
УК	Кейптаун	ЮАР	26.09.12 - 12.10.12
ИФ НАН	Баку	Азербайджан	23.07.12 - 27.07.12
ШИКТ МУНТ, ШИТ НУМ	Улан-Батор	Монголия	23.07.12 - 27.07.12
КУ	Каир	Египет	23.07.12 - 27.07.12
Участники международной студенческой практики	*	*	13.07.11 - 20.07.11
ЕНУ имени Л.Н. Гумилева	Астана	Казахстан	14.06.11 - 10.07.11
ШИКТ МУНТ, ШИТ НУМ	Улан-Батор	Монголия	17.05.11 - 31.05.11
Участники школы ОИЯИ/ЦЕРН	*	Россия	28.10.10 - 29.10.10
Участники практики студентов из Египта	*	Египет	17.05.10 - 04.06.10
ИМИТ НАН	Ташкент	Узбекистан	30.11.09 - 07.12.09

Пояснение аббревиатур организаций к этой таблице:

УК — Университет Кейптауна,

ИФ НАН — Институт физики Национальной академии наук Азербайджана,

ШИКТ МУНТ — Школа информационных и коммуникационных технологий Монгольского университета науки и технологии,

ШИТ НУМ — Школа информационных технологий Национального университета Монголии,

КУ — Каирский университет,

ЕНУ им. Л.Н. Гумилева — Евразийский национальный университет имени Л.Н. Гумилева,

ИМИТ НАН — Институт математики и информационных технологий Академии Наук Республики Узбекистан.

3.1.2. Обучение системных администраторов

Обучение системных администраторов представлено учебными курсами для коллег из ОИЯИ и организаций из его стран-участниц по установке, настройке и проверке работоспособности грид-сайтов на базе ППО ЕМІ.

Если обучение пользователей всегда проводилось преподавателем лично, то учебные курсы для системных администраторов грид-сервисов проводились в том числе в режиме видеоконференции, а также в виде консультаций посредством электронной почты, служб обмена мгновенными сообщениями или средств голосового общения.

Для этих курсов каждый обучаемый должен иметь в своём распоряжении набор серверов с правами администратора. При наличии у участников необходимых ресурсов (например, размещённых в их организациях) обучение проводилось с использованием этих ресурсов. Развёрнутые таким образом по окончании курса грид-сайты могли быть интегрированы в распределённый полигон исследовательского комплекса на базе ППО ЕМІ. Если же подобные ресурсы у обучаемых отсутствовали, каждому из них предоставлялся набор виртуальных машин, размещённых в облачной среде исследовательского комплекса ЛИТ ОИЯИ. По окончании курса эти виртуальные машины уничтожались.

Чтобы грид-сайт мог стать частью какой-либо из производственных грид-систем, он должен соответствовать определённым требованиям по вычислительной мощности рабочих узлов, объёму оперативной памяти на ядро, пропускной способности внешнего канала, объёму элементов хранения данных, надёжности работы грид-сервисов и ресурсов, а также их доступности. До тех пор, пока грид-сайт не удовлетворяет этим требованиям, он может являться частью распределённого исследовательского полигона ЕМІ.

Список проведённых курсов для системных администраторов представлен в таблице 6.

Пояснение аббревиатур организаций к этой таблице, которые не были

разъяснены выше.

ИТФ им. Н.Н. Боголюбова НАН — Институт теоретической физики имени Н.Н. Боголюбова Национальной академии наук Украины,

НТУУ «КПИ» — Национальный технический университет Украины «Киевский политехнический институт»,

ФТИНТ им. Б.И. Веркина НАН — Физико-технический институт низких температур им. Б.И. Веркина Национальной академии наук Украины,

НИИРИМТ — Национальный институт для исследования и развития изотопных и молекулярных технологий,

ННУЦ ФЧВЭ БГУ — Национальный научно-учебный центр физики частиц и высоких энергий Белорусского государственного университета.

Таблица 6. Список проведённых курсов по обучению системных администраторов

Организация	Город	Страна	Даты	Способ проведения
ШИТ НУМ	Улан-Батор	Монголия	2013	эл. почта
ШИКТ МУНТ	Улан-Батор	Монголия	2013	эл. почта
ИФ НАН	Баку	Азербайджан	23.07.12 - 27.07.12	лично
КУ	Каир	Египет	23.07.12 - 27.07.12	лично
ИТФ им. Н.Н. Боголюбова НАН, НТУУ «КПИ»	Киев	Украина	24.09.12 - 28.09.12, 05.09.11 - 09.09.11	лично
ИФ НАН	Баку	Азербайджан	13.07.11 - 31.08.11	лично
ЕНУ имени Л.Н. Гумилева	Астана	Казахстан	14.06.11 - 10.07.11	лично
ШИКТ МУНТ, ШИТ НУМ	Улан-Батор	Монголия	17.05.11 - 20.06.11	лично
ФТИНТ им. Б.И. Веркина НАН	Харьков	Украина	2011	эл. почта

Организация	Город	Страна	Даты	Способ проведения
ИМИТ НАН	Ташкент	Узбекистан	30.11.09 - 13.12.09	лично
ИТФ им. Н.Н. Боголюбова НАН, НТУУ «КПИ»	Киев	Украина	09.11.09 - 27.11.09	видеоконференция
НИИРИМТ	Клуж-Напока	Румыния	15.10.09 - 30.10.09	лично
УК	Кейптаун	ЮАР	01.06.08 - 13.06.08	лично
ННУЦ ФЧВЭ БГУ	Минск	Беларусь	2008	электронная почта

3.1.3. Обучение разработчиков

Обучение разработчиков осуществляется по мере необходимости привлечения новых кадров для участия в том или ином проекте либо по запросу от стран-участниц и партнёров ОИЯИ. Так например, в рамках обучения грид-технологиям коллег из Египта, помимо курсов для пользователей и системных администраторов, также был проведен курс по адаптации приложений для запуска в среде ЕМІ. Во время данного мероприятия были изложены как общие принципы, касающиеся типов вычислительных ресурсов и подходящих для них приложений, так и проведено портирование конкретного прикладного пакета в эту среду с последующим тестированием.

3.1.4. Выполнение обязательств по проекту WLCG

Полигон ЕМІ также использовался для выполнения работ в рамках участия ОИЯИ в проекте WLCG. В эти работы входило следующее:

- проверка функционирования различных типов вычислительных элементов с поддержкой MPI, сконфигурированных при помощи утилиты YAİM, а также сертификация патчей для них;

- создание набора функциональных тестов для интерфейса программирования приложений на языке Perl для сервиса LFC.

Для минимизации влияния проводимых работ на сервисы полигона дополнительно устанавливались новые экземпляры соответствующих сервисов (вычислительные элементы CREAM и рабочие узлы с поддержкой MPI, а также LFC).

Для запуска тестов использовался UI этого полигона, а также сервисы WMS, LB и элементы хранения данных.

3.1.5. Адаптация приложений

С использованием полигона EMI также были разработаны методы адаптации прикладных пакетов для грид-сред на базе этого ППО [14]. С использованием данных методов на этом полигоне был адаптирован набор приложений. Детальная информация изложена в п. 3.4.1.

3.2. Полигон ТЗМОН

Программные компоненты для сбора данных мониторинга (т. н. «коллекторы») кластеров этого полигона, а также вся цепочка обработки этих данных должны быть соответствующим образом верифицированы. Основными данными для мониторинга являются операции над задачами и файлами, которые на грид-сайтах обычно выполняются пользователями. В связи с этим возникает необходимость имитации на этом полигоне деятельности пользователей, т. е. выполняемых ими типовых действий во время выполнения анализа данных эксперимента, к котором можно отнести следующие:

- 1) над задачами — произвольная отправка задач на счёт с конфигурируемой частотой, потреблением ОЗУ и загрузкой ЦПУ;
- 2) над файлами — загрузка файла произвольного размера на элемент хранения данных в произвольный момент времени, проверка существования файла, а также его удаление по истечении некоторого времени.

Для реализации подобного поведения был разработан набор утилит для

каждого из развёрнутых кластеров. Он позволяет задавать такие параметры, как время запуска, размер файла, объём потребляемой оперативной памяти, загрузку ЦПУ и т. д. Указанные параметры операций по умолчанию имеют непрерывное равномерное распределение, а максимальные значения могут быть отрегулированы в соответствии с конфигурацией каждого кластера.

Этот набор для имитации деятельности пользователей написан на языках Python и Bash. Работа с ним осуществляется посредством интерфейса командной строки. Генерация событий выполняется системным планировщиком cron.

На базе полигона ТЗМОН были разработаны коллекторы для кластеров Lustre, PBS, XRootD и PROOF. Собранные этими модулями локальные данные передаются в систему глобального мониторинга WLCG — Dashboard [82]. На завершающем этапе находится создание аналогичных модулей для Condor и OGE. Модули для XRootD и PROOF проходят проверку на реальных кластерах в Лаборатории ядерных проблем (ЛЯП) ОИЯИ, а также BNL [83], Duke [84] и DESY [85].

С помощью разработанных коллекторов также мониторируются кластера этого полигона. Снимки экрана веб-интерфейса Ganglia с соответствующей информацией о кластерах представлены на рис. 10.

Помимо выполнения на этом полигоне работ в рамках проекта «ТЗМОН», он также был задействован для адаптации системы управления загрузкой эксперимента ATLAS — PanDA [86, 87] — для работы с суперкомпьютерами.

3.3. Полигон PFC

На данном полигоне была разработана подсистема хранения данных PFC на базе сервиса GridFTP с возможностью каталогизации исходных данных и результатов расчетов, обеспечивающая аутентификацию и авторизацию пользователей с использованием цифровых сертификатов формата X.509.

В этом сервисе реализована возможность публикации данных как для отдельных пользователей, так и для их групп, обладающих соответствующими

ПОЛНОМОЧИЯМИ.



Рисунок 10. Снимки экрана веб-интерфейса Ganglia с информацией о кластерах полигона ТЗМОН

Кроме того, предоставляется возможность интерактивного доступа к ПХД через веб-интерфейс для обмена файлами между локальным компьютером пользователя и хранилищем, а также между двумя хранилищами.

Веб-интерфейс ПХД, снимок экрана начальной страницы которого представлен на рис. 11, обеспечивает следующие функции:

- авторизация пользователей на основе цифровых сертификатов X.509;
- приём и передача данных с использованием протоколов GridFTP и HTTPS;
- управление файлами на ЭХД (удаление, репликация, контроль доступа);
- прямой обмен файлами между хранилищами;
- поиск файлов по нескольким хранилищам;
- резервирование и отказоустойчивость, возможность прозрачной работы со вторичными репликами данных при отказе первичного хранилища;

- создание логических наборов данных, операции над наборами данных.

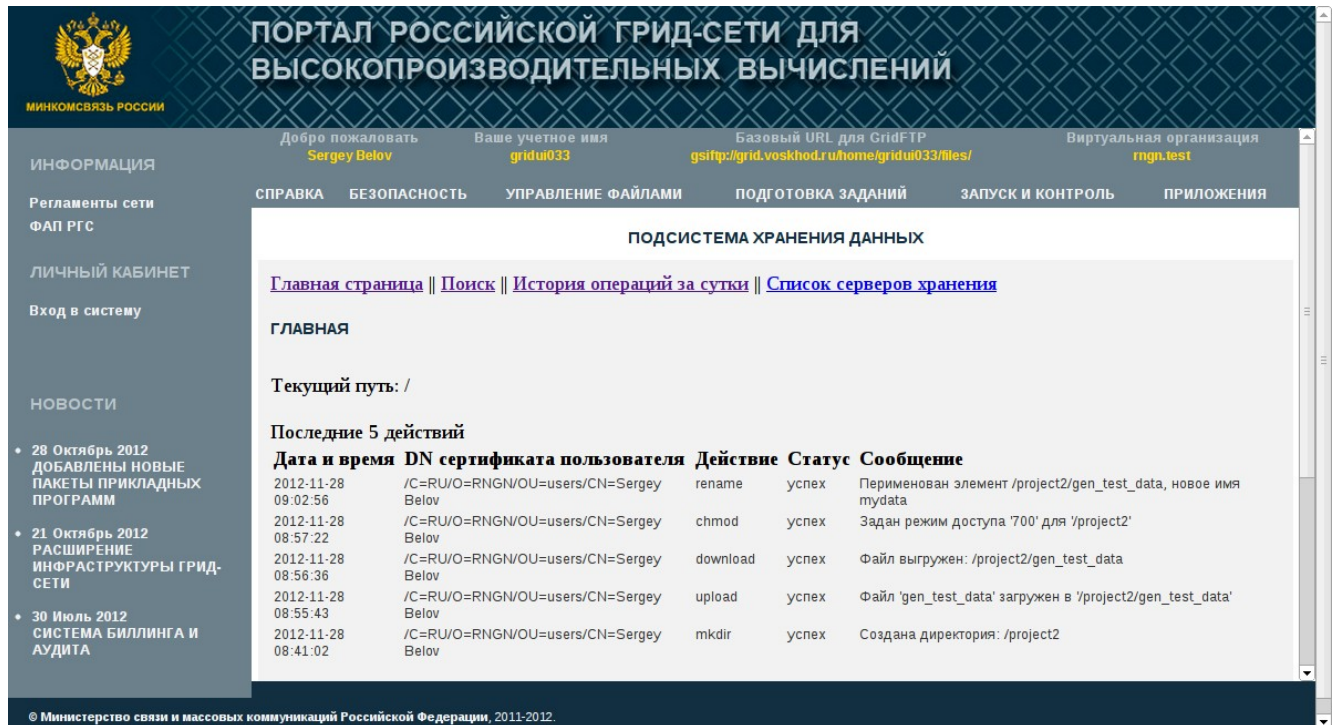


Рисунок 11. Снимок экрана начальной страницы веб-интерфейса ПХД РГС

С использованием полигона РГС также были разработаны методы адаптации прикладных пакетов для этой грид-среды [14]. С помощью данных методов на этом полигоне был адаптирован набор приложений. Детальная информация изложена в п. 3.4.2.

3.4. Методы адаптации определённого класса приложений для грид-сред на базе ЕМІ и РГС

Набор действий по адаптации конкретного прикладного программного пакета (ППП) для использования в той или иной грид-инфраструктуре зависит от нескольких факторов, среди которых ключевыми можно назвать следующие:

- 1) средства взаимодействия пользователя с грид-средой для запуска задач;
- 2) тип вычислительных ресурсов, пригодных для работы адаптируемого ППП и представленных в конкретной грид-инфраструктуре.

Оба фактора зависят от ППО, на базе которого построена конкретная грид-инфраструктура.

На данный момент существует достаточно много ППО, реализующего

концепцию Грид. Оно различается по используемым языкам программирования, по набору сервисов и предоставляемой функциональности, типу вычислительных ресурсов и элементов хранения данных, поддерживаемым операционным системам, условиям распространения и использования и т. д.

Для работы пользователя в грид-среде ему предоставляется один или несколько интерфейсов. Например, интерфейс командной строки, где взаимодействие пользователя с грид-инфраструктурой осуществляется путём вызова команд с определённым набором параметров. Для запуска задачи посредством такого интерфейса, как правило, пользователь должен предварительно составить файл её описания на понятном данной грид-среде языке. В качестве примеров таких интерфейсов можно привести интерфейс командной строки ГридННС – ПИКС [88], интерфейс командной строки ЕМІ, Globus Toolkit или PGC. Другим возможным способом взаимодействия пользователя с грид-средой является веб-интерфейс, как например ВИГ [89] (веб-интерфейс ГридННС и PGC). Также есть разработки, предоставляющие пользователям единый веб-интерфейс к различным грид-средами. В качестве примера такого можно привести упоминавшийся ранее портал WS-PGRADE.

Помимо этого, грид-сервисы различных ППО имеют интерфейсы программирования приложений (application programming interface, API), используя которые можно создать собственные интерфейсы, обеспечивающие необходимую конечным пользователям функциональность для взаимодействия с грид-средой.

Что же касается вычислительных ресурсов, представленных в грид-инфраструктурах, то в них можно условно выделить следующие типы:

- 1) суперкомпьютеры – высокопроизводительные вычислительные комплексы для решения сильносвязанных параллельных задач;
- 2) кластеры – вычислительные системы для обработки независимых последовательных задач;
- 3) гетерогенные среды, в качестве вычислительных ресурсов которых выступают персональные компьютеры (т. н. «DesktopGrid»).

В конкретной грид-инфраструктуре могут быть представлены как один, так и несколько или даже все из перечисленных выше типов вычислительных ресурсов.

Подходящий тип вычислительных ресурсов (а значит и процедура его интеграции в грид-среду) для конкретного приложения зависит от возможных режимов его работы, а именно:

- 1) параллельный, когда выполняющие одну задачу процессоры могут обмениваться между собой сообщениями, например, посредством библиотек MPI;
- 2) последовательный, когда каждый процессор выполняет независимую часть задачи.

Для использования приложений в параллельном режиме подходят суперкомпьютеры и кластеры с поддержкой счёта параллельных задач, тогда как для последовательных задач возможно использование всех типов вычислительных ресурсов, хотя некоторые из них не всегда оптимальны.

Действия по адаптации приложений для работы в среде грид зависят как от режима работы пакета, так и от способа запуска задач.

Предлагаемые в данной работе методы адаптации ППП для грид-сред на базе ППО ЕМІ и РГС предполагают, что приложения удовлетворяют следующим требованиям:

- 1) совместимость прикладного пакета с операционной системой (ОС) семейства Linux, т. к. ОС этого типа установлены на рабочих узлах ресурсных центров соответствующих грид-инфраструктур;
- 2) возможность вызова приложения из командной строки, т. к. непосредственный его вызов на рабочих узлах осуществляется из скриптов;
- 3) отсутствие необходимости интерактивного ввода какой-либо информации, т. к. у пользователя нет возможность её предоставить при таком способе запуска;
- 4) возможность передачи параметров вызова приложения и входных данных

через аргументы в командной строке и (или) в одном или нескольких отдельных файлах.

Общие положения методов адаптации ППП для обоих грид-сред можно сформулировать следующим образом:

- 1) проверка приложения на соответствие приведённым выше требованиям;
- 2) создание вспомогательного скрипта (файла-сценария), осуществляющего запуск приложения с нужными параметрами, а также при необходимости выполняющего набор операций до и после вызова прикладного пакета;
- 3) разработка файла описания задания на соответствующем языке конкретной грид-среды;
- 4) размещение приложения на рабочих узлах кластера одного из грид-сайтов;
- 5) внесение соответствующей информации об установленном пакете в информационную систему грид-сайта.

Однако, каждая из грид-сред имеет свою специфику, в связи с чем есть некоторые отличия в методах адаптации приложений для каждой из них.

3.4.1. Методы адаптации приложений для грид-среды на базе ППО ЕМІ

Методы адаптации приложений для грид-сред на базе ППО ЕМІ различаются в зависимости от режима работы портируемого пакета.

Если приложение работает в последовательном режиме, то в общем случае необходимо выполнить следующие шаги.

- 1) Создать скрипт, предназначенный для исполнения непосредственно на рабочих узлах кластера ВЭ и осуществляющий такие действия, как
 - предварительную подготовку (при необходимости) до вызова приложения (например, извлечение файлов со входными данными из архива, проверку соблюдения обязательных для работы пакета условий и т. п.);
 - запуск приложения с необходимыми параметрами;
 - пост-обработку (упаковка результатов в архив и т. п.).

- 2) Создать файл описания задачи на языке JDL [90], где указать в атрибуте Executable имя файла скрипта, сформированного на первом шаге, и его же в атрибуте InputSandbox для передачи на рабочие узлы. Кроме того, в этом же атрибуте необходимо перечислить набор файлов со входными данными либо их архив.
- 3) Сделать доступным на рабочих узлах адаптируемое приложение (обычно, его размещают в сетевой директории, смонтированной на всех рабочих узлах). Как правило, его устанавливает пользователь, являющийся членом конкретной ВО и имеющий в ней роль менеджера программной группы (software group manager, sgm). У него есть необходимые права доступа в данную директорию, в том числе и на запись.
- 4) Также этот sgm-пользователь ответственен за внесение соответствующих данных об установленных пакетах, которая затем публично доступна из информационной системы грид-сайта. Для этих целей используются специальные утилиты lcg-tags и lcg-ManageVOTag.

Чтобы не перечислять имена всех файлов со входными данными в файле описания задачи (jdl-файле) удобно их поместить в архив. Использование одного и того же имени архива для разных наборов файлов со входными данными позволяет избавиться от необходимости менять его имя в атрибуте InputSandbox. Если же файл со входными данными для данного приложения один и он относительно небольшого объёма, тогда нет необходимости его архивировать, хотя по-прежнему удобно использовать одно и то же имя, чтобы не редактировать jdl-файл при изменении имени входного файла.

С использованием описанных выше методов было адаптировано приложение Blender [91] (бесплатный программный пакет с открытым исходным кодом для создания 3D-моделей и работы с ними). Содержимое jdl-файла и скрипта для этого приложения приводятся ниже в качестве примера.

Содержимое файла blender.jdl:

```
Executable = "blender.sh";
```

```
StdOutput = "std.out";
StdError = "std.err";
OutputSandbox = {"std.out", "std.err", "result.tar.gz"};
InputSandbox = {"blender.sh", "input.blend"};
```

Содержимое файла `blender.sh`:

```
#!/bin/bash
blender -b input.blend -o //blender -f 1
tar -czf result.tar.gz blender*.jpg
rm -rf blender*.jpg
```

В данном примере в качестве входного файла используется файл с именем `input.blend`, выходные данные будут упакованы в файл `result.tar.gz`, содержащий все файлы с расширением `jpg`, созданные в текущей директории задачи в результате её запуска на рабочем узле.

Возможность разбиения некоторых ресурсоёмких вычислительных операций в ППП Blender на независимые части (например, визуализацию анимации можно разбить на более мелкие итерации – обработку одного или нескольких кадров, и выполнять каждую из них независимо как отдельную задачу), позволяет использовать данное приложение на всех трёх типах вычислительных ресурсов грид-сред.

Методы адаптации параллельных приложений имеют определённые отличия в части пунктов 1 и 2, описанных в методах выше.

Если приложение поддерживает работу в параллельном режиме (например, с помощью библиотеки MPI), тогда для его запуска в среде ЕМІ удобно использовать пакет `mpi-start` [92]. Он представляет собой набор вспомогательных скриптов, предоставляющих единый интерфейс для запуска параллельных задач в различных средах выполнения и позволяющий пользователю облегчить запуск MPI-приложений в грид-инфраструктурах на базе ППО ЕМІ.

Обычно данный пакет предустановлен на рабочих узлах. Однако в случае его отсутствия, пользователь может переслать архив с ним на рабочие узлы вместе со входными файлами для своей параллельной задачи, распаковать этот архив и

установить соответствующую переменную среды окружения I2G_MPI_START в значение \$PWD/bin/mpi-start.

Использование `mpi-start` предъявляет определённые требования к составлению файла описания задачи и предполагает применение одного или двух вспомогательных скриптов. Они устанавливают соответствующие переменные среды окружения и обеспечивают вызов `mpi-start` с параметрами для корректного запуска пользовательского приложения (как правило, данный функционал реализуют в скрипте с именем `mpi-start-wrapper.sh`, хотя имя жёстко не фиксировано и может быть любым). Также эти скрипты указывают необходимые к выполнению до и после непосредственного вызова приложения команды с нужными параметрами (обычно задаются в файле с именем `mpi-hooks.sh`).

Скрипт `mpi-start-wrapper.sh` подходит для запуска большинства параллельных приложений и, как правило, нет необходимости его модифицировать. В файле `mpi-hooks.sh` в функциях `pre_run_hook` и `post_run_hook` указывают набор команд, которые должны быть выполнены до и после запуска приложения соответственно.

В `jdl`-файле необходимо указать значения для следующих атрибутов:

- `JobType` — `Normal`, несмотря на то, что это параллельная задача (необходимые действия по вызову приложения в параллельном режиме будут выполнены пакетом `mpi-start`);
- `CPUNumber` должно быть равно количеству процессоров, необходимому для работы параллельного приложения;
- `Executable` должно иметь в качестве значения имя файла исполняемого скрипта `mpi-start-wrapper.sh`, т.к. именно он будет первым вызван на рабочем узле;
- `Arguments` должно содержать тип библиотеки MPI, исполняемый файл параллельного приложения и аргументы для него;
- в `InputSandbox` необходимо перечислить имена всех файлов, которые нужно передать на рабочие узлы, включая вспомогательные скрипты и файлы со

входными данными;

- в OutputSandbox указываются файлы, которые должны быть доставлены пользователю (обычно это файлы стандартного вывода и ошибок, а также файлы с выходными данными);
- в качестве значений для атрибута Requirements обычно задают требования, которые позволяют выбрать сервису WMS подходящий вычислительный ресурс (в случае использования пакета mpi-start указывают необходимость его наличия на кластере, а также нужную реализацию библиотеки MPI);
- в Environment задаются дополнительные переменные среды окружения (например, при необходимости выполнения каких-то операций до или после запуска параллельного приложения и использования для этого mpi-hooks.sh, посредством соответствующих переменных можно обеспечить вызов нужных команд).

С использованием разработанных методов адаптации параллельных приложений для работы в среде EMI были портированы следующие пакеты:

- DL_POLY [93] – пакет программ общего назначения для моделирования процессов в области классической молекулярной динамики, разработанный в Лаборатории Даресбари И.Т. Тодоровым и У. Смитом;
- Elmer [94] – универсальная программная система с открытым исходным кодом для анализа методом конечных элементов широкого спектра физических моделей;
- Molpro [95] – набор программ ab initio для расчёта электронной структуры молекул с высокой точностью;
- MEER (MIT Electromagnetic Equation Propagation) [96] – программный пакет для моделирования электромагнитных систем методом конечных разностей во временной области. Наиболее часто используется для расчета спектров отражения и прохождения, расчета резонансных частот и соответствующих им мод, расчета распределений полей, возникающих в результате воздействия произвольных источников.

- FDS (Fire Dynamics Simulator) [97] – программный пакет, реализующий вычислительную гидродинамическую модель тепломассопереноса при горении. FDS численно решает уравнения Навье-стокса для низкоскоростных температурно-зависимых потоков. В данном ППП особое внимание уделяется распространению дыма и теплопередаче при пожаре.

В качестве примера ниже приведено содержимое jdl-файла и вспомогательных скриптов для приложения FDS.

Содержимое файла `fds.jdl`:

```
JobType = "Normal";
CPUNumber = 3;
Executable = "mpi-start-wrapper.sh";
Arguments = "OPENMPI fds_mpi input.fds";
InputSandbox = {"mpi-start-wrapper.sh", "input.fds", "mpi-
hooks.sh"};
OutputSandbox = {"std.out", "std.err", "output.tgz"};
StdOutput = "std.out";
StdError = "std.err";
Requirements = member("MPI-START",
other.GlueHostApplicationSoftwareRunTimeEnvironment)
&& member("OPENMPI",
other.GlueHostApplicationSoftwareRunTimeEnvironment);
Environment = {"I2G_MPI_POST_RUN_HOOK=hooks.sh"};
```

Содержимое файла `mpi-start-wrapper.sh`:

```
#!/bin/bash
MPI_FLAVOR=$1
MPI_FLAVOR_LOWER=`echo $MPI_FLAVOR | tr '[:upper:]'
'[:lower:]'`
export I2G_MPI_TYPE=$MPI_FLAVOR_LOWER
shift
export I2G_MPI_APPLICATION=$1
shift
export I2G_MPI_APPLICATION_ARGS=$*
```

\$I2G_MPI_START

Содержимое файла mpi-hooks.sh:

```
#!/bin/sh
post_run_hook () {
    if [ $? ]; then
        tar -czf output.tgz * --exclude *.fds
    else
        echo "FDS has finished with errors"
        exit 1
    fi
    return 0
}
```

Все адаптированные для среды на базе ППО ЕМІ приложения, описанные выше, запускаются из командной строки интерфейса пользователя.

3.4.2. Методы адаптации приложений для грид-среды на базе ППО РГС

Одной из задач в проекте создания российской грид-сети (РГС) для высокопроизводительных вычислений, инициированном Министерством связи и массовых коммуникаций Российской Федерации, с целью облегчения работы пользователей в этой инфраструктуре являлась разработка специализированных проблемно-ориентированных веб-интерфейсов (ПОИ) для набора прикладных пакетов, которые бы позволяли упростить формирование и запуск счётных заданий, использующие конкретный ППП, контроль хода их выполнения и получение результатов вычислений.

В инфраструктуре РГС в качестве вычислительных ресурсов используются суперкомпьютеры и кластеры с поддержкой счёта параллельных задач.

Использование ППП в данной среде возможно посредством запуска задач как из интерфейса командной строки, так и с помощью веб-интерфейса (ВИГ) либо специальных модулей расширения (плагинов) к нему, специфичных для каждого конкретного прикладного пакета.

Набор действий при адаптации последовательного приложения для работы в

РГС подобен описанному в начале п. 3.4.1. Однако есть некоторые отличия, которые представлены ниже.

В РГС различают понятия «задание» и «задача». Задание — это совокупность задач, представленная в виде направленного ациклического графа. Задачи в рамках одного задания могут быть иметь различные взаимозависимости.

Для составления файлов описания заданий и задач в РГС используется формат JSON [98], из-за чего их содержимое имеет другой вид.

Так в качестве значения атрибута `executable` рекомендуется указать абсолютный путь к исполняемому файлу вызова оболочки, для которой написан вспомогательный скрипт. Это связано с тем, что все входящие файлы, указанные в атрибуте `input_files`, скачиваются ВЭ с помощью соответствующих утилит по протоколу `gsiftp`. Полученные таким способом файлы не имеют прав на исполнение (соответствующий бит установлен в 0). В значении атрибута `arguments` указывается имя вспомогательного файла-скрипта, которое также перечисляется в атрибуте `input_files` среди всех входных файлов, включая файлы с данными. Все эти файлы (вспомогательный скрипт, а также файлы со входными данными) должны быть доступны ВЭ по сети по протоколу `gsiftp`, а путь к директории с ним необходимо указать либо в значении атрибута `default_storage_base`, либо для каждого файла отдельно в атрибуте `input_files`.

В атрибуте `output_files` нужно перечислить имена всех файлов, которые необходимо сохранить по результатам работы приложения.

Требования к установленному на ВЭ ПО необходимо задать с помощью вложенного атрибута `software` основного атрибута `requirements`.

Помимо этого, прикладное ПО на рабочих узлах ВЭ устанавливает системный администратор соответствующего грид-сайта, а также прописывает профили окружения для этого ПО и заносит необходимую о нём информацию в информационную систему.

Файлы со входными и выходными данными также удобно размещать в архивах, соответствующие операции над которыми выполняет вспомогательный

скрипт.

В качестве примера ниже приведено содержимое файла описания задания и вспомогательного скрипта для приложения GEANT4-DNA.

Содержимое файла описания задания `geant4-dna.js`:

```
{ "version": 2,
  "description": "GEANT4-DNA",
  "default_storage_base":
"gsiftp://rgn.jinr.ru/shared/home/kut/geant4-dna/",
  "tasks": [ { "id": "a",
    "description": "task #1",
    "definition": { "version": 2,
      "executable": "/bin/sh",
      "arguments": ["geant4-dna.sh"],
      "stdout": "stdout.txt",
      "stderr": "stderr.txt",
      "input_files": {
        "geant4-dna.sh": "geant4-dna.sh",
        "input.tgz": "input.tgz"
      },
      "output_files": {
        "output.tgz": "output.tgz"
      }
    }
  }
],
  "requirements": { "software": "geant4-dna" }
}
```

Содержимое вспомогательного скрипта `geant4-dna.sh`:

```
#!/bin/bash
tar -xzf input.tgz
Microdosimetry input.mac
tar -czf output.tgz * --exclude input.*
```

Профили окружения индивидуально для каждого приложения можно настраивать с помощью пакета `softenv` [99].

Пример профиля для приложения GEANT4-DNA, установленного на кластере грид-сайта ОИЯИ инфраструктуры РГС имеет следующий вид (файл `/etc/softenv/db.d/geant4.db`):

```
(+geant4-4.9.4p01) {
  {desc: "GEANT4-DNA software version 4.9.4p01"}
  {flags: power}
  [] {
    G4ANALYSIS_USE 1
    source
'/shared/geant4.9.4p01/src/geant4/.config/bin/Linux-g+
+/env'
    source '/shared/OpenScientist/16.11.5/aida-
setup'
    PATH /shared/geant4.9.4p01/bin
    LD_LIBRARY_PATH
/shared/geant4.9.4p01/lib/Microdosimetry
  }
}
```

Соответствующая этому приложению часть в файле описания конфигурации грид-сайта:

```
<ng:Software>
  <ng:LocalID>geant4-4.9.4p01</ng:LocalID>
  <ng:Name>geant4</ng:Name>
  <ng:Version>4.9.4p01</ng:Version>
<ng:InstalledRoot>/shared/geant4.9.4p01/</ng:InstalledRoot>
  <ng:ModuleName>Microdosimetry</ng:ModuleName>
  <ng:EnvironmentSetup>
  <ng:ExtensionType>softenv</ng:ExtensionType>
  <ng:ExtensionValue>+geant4-
4.9.4p01</ng:ExtensionValue>
```

```

</ng:EnvironmentSetup>
<ng:ACL>
<ng:Rule>VOMS:/rngn.test</ng:Rule>
</ng:ACL>
</ng:Software>

```

Для запуска MPI-задач в среде РГС предусмотрен атрибут `count`, в качестве значения которого указывается целое число, равное необходимому для работы данного приложения количеству процессоров. Однако в силу того, что в ППО РГС не предусмотрено отдельное выполнение набора команд до и после вызова приложения, то указание в качестве исполняемого файла имени вспомогательного скрипта приводит к одновременному запуску N его копий, где N — количество процессоров, указанных в атрибуте `count`. Т. е. если скрипт выполняет распаковку архива, то эта операция будет запущена на N процессорах одновременно, что не является корректным поведением. Для избежания этого вспомогательный скрипт должен быть написан таким образом, чтобы команды предварительной подготовки и последующей обработки выполнялись одним процессором, а непосредственная работа параллельного приложения — на заданном пользователем их количестве.

Например, для параллельного приложения FDS, соответствующие файлы или их части выглядят, как представлено ниже.

Содержимое файла описания задания `fds.js`:

```

{ "version": 2,
  "description": "FDS",
  "default_storage_base":
"gsiftp://rngn.jinr.ru/shared/home/kut/fds/",
  "tasks": [ { "id": "a",
    "description": "task #1",
    "definition": { "version": 2,
      "executable": "/bin/sh",
      "arguments": ["fds.sh"],
      "stdout": "stdout.txt",
      "stderr": "stderr.txt",

```

```

        "input_files": {
            "fds.sh": "fds.sh",
            "input.tgz": "input.tgz"
        },
        "output_files": {
            "output.tgz": "output.tgz"
        },
        "count": 8
    }
}
],
    "requirements": { "software": "fds" }
}

```

Содержимое вспомогательного скрипта `fds.sh`:

```

#!/bin/sh
id_my=$OMPI_COMM_WORLD_RANK
id_max=$OMPI_COMM_WORLD_SIZE
if [ $id_my = 0 ]
then
    tar zxf input.tgz
    touch .mpi-arch.0
else
    while true
    do
        if [ -f .mpi-arch.0 ]
        then break
        else sleep 3
        fi
    done
fi

fds_mpi_intel_linux_64 *.fds

```

```

if [ $sid_my = 0 ]
then
  i=0
  while true
  do
    if [ $i = $sid_max ]
    then break
    fi
    if [ -f .mpi-arch.$i ]
    then
      i=`expr $i + 1`
    else
      sleep 3
    fi
  done
else
  touch .mpi-arch.$sid_my
fi

if [ $sid_my = 0 ]
then
  tar --exclude '*.sh' --exclude '*.tgz' --exclude
  '*.tar' --exclude '*.mpi-*' --exclude '*.tmp' --exclude
  'err.*' --exclude 'out.*' -czf output.tgz .
  sleep 5
fi

```

Профиль среды окружения для этого приложения и соответствующая ему часть в файле описания конфигурации грид-сайта подобна приведённым для ППП GEANT4-DNA.

Как было упомянуто ранее, помимо запуска заданий из командной строки, РГС предоставляет пользователям возможность взаимодействия с этой средой посредством графического веб-интерфейса. Для удобства работы пользователей с

конкретными прикладными пакетами разработчики предусмотрели возможность создания специализированных веб-интерфейсов в виде плагинов к ВИГ, что позволяет использовать его готовые модули по работе с сертификатами и входными/выходными файлами. Такие плагины принято называть проблемно-ориентированными веб-интерфейсами (ПОИ). Они должны быть написаны на языке JavaScript стандарта ECMAScript [100] в соответствии с документацией по прикладному программному интерфейсу (ППИ) ВИГ [101].

Отличие методов адаптации приложений в виде ПОИ заключается в следующем. На основании данных, заданных пользователем в проблемно-ориентированном веб-интерфейсе, этот ПОИ формирует вспомогательный скрипт, сохраняет его в отдельный файл на GridFTP сервере, а файл с описанием задания создаётся в виде объекта и при отправке задания на счёт соответствующей функцией передаётся необходимой компоненте.

Для данной среды с использованием описанных выше методов был адаптирован набор приложений, информация о которых представлена в таблице 7.

Таблица 7. Информация об адаптированных для РГС прикладных пакетах

Название пакета	Интерфейс запуска	Режим работы	Описание области применения пакета
DL_POLY	командная строка, ПОИ	параллельный	молекулярная динамика
Elmer	ПОИ	параллельный	инженерные расчёты
FDS	ПОИ	параллельный	моделирование задымлений и пожаров
GEANT4-DNA	командная строка, ПОИ	последовательный	моделирование биологических повреждений, вызванных ионизирующей радиацией на клеточном или субклеточном уровне
ZondGeoStat	ПОИ	последовательный	обработка геофизических,

			геохимических полей и данных дистанционных зондирований
--	--	--	---

Разработанные ПОИ (снимки экранов некоторых из них представлены на рис. 12 - 13) позволяют пользователю выполнять следующие действия:

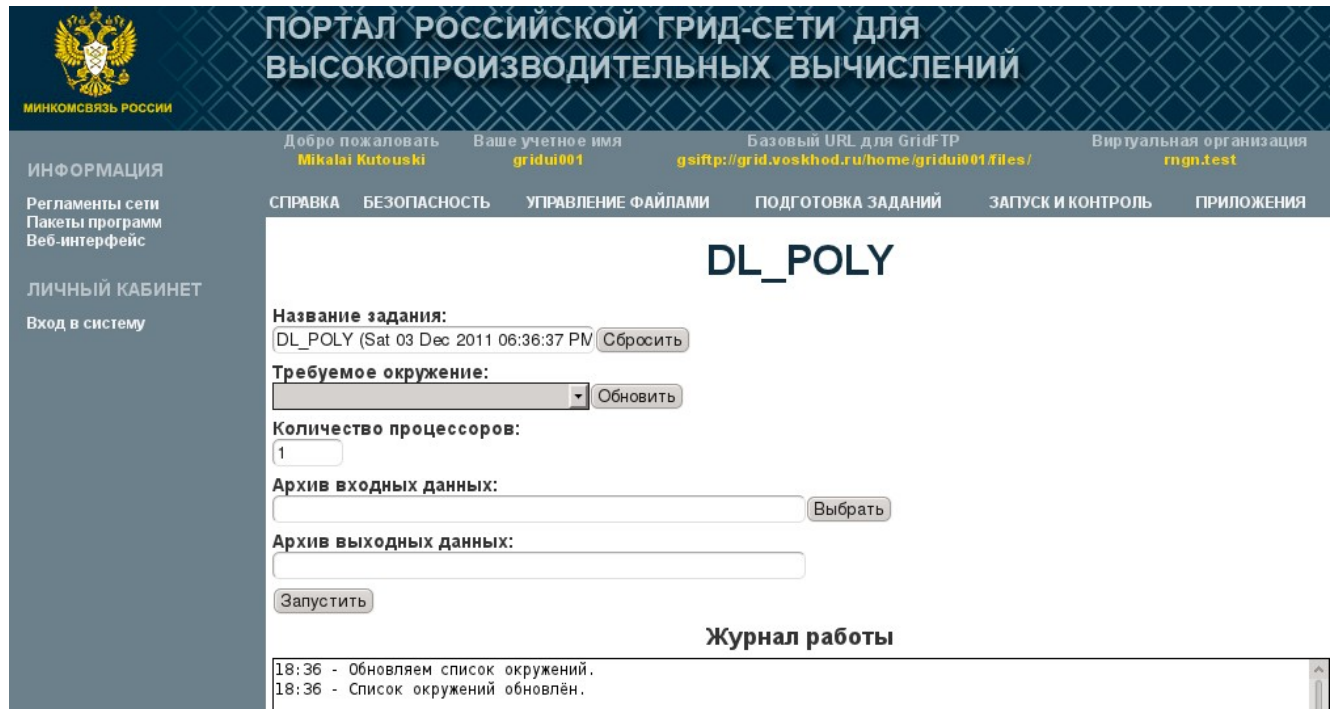


Рисунок 12. Снимок экрана ПОИ DL_POLY для РГС

- указать название задания (по умолчанию ПОИ автоматически генерирует в соответствующем поле название, состоящее из имени пакета, текущей даты и времени);
- выбрать требуемое окружение на вычислительных узлах;
- задать необходимое количество процессоров (только для DL_POLY, Elmer и FDS, способных работать в параллельном режиме) либо начальный размер сканирующего окна, шаг его изменения, количество окон и функцию статистики (для ППП ZondGeoStat);
- выбрать архив входных данных, которые должны быть подготовлены теми же средствами, что и при работе с пакетом вне грид-среды;
- указать имя файла для архива с выходными данными (если имя в поле

«Архив выходных данных» пользователем не задано, тогда будет использоваться имя output.tar.gz);

- запускать задания на счёт путём нажатия на кнопку ПОИ «Запустить» (обязательные поля должны быть заполнены);
- просматривать результаты операций в окне «Журнал работы».

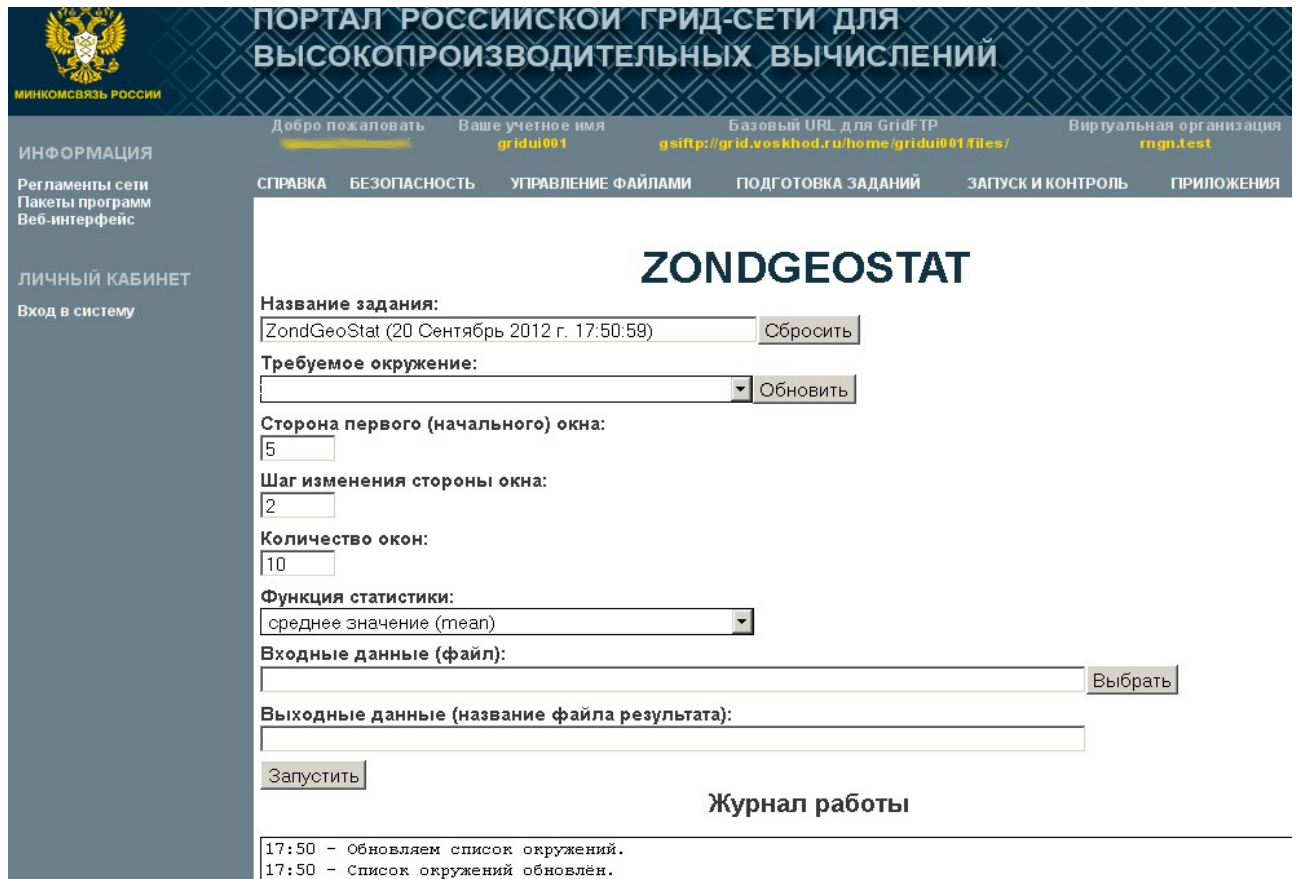


Рисунок 13. Снимок экрана ПОИ ZondGeoStat для PFC

В пункте «Запуск и контроль» основного меню ВИГ можно отслеживать статус отправленных заданий.

Т. к. ППП DL_POLY, Elmer и FDS имеют одинаковый набор входных параметров, то и ПОИ для этих пакетов тоже имеют одинаковый вид, подобно представленному на рис. 12, а также код модуля расширения ВИГ (в приложении А в качестве примера приведён код такого плагина для ППП FDS).

Пакет GEANT4-DNA может работать только в последовательном режиме. Поэтому его ПОИ не имеет поля «Количество процессоров», т. к. оно всегда равно

единице (см. рис. 14).

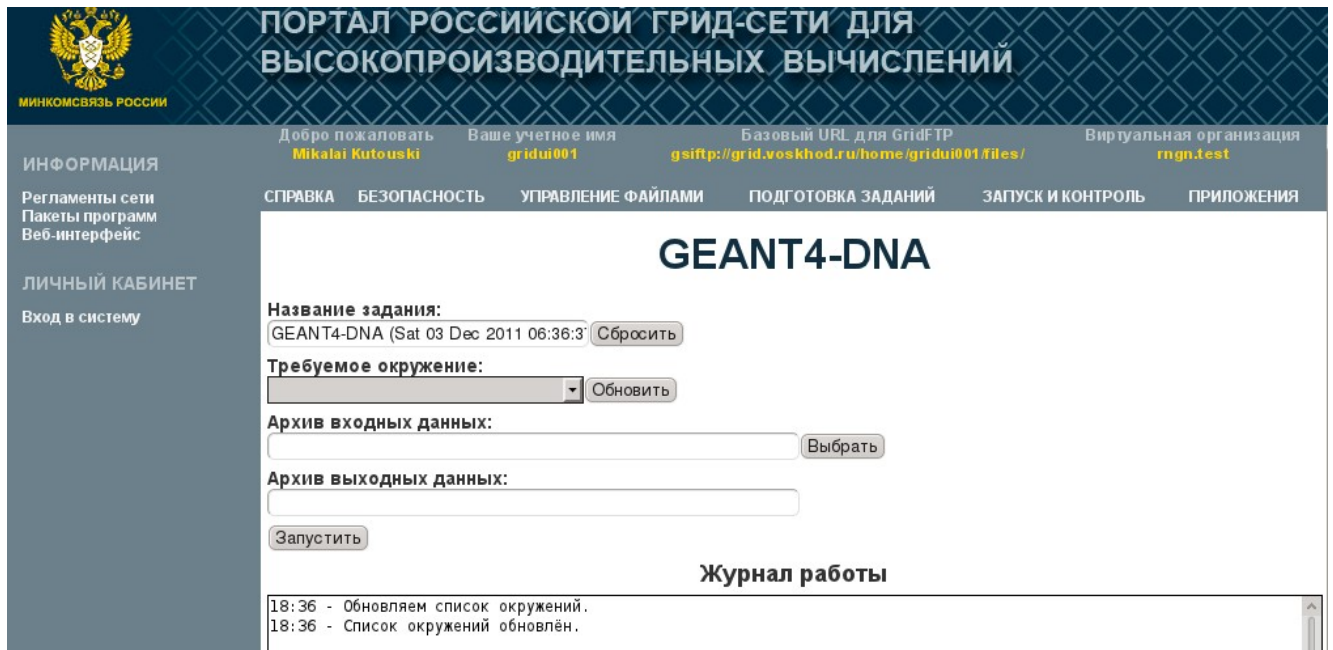


Рисунок 14. Снимок экрана ПОИ GEANT4-DNA для РГС

ПОИ ZondGeoStat (снимок его экрана представлен на рис. 10) несколько отличается от ПОИ предыдущих ППП, адаптированных для работы в РГС, в силу специфики самого пакета. Она заключается в том, что входные данные должны обрабатываться одним и тем же выбранным алгоритмом, но с меняющимся параметром (размером сканирующего окна), диапазон изменения которого задаётся пользователем в соответствующих полях ПОИ: «Сторона первого (начального) окна», «Шаг изменения стороны окна», «Количество окон». По этой причине ПОИ формирует задание, состоящее из определённого числа задач, соответствующего указанному в поле «Количество окон» значению. В отличие от параллельных приложений, эти задачи не взаимодействуют между собой во время вычислений, а каждая из них обрабатывает входные данные согласно заданным параметрам алгоритма и формирует свой набор файлов с результатами.

Программный код модуля расширения ВИГ РГС для ППП ZondGeoStat приведён в приложении Б.

Заключение

В результате выполнения диссертационной работы был проведён анализ задач в области ФВЭ, связанных с хранением, обработкой и анализом данных, и вариантов синтеза облачных и грид-технологий. Развита методика создания специализированных комплексов, позволяющие сократить время решения исследовательских задач и повысить эффективность использования компьютерных ресурсов за счёт размещения грид-сервисов на виртуальных машинах в облачной среде и применения виртуализации на уровне операционной системы.

С использованием развитых в данной работе методов в ЛИТ ОИЯИ создана облачная грид-система, которая применяется для решения широкого круга исследовательских и прикладных задач в области ФВЭ. Помимо этого, в ряде организаций из стран-участниц ОИЯИ: Центре суперкомпьютерных вычислений Национального технического университета Украины «Киевский политехнический институт», Физико-техническом институте низких температур им. Б.И. Веркина (Украина), Институте теоретической физики им. Н.Н. Боголюбова (Украина), Институте физики Национальной академии наук Азербайджана — с использованием этих же методов были построены аналогичные комплексы, со временем ставшие основой национальных грид-инфраструктур.

Разработана система мониторинга ресурсов уровня Tier-3, которая повысила эффективность использования ресурсов, позволив ускорить проведение анализа данных и получение физических результатов эксперимента ATLAS.

Разработаны методы адаптации определённого класса приложений для грид-сред на базе EMI и PGC, а также проведено их тестирование путём переноса ряда приложений в EMI (Blender, DL_POLY, Molpro, Elmer, MEEP, FDS) и PGC (DL_POLY, Elmer, FDS, GEANT4-DNA, ZondGeoStat).

Список терминов

3G-мост:	Generic Grid-Grid (3G) Bridge, программная компонента с открытым исходным кодом, позволяющая связывать различные грид-инфраструктуры.
БАК:	Большой адронный коллайдер.
ВО:	виртуальная организация, которая представляет собой группу людей или организаций, разделяющих между собой вычислительные, дисковые, информационные и сетевые ресурсы грида в общих целях.
ВЭ:	вычислительный элемент – сервис для использования вычислительных ресурсов грид-сайтов.
ГридННС:	проект для обеспечения географически распределенных научных и инженерных коллективов-участников российской национальной нанотехнологической сети (ННС) возможностью эффективного удаленного использования информационной, коммуникационной и вычислительной инфраструктуры.
ЕНУ:	Евразийский национальный университет имени Л.Н. Гумилева.
ИМИТ НАН:	Институт математики и информационных технологий Академии Наук Республики Узбекистан.
ИПП:	интерфейс программирования приложений (то же, что и API).
ИТФ:	Институт теоретической физики имени Н.Н. Боголюбова Национальной академии наук Украины.
ИФ НАН:	Институт физики Национальной академии наук Азербайджана.
КУ:	Каирский университет.
ЛИТ:	лаборатория информационных технологий ОИЯИ.
НИИРИМТ:	Национальный институт для исследования и развития изотопных и молекулярных технологий.
НИЦ:	научно-исследовательский центр.
ННУЦ ФЧВЭ	Национальный научно-учебный центр физики частиц и высоких

БГУ:	энергий Белорусского государственного университета.
НТУУ «КПИ»:	Национальный технический университет Украины «Киевский политехнический институт».
ОИЯИ:	объединённый институт ядерных исследований.
ОС:	операционная система.
ПИКС:	пользовательский интерфейс командной строки ГридННС.
ПО:	программное обеспечение.
ППО:	промежуточное программное обеспечение, т.е. ПО, выполняющее посреднические функции между приложениями пользователей и операционной системой грид-ресурсов.
ППП:	прикладной программный пакет.
РГС:	российская грид-сеть для высокопроизводительных вычислений, территориально-распределенная телекоммуникационная автоматизированная система общегосударственного уровня для обеспечения доступа к ресурсам российских суперкомпьютерных центров при организации распределенных высокопроизводительных вычислений.
РЦ:	ресурсный центр, т.е. оборудование, программное обеспечение, данные и вспомогательная инфраструктура, необходимые для работы одного или нескольких грид-сервисов в рамках одной организации.
УК:	университет Кейптауна.
УЦ:	полномочный орган, который выдаёт и управляет сертификатами и открытыми ключами.
ФВЭ:	физика высоких энергий
ФТИНТ:	Физико-технический институт низких температур им. Б.И. Веркина Национальной академии наук Украины.
ЦЕРН:	Европейская организация ядерных исследований.
ШИКТ МУНТ:	Школа информационных и коммуникационных технологий

- Монгольского университета науки и технологии.
- ШИТ НУМ: Школа информационных технологий Национального университета Монголии.
- ЭХД: элемент хранения данных, сервис для использования ресурсов хранения данных.
- APEL: Accounting Processor for Event Logs, сервис ППО ЕМІ для учёта потреблённых вычислительных ресурсов.
- API: application programming interface, набор готовых констант, структур и функций, используемых при программировании пользовательских приложений и обеспечивающих правильное взаимодействие между пользовательским приложением и операционной системой.
- ARC: Advanced Resource Connector, ППО грид, разрабатываемое коллаборацией NorduGrid (<http://www.nordugrid.org/arc>).
- ATLAS: A Toroidal LHC Apparatus, один из экспериментов на БАК в ЦЕРН.
- BDII: Berkeley Database Information Index, реализация на основе базы данных Berkley информационного индекса.
- BeinGRID: Business experiments in GRID, европейский проект по изучению возможности трансфера грид-технологий из науки в коммерческую сферу.
- BOINC: Berkeley Open Infrastructure for Network Computing, открытая программная платформа для организации распределённых вычислений на базе персональных компьютеров.
- CE: Computing Element, то же, что и ВЭ.
- Condor: Специализированная система организации высокоинтенсивных вычислений на большом количестве распределённых ресурсов.
- CREAM: Computing Resource Execution And Management, один из типов ВЭ.
- Dashboard: Система глобального мониторинга WLCG.

- dCache: проект, целью которого является создание программного комплекса для хранения и управления сверхбольшими массивами данных, распределённых между большим количеством гетерогенных узлов, в рамках единого дерева виртуальной файловой системы с широким набором стандартных методов доступа.
- DPM: Disk Pool Manager, один из типов ЭХД в ЕМІ.
- EDGI: European desktop grid initiative, европейский проект для разработки ППО, позволяющего использовать персональные компьютеры в качестве вычислительных ресурсов грид-среды.
- EGEE-II: преемник проекта EGEE.
- EGEE-III: преемник проекта EGEE-II.
- EGEE: проект по созданию глобальной грид-инфраструктуры, круглосуточно доступной учёным, независимо от их географического местоположения, для выполнения научных исследований в различных областях.
- EGI: European Grid Infrastructure, объединение поставщиков ресурсов для предоставления на постоянной основе комплексных и безопасных сервисов европейским исследователям и их международным партнёрам.
- EGI.eu: некоммерческий фонд для управления инфраструктурой EGI от лица её участников – национальных грид-инициатив и европейских межправительственных исследовательских организаций.
- EMІ: European middleware initiative, программная платформа для высокопроизводительных распределённых вычислений, создаваемая коллаборацией четырёх основных поставщиков европейского ППО: ARC, dCache, gLite и UNICORE.
- FTS: File Transfer Service, сервис передачи файлов, предназначенный

для распределения данных с БАК между центрами различных уровней.

- Ganglia:** масштабируемая распределенная система с иерархической структурой для мониторинга кластеров параллельных и распределенных вычислений и облачных систем.
- GENIUS:** веб-портал, позволяющий работать с инфраструктурой EGEE с помощью графического интерфейса.
- GILDA:** Grid INFN Laboratory for Dissemination Activities, виртуальная лаборатория для демонстрации возможностей грид-технологий и их популяризации.
- gLite:** ППО, созданное и развиваемое в рамках проекта EGEE, а затем как часть EMI, и используемое для создания глобальной грид-инфраструктуры.
- Globus Toolkit:** ППО с открытым исходным кодом для создания грид-систем и приложений.
- GOCDDB:** Grid Operations Center Database, база данных операционного центра производственной инфраструктуры WLCG, позволяющая вносить различного рода информацию о сервисах как самой инфраструктуры, так и грид-сайтах (количество вычислительных ресурсов и объём элементов хранения данных, сроки проведения профилактических работ, контакты системных администраторов грид-сайтов и т. п.), а также предоставлять её различным потребителям.
- GridFTP** высокопроизводительный, безопасный и надёжный протокол передачи данных, оптимизированный для использования в высокоскоростных сетях
- GridView:** инструмент для мониторинга и визуализации, предоставляющий такую информацию о различных аспектах работы инфраструктуры WLCG, как статистика передачи данных,

- запущенных задачах и доступности сервисов.
- GStat: система отображения информации о функционировании грид-сервисов, самой информационной системы инфраструктуры WLCG и соответствующих метриках.
- GT: Globus Toolkit.
- gUSE: grid and cloud User Support Environment, программная платформа с открытым исходным кодом для распределённых вычислений на базе облачных и грид-инфраструктур.
- Hyper-V: система аппаратной виртуализации для x64-систем на основе гипервизора компании Microsoft.
- IGE: initiative for globus in Europe, проект, предоставляющий европейским исследователям программные средства для совместного использования вычислительных мощностей, баз данных и других онлайн инструментов, основанный на ППО Globus Toolkit.
- IGTF: International Grid Trust Federation, объединение для выработки общих политик и правил между его участниками — органами управления политиками.
- JDL: Job Description Language, язык описания задач в среде gLite и EMI.
- JSON: JavaScript Object Notation, текстовый формат обмена данными, основанный на JavaScript и обычно используемый именно с этим языком.
- KVM: программное решение, обеспечивающее виртуализацию в среде Linux на платформе x86, которая поддерживает аппаратную виртуализацию на базе Intel Virtualization Technology либо AMD Secure Virtual Machine.
- LB: Logging&Bookkeeping Service, сервис ведения журналов работы.
- LFC: LCG file catalog, файловый каталог LCG.
- Lustre: параллельная распределённая файловая система.

- MammoGrid: европейский проект по созданию базы данных маммограмм, реализованный с использованием грид-технологий.
- MammoGrid+: преемник проекта MammoGrid.
- MTA SZTAKI: институт информатики и управления Венгерской академии наук.
- MyProxy: сервис для автоматического обновления прокси-сертификатов.
- NA3: одно из направлений проектов EGEE, связанное с обучением грид-технологиям.
- NeuGRID: проект по созданию набора медицинских грид-сервисов для изучения дегенеративных болезней мозга.
- NGI national grid initiative or infrastructure, национальная грид-инициатива или инфраструктура – созданная конкретной страной организация для управления ресурсами, предоставляемыми ею в EGI.
- OGE: Oracle Grid Engine, ранее известное как SGE (Sun Grid Engine), система с открытым исходным кодом для организации распределённых вычислений.
- OpenNebula: проект по разработке решения уровня промышленного стандарта для создания и управления дата-центров предприятий и локальных облачных инфраструктур.
- OpenStack: программная платформа, позиционируемая как облачная операционная система, способная управлять большим объёмом вычислительных и сетевых ресурсов, а также ресурсов хранения посредством удобного веб-интерфейса.
- OpenVZ: реализация технологии виртуализации на уровне операционной системы, которая базируется на ядре Linux.
- OSG: open science grid, объединение различного рода инфраструктур для удовлетворения потребностей исследовательского и академического сообществ.
- OSGEDU: виртуальная организация для проведения учебных практических занятий на грид-инфраструктуре проекта OSG.

- P-GRADE: портал для создания, выполнения и мониторинга вычислительных заданий в грид-средах посредством высокоуровневого графического веб-интерфейса.
- PanDA: Production and Distributed Analysis system, система для распределённого анализа, разработанная в рамках эксперимента ATLAS.
- PBS: Одна из систем управления локальными вычислительными ресурсами.
- PROOF: Parallel ROOT Facility, расширение программного пакета ROOT для проведения интерактивного параллельного анализа больших наборов ROOT-файлов на кластерах или многоядерных машинах.
- PSNC: Poznan Super Computing and Networking Center, Познаньский суперкомпьютерный и сетевой центр.
- QCG: QosCosGrid, система с возможностью расширенного управления задачами и ресурсами, предоставляющая конечным пользователям сравнимую с суперкомпьютерами производительность и архитектуру.
- rBDII: resource Berkeley Database Information Index, база данных Berkeley информационного индекса уровня ресурса.
- RDIG: Russian Data Intensive Grid, консорциум, целью которого является создание грид инфраструктуры для интенсивных операций с научными данными.
- ROC: Regional Operation Center, региональный операционный центр обеспечивает функционирование закреплённого за ним сегмента глобальной грид-инфраструктуры WLCG.
- SA3: направление в проекте EGEE, выполняющее функции по созданию качественного ППО gLite и соответствующих инструментов для конфигурации, по улучшению поддержки

- работы ППО на других платформах, улучшению интероперабельности с разными грид-инфраструктурами.
- SAGA:** Simple API for Grid Applications, проект по созданию стандартизированного интерфейса программирования приложений (ИПП) для разработки распределённого прикладного ПО с целью его запуска в облачных и грид-средах.
- SAM:** Service and availability monitoring, система тестирования состояния ресурсов WLCG.
- sBDII:** site Berkeley Database Information Index, база данных Berkeley информационного индекса уровня грид-сайта.
- SE:** Storage Element, то же, что и ЭХД.
- SGE:** см. OGE.
- SGM:** Software group manager, менеджер программной группы.
- StratusLab:** Проект по использованию виртуализации и облачных технологий в грид-инфраструктурах.
- tBDII:** top-level Berkeley Database Information Index, база данных Berkeley информационного индекса верхнего уровня.
- Torque:** Terascale Open-Source Resource and QUEue Manager, менеджер распределённых ресурсов для вычислительных кластеров под управлением ОС Linux и других Unix-подобных операционных систем, одна из современных версий PBS.
- UI:** User Interface, интерфейс пользователя, предоставляющий пользователям грид-среды возможность взаимодействия с ней путём выполнения команд.
- UMD:** unified middleware distribution, унифицированный дистрибутив ППО, предоставляющий исследователям единую платформу из стабильных программных компонент общего назначения для грид-вычислений.
- UNICORE:** ППО, позволяющее создавать готовые к работе распределённые грид-инфраструктуры.

- VOBOX:** Virtual Organisation box, грид-сервис, предоставляющий привилегированным пользователям конкретной ВО возможность выполнять на грид-сайте такие действия, как установка дополнительного ПО, запуск и остановка специфических для ВО сервисов и т.д.
- VOMS:** Virtual organizations membership service, сервис управления виртуальной организацией.
- WISDOM-II:** преемник проекта WISDOM.
- WISDOM:** проект для исследования посадочных леганд в поиске лекарства от малярии.
- WLCG:** проект по созданию и поддержанию работоспособности распределённой инфраструктуры для анализа и хранения данных с БАК.
- WMS:** Workload Management System, система управления загрузкой.
- WN:** Worker Node, рабочий узел кластера, на котором непосредственно происходит счёт задачи.
- WS-PGRADE:** веб-порталом системы gUSE который позволяет выполнять разработку и запуск распределённых приложений, исполняемых на вычислительных ресурсах различных распределённых инфраструктур, включая кластеры, производственные грид-инфраструктуры.
- Xen:** кроссплатформенный гипервизор, разработанный в компьютерной лаборатории Кембриджского университета и распространяемый на условиях лицензии GPL.
- XRootD:** Масштабируемая высокопроизводительная система для организации надёжного хранения данных и быстрого доступа к ним.
- YAIM:** Yet another installation method, программный инструмент для настройки грид-сервисов.

Список литературы

1. Foster I., Kesselman C., Tuecke S. Anatomy of the Grid: Enabling Scalable Virtual Organizations // International Journal of High Performance Computing Applications. ISSN: 1094-3420. 2001. №15 (3). P. 200 - 222.
2. I. Foster, C. Kesselman The Grid: Blueprint for a New Computing Infrastructure, Morgan-Kaufman. 1999. ISBN 978-1-5586-0475-9.
3. Веб-портал проекта WLCG. 2013 [Электронный ресурс]. URL: <http://wlcg.web.cern.ch/> (дата обращения: 04.04.2013).
4. Веб-страница с информацией о проектах MammoGrid и MammoGrid+. 2010 [Электронный ресурс]. URL: <http://ehealth.kactoo.com/mammogrid.htm> (дата обращения: 05.04.2013).
5. Веб-портал проекта WISDOM. 2006 [Электронный ресурс]. URL: <http://wisdom.healthgrid.org/> (дата обращения: 05.04.2013).
6. Веб-портал проекта neuGRID. 2012 [Электронный ресурс]. URL: <http://www.neugrid.eu> (дата обращения: 05.04.2013).
7. Портал HealthGrid. 2010 [Электронный ресурс]. URL: <http://healthgrid.org> (дата обращения: 05.04.2013).
8. Antje Wolf, Mohammad Shahid, Vinod Kasam, Wolfgang Ziegler and Martin Hofmann-Apitius In Silico Drug Discovery Approaches on Grid Computing Infrastructures // Current Clinical Pharmacology. ISSN: 1574-8847. 2010. №5 (1). P. 37-46.
9. Веб-страница проекта BEinGRID на портале IT-tude.com. 2012 [Электронный ресурс]. URL: <http://www.it-tude.com/projects/beingrid> (дата обращения: 03.04.2013).
10. The future of innovation: developing Europe's ICT infrastructures // Проект "GridTalk". 2009 [Электронный ресурс]. Систем требования: Adobe Acrobat Reader.

URL: <http://www.gridtalk.org/briefings/einfrastructures.pdf> (дата обращения: 03.06.2013).

11. Н.А. Кутовский Облачные автономные грид-инфраструктуры для учебно-исследовательских и тестовых целей // Информатизация образования и науки. ISSN: 2073-7572. 2013. №4 (20). С. 15-29.

12. S. Belov et al VM-based infrastructure for simulating different cluster and storage solutions used on ATLAS Tier-3 sites // Journal of Physics: Conference Series. 2012. Vol. 396. Part 4. 5 pp. doi:10.1088/1742-6596/396/4/042036

13. Н.А. Кутовский, В.В. Кореньков Инфраструктура обучения grid-технологиям // Открытые системы. СУБД. ISSN: 1028-7493. 2009. №10 (139). С. 48-51.

14. Кореньков В.В., Кутовский Н.А., Семенов Р.Н. Опыт адаптации прикладных программных пакетов для работы в грид-средах // Компьютерные исследования и моделирование. ISSN: 2076-7633. 2012. Т. 4. № 2. С. 339-344.

15. Веб-страница с информацией о направлении SA3 проекта EGEE-II. 2007 [Электронный ресурс]. URL: <http://egee-sa3.web.cern.ch/egee-sa3/> (дата обращения: 03.05.2013).

16. Веб-страница с информацией о направлении NA3 проекта EGEE-III. 2008 [Электронный ресурс]. URL: <http://www.egee-rdig.ru/tasks/na3.php> (дата обращения: 03.05.2013).

17. Веб-портал проекта GENIUS. 2010 [Электронный ресурс]. URL: <http://egee.cesnet.cz/en/user/genius.html> (дата обращения: 03.05.2013).

18. Веб-портал проекта GILDA. 2011 [Электронный ресурс]. URL: <http://gilda.ct.infn.it> (дата обращения: 03.05.2013).

19. Веб-портал проекта EGI. 2013 [Электронный ресурс]. URL: <http://www.egi.eu> (дата обращения: 06.05.2013).

20. Веб-страница с информацией о проекте EGI. 2012 [Электронный ресурс].

URL: <http://www.egi.eu/about> (дата обращения: 04.05.2013).

21. Веб-страница с информацией о внешних партнёрах EGI.eu. 2013 [Электронный ресурс]. URL: <https://www.egi.eu/community/collaborations> (дата обращения: 04.05.2013).

22. Веб-страница с информацией о поставщиках технологий для EGI. 2013 [Электронный ресурс]. URL: <https://www.egi.eu/infrastructure/technology/technology-providers/index.html> (дата обращения: 04.05.2013).

23. Веб-портал проекта EMI. 2013 [Электронный ресурс]. URL: <http://www.eu-emi.eu> (дата обращения: 04.05.2013).

24. Веб-страница с информацией об EMI на портале EGI. 2013 [Электронный ресурс]. URL: <https://www.egi.eu/community/collaborations/EMI.html> (дата обращения: 04.05.2013).

25. Веб-страница ППО ARC. 2013 [Электронный ресурс]. URL: <http://www.nordugrid.org/arc> (дата обращения: 05.05.2013).

26. Веб-страница проекта dCache. 2012 [Электронный ресурс]. URL: <https://www.dcache.org> (дата обращения: 05.05.2013).

27. Веб-страница проекта gLite. 2010 [Электронный ресурс]. URL: <http://glite.cern.ch> (дата обращения: 05.05.2013).

28. Веб-страница проекта UNICORE. 2012 [Электронный ресурс]. URL: <http://www.unicore.eu> (дата обращения: 05.05.2013).

29. Веб-страница с информацией об UMD. 2013 [Электронный ресурс]. URL: http://repository.egi.eu/category/umd_releases/ (дата обращения: 04.05.2013).

30. Веб-страница с информацией о IGE на портале EGI. 2012 [Электронный ресурс]. URL: <https://www.egi.eu/community/collaborations/IGE.html> (дата обращения: 04.05.2013).

31. Веб-портал проекта IGE. 2013 [Электронный ресурс]. URL: <http://www.ige-project.eu> (дата обращения: 04.05.2013).
32. Веб-страница инструментария Globus Toolkit. 2013 [Электронный ресурс]. URL: <http://globus.org/toolkit> (дата обращения: 04.05.2013).
33. Веб-портал Познаньского суперкомпьютерного и сетевого центра. 2013 [Электронный ресурс]. URL: <http://www.man.poznan.pl/online/en/> (дата обращения: 04.05.2013).
34. Веб-страница с информацией о поставщике технологий PSNC на портале EGI. 2012 [Электронный ресурс]. URL: <https://www.egi.eu/community/collaborations/PSNC.html> (дата обращения: 04.05.2013).
35. Веб-страница проекта SAGA. 2013 [Электронный ресурс]. URL: <http://saga-project.github.io> (дата обращения: 04.05.2013).
36. Веб-страница с информацией о поставщике технологий SAGA на портале EGI. 2012 [Электронный ресурс]. URL: <https://www.egi.eu/community/collaborations/SAGA.html> (дата обращения: 04.05.2013).
37. Веб-портал проекта StratusLab. 2013 [Электронный ресурс]. URL: <http://stratuslab.eu> (дата обращения: 07.05.2013).
38. Веб-страница портала EGI с информацией . 2012 [Электронный ресурс]. URL: <http://www.egi.eu/community/collaborations/StratusLab.html> (дата обращения: 07.05.2013).
39. Веб-страница с информацией о сервисе EGI Training Marketplace. 2012 [Электронный ресурс]. URL: http://www.egi.eu/services/support/training_marketplace/index.html (дата обращения: 06.05.2013).
40. Веб-страница школы GridKa School. 2012 [Электронный ресурс]. URL:

http://wiki.scc.kit.edu/gridkaschool/index.php/Main_Page (дата обращения: 05.05.2013).

41. Вики-страница с информацией о командах поддержки пользователей национальных грид-инициатив. 2013 [Электронный ресурс]. URL: https://wiki.egi.eu/wiki/TNA3.3_NGI_User_Support_Teams (дата обращения: 05.05.2013).

42. Веб-портал проекта Open Science Grid. 2013 [Электронный ресурс]. URL: <http://www.opensciencegrid.org> (дата обращения: 05.05.2013).

43. Вики-страница на портале OSG с информацией о функциях команды по интеграции ППО. 2012 [Электронный ресурс]. URL: <https://www.opensciencegrid.org/bin/view/Integration/WebHome> (дата обращения: 05.05.2013).

44. Раздел с часто задаваемыми вопросами на портале OSG. 2013 [Электронный ресурс]. URL: <https://www.opensciencegrid.org/bin/view/Documentation/OsgFaq> (дата обращения: 05.05.2013).

45. Вики-страница с информацией о целях образовательного направления в рамках проекта OSG. [Электронный ресурс]. URL: <https://twiki.grid.iu.edu/bin/view/Education/GridWorkshops> (дата обращения: 06.05.2013).

46. Вики-страница с информацией о виртуальной организации OSGEDU. 2009 [Электронный ресурс]. URL: <https://twiki.grid.iu.edu/bin/view/VirtualOrganizations/VOInfo/OSGEDU> (дата обращения: 05.05.2013).

47. Веб-портал WS-PGRADE. 2013 [Электронный ресурс]. URL: <http://guse.sztaki.hu> (дата обращения: 06.05.2013).

48. Веб-портал gUSE. 2012 [Электронный ресурс]. URL: <http://www.guse.hu> (дата обращения: 06.05.2013).

49. Веб-портал консорциума RDIG. 2009 [Электронный ресурс]. URL: <http://egeerdig.ru> (дата обращения: 06.05.2013).
50. Веб-страница с информацией о целях и задачах Программы "СКИФ-ГРИД". 2007 [Электронный ресурс]. URL: http://skif-grid.botik.ru/index.php?option=com_content&task=view&id=22&Itemid=35 (дата обращения: 06.05.2013).
51. Веб-страница с информацией о целях проекта "СКИФ-Полигон". 2008 [Электронный ресурс]. URL: http://skif-grid.botik.ru/index.php?option=com_content&task=view&id=98&Itemid=91 (дата обращения: 06.05.2013).
52. Совместный пресс-релиз Суперкомпьютерной платформы «СКИФ», ИПС им. А.К. Айламазяна РАН, Южно-Уральского государственного Университета, компании «Alt-Linux», «ТЕСИС», «РСК СКИФ» и «СКИРУС» об инновационной инициативе "СКИФ-ГРИД" образованию и науке" // . 2010 [Электронный ресурс]. Систем требования: . URL: <http://skif-grid.botik.ru/images/stories/presentation/pressrelease/2010-10-27-press-release.pdf> (дата обращения: 06.05.2013).
53. Интервью С.М. Абрамова блогу Intel Galaxy. 2009 [Электронный ресурс]. URL: <https://ru.intel.com/business/community/?automodule=blog&blogid=7605&showentry=1040> (дата обращения: 06.05.2013).
54. Веб-портал проекта ГридННС. 2012 [Электронный ресурс]. URL: <http://ngrid.ru> (дата обращения: 06.05.2013).
55. Веб-страница с информацией о проекте Российской грид-сети. 2012 [Электронный ресурс]. URL: <http://www.voskhod.ru/index.php?id=9&item=25> (дата обращения: 06.05.2013).
56. Веб-страница с информацией о гипервизоре Hyper-V. 2013 [Электронный ресурс]. URL: <http://microsoft.com/hyper-v> (дата обращения: 05.07.2013).
57. Веб-портал программного продукта Windows Azure. 2013 [Электронный ресурс]. URL: <http://microsoft.com/hyper-v> (дата обращения: 05.07.2013).

58. Веб-портал проекта Xen. 2013 [Электронный ресурс]. URL: <http://www.xenproject.org> (дата обращения: 05.07.2013).
59. Веб-портал проекта KVM. 2013 [Электронный ресурс]. URL: <http://www.linux-kvm.org> (дата обращения: 05.07.2013).
60. Веб-портал облачной платформы OpenNebula. 2013 [Электронный ресурс]. URL: <http://opennebula.org> (дата обращения: 07.06.2013).
61. Веб-портал проекта OpenStack. 2013 [Электронный ресурс]. URL: <http://www.openstack.org> (дата обращения: 05.07.2013).
62. Веб-портал компании C12G Labs. 2013 [Электронный ресурс]. URL: <http://c12g.com/company/> (дата обращения:).
63. Веб-портал компании Rackspace. 2013 [Электронный ресурс]. URL: <http://www.rackspace.com> (дата обращения: 05.07.2013).
64. Веб-портал с информацией об учебных курсах по Windows Azure. 2013 [Электронный ресурс]. URL: <http://www.microsoftvirtualacademy.com/topics/product-windows-azure-en> (дата обращения: 05.07.2013).
65. Веб-страница с информацией об учебных курсах по облачному решению Amazon - AWS. 2013 [Электронный ресурс]. URL: <http://aws.amazon.com/training> (дата обращения: 05.07.2013).
66. Веб-страница с информацией об учебных курсах по продукту OpenStack. 2013 [Электронный ресурс]. URL: <http://www.rackspace.com/cloud/private/training> (дата обращения: 05.07.2013).
67. Веб-портал программного продукта OpenVZ. [Электронный ресурс]. URL: <http://openvz.org> (дата обращения: 03.05.2013).
68. Веб-страница с информацией о драйвере OpenVZ для облачной платформы OpenNebula 2.2.1. 2013 [Электронный ресурс]. URL:

<http://wiki.opennebula.org/openvz4opennebula> (дата обращения: 02.08.2013).

69. Веб-страница с информацией о текущей версии драйвера OpenVZ для облачной платформы OpenNebula. 2013 [Электронный ресурс]. URL:

https://bitbucket.org/hpcc_kpi/opennebula-openvz (дата обращения: 02.08.2013).

70. Kutovskiy N.A. Distributed training and testing grid infrastructure evolution // The 5th International Conference «Distributed Computing and Grid-technologies in Science and Education» (Dubna, 16-21 July 3, 2012): proceedings. Dubna, 2012. P. 180-185.

71. V.V. Korenkov, N.A. Kutovskiy Distributed training and testing grid infrastructure // The 4th International Conference «Distributed Computing and Grid-technologies in Science and Education» (Dubna, June 28 - July 3, 2010): proceedings. Dubna, 2010. P. 148-152.

72. Belov S.D., Korenkov V.V., Kutovskiy N.A. Educational Grid infrastructure at JINR // The 3rd International Conference «Distributed Computing and Grid-Technologies in Science and Education» (Dubna, 30 June - 4 July, 2008): proceedings. Dubna, 2008. P. 341-342.

73. Веб-портал международного сообщества International Grid Trust Federation. 2013 [Электронный ресурс]. URL: <http://www.igtf.net> (дата обращения: 20.05.2013).

74. Веб-страница с информацией о программной компоненте EDGI executor. 2013 [Электронный ресурс]. URL: <http://doc.desktopgrid.hu/doku.php?id=manual:glitemce> (дата обращения: 07.08.2013).

75. J. Andreeva et al. Tier-3 Monitoring Software Suite (ТЗМОН) proposal // ATLAS note. 2011.

76. Веб-страница проекта Job Monarch. 2013 [Электронный ресурс]. URL: <http://sourceforge.net/projects/jobmonarch> (дата обращения: 28.06.2013).

77. Desktop grids: Connecting everyone to science. 2011 [Электронный ресурс]. . URL: <http://www.e-sciencetalk.org/briefings/EST-Briefing-19-DesktopGrid-w.pdf> (дата обращения: 10.09.2013).

78. Статья в Википедии о проектах добровольных вычислений. 2013 [Электронный ресурс]. URL: http://en.wikipedia.org/wiki/List_of_distributed_computing_projects (дата обращения: 04.03.2013).
79. Posypkin, Mikhail and Semenov, Alexander and Zaikin, Oleg Using BOINC desktop grid to solve large scale SAT problems // Computer Science. 1508-2806. 2012. №13 (1). 25-34.
80. О.С. Заикин, М.А. Посыпкин, А.А. Семёнов, Н.П. Храпов Опыт организации добровольных вычислений на примере проектов OPTIMA@HOME и SAT@HOME // Вестник Нижегородского университета им. Н.И. Лобачевского. 1993 - 1778. 2012. №5 (2). 340-347.
81. Веб-страница с информацией о компоненте 3G-мост. 2013 [Электронный ресурс]. URL: <http://doc.desktopgrid.hu/doku.php?id=component:3gbridge> (дата обращения: 05.07.2013).
82. Веб-портал системы мониторинга Dashboard инфраструктуры WLCG. 2013 [Электронный ресурс]. URL: <http://dashboard.cern.ch> (дата обращения: 28.06.2013).
83. Веб-портал Брукхейвенской национальной лаборатории. 2013 [Электронный ресурс]. URL: <http://www.bnl.gov> (дата обращения: 28.06.2013).
84. Веб-портал университета Дьюка. 2013 [Электронный ресурс]. URL: <http://duke.edu> (дата обращения: 28.06.2013).
85. Веб-портал DESY. 2013 [Электронный ресурс]. URL: <http://www.desy.de> (дата обращения: 28.06.2013).
86. P Nilsson et al The ATLAS PanDA Pilot in Operation // Journal of Physics: Conference Series. 2011. Vol. 331. Part 6. 8 pp. doi:10.1088/1742-6596/331/6/062040
87. P Nilsson et al Recent Improvements in the ATLAS PanDA Pilot // Journal of Physics: Conference Series. 2012. Vol. 396. Part 3. 8 pp. doi:10.1088/1742-6596/396/3/032080

88. Веб-страница с информацией по работе в среде ГридННС через Пользовательский интерфейс командной строки. 2011 [Электронный ресурс]. URL: <http://ngrid.ru/ngrid/support/user/ucli> (дата обращения: 18.06.2012).
89. Веб-страница с информацией по работе в ГридННС посредством ВИГ. 2013 [Электронный ресурс]. URL: http://ngrid.ru/ngrid/support/user/quick_start#веб-интерфейс__гридннс_виг (дата обращения: 18.06.2012).
90. JDL attributes specification. 2005 [Электронный ресурс]. URL: <https://edms.cern.ch/document/590869/1> (дата обращения: 24.06.2012).
91. Веб-портал прикладного пакета Blender. 2013 [Электронный ресурс]. URL: <http://www.blender.org> (дата обращения: 25.06.2013).
92. Вики-страница пакета mpi-start. 2013 [Электронный ресурс]. URL: <https://grid.ifca.es/wiki/Middleware/MpiStart> (дата обращения: 24.06.2013).
93. Веб-страница приложения DL_POLY. 2013 [Электронный ресурс]. URL: <http://www.stfc.ac.uk/cse/25526.aspx> (дата обращения: 21.06.2013).
94. Веб-страница приложения Elmer. 2012 [Электронный ресурс]. URL: <http://www.csc.fi/english/pages/elmer> (дата обращения: 21.06.2013).
95. Веб-страница пакета Molpro. 2012 [Электронный ресурс]. URL: <http://www.molpro.net> (дата обращения: 21.06.2013).
96. Веб-страница пакета MEER. 2012 [Электронный ресурс]. URL: <http://ab-initio.mit.edu/wiki/index.php/Meer> (дата обращения: 21.06.2013).
97. Веб-страница приложения FDS. 2013 [Электронный ресурс]. URL: <http://code.google.com/p/fds-smv> (дата обращения: 21.06.2013).
98. Веб-страница с информацией по формату JSON. 2013 [Электронный ресурс]. URL: <http://json.org/json-ru.html> (дата обращения: 25.06.2013).
99. Веб-страница пакета softenv. 2013 [Электронный ресурс]. URL: <http://www.mcs.anl.gov/hs/software/systems/softenv/softenv-admin.html> (дата

обращения: 27.06.2013).

100. Веб-страница языка ECMAScript. 2013 [Электронный ресурс]. URL: <http://www.ecmascript.org> (дата обращения: 27.06.2013).

101. Документация по ППИ ВИГ. 2011 [Электронный ресурс]. Системные требования: Adobe Acrobat Reader. URL: http://ngrid.ru/ngrid/_media/poi/vig-api-plugin.pdf (дата обращения: 27.06.2013).

Список иллюстративного материала

Рисунок 1. Схемы возможных вариантов синтеза облачных и грид-технологий....	11
Рисунок 2. Схема взаимосвязей между организациями и проектами, являющимися участниками EGI.....	18
Рисунок 3. Схема структуры комплекса для исследовательских задач.....	33
Рисунок 4. Общая схема облачной грид-системы ЛИТ ОИЯИ для исследовательских задач.....	35
Рисунок 5. Схема распределённого полигона на базе ППО ЕМІ.....	40
Рисунок 6. Схема компонент полигона ТЗМОН.....	43
Рисунок 7. Схема компонент полигона РГС.....	45
Рисунок 8. Схема компонент полигона DesktopGrid.....	45
Рисунок 9. Снимки экрана веб-портала облачной грид-системы.....	46
Рисунок 10. Снимки экрана веб-интерфейса Ganglia с информацией о кластерах полигона ТЗМОН.....	56
Рисунок 11. Снимок экрана начальной страницы веб-интерфейса ПХД РГС.....	57
Рисунок 12. Снимок экрана ПОИ DL_POLY для РГС.....	74
Рисунок 13. Снимок экрана ПОИ ZondGeoStat для РГС.....	75
Рисунок 14. Снимок экрана ПОИ GEANT4-DNA для РГС.....	76
Таблица 1. Соответствие задач возможным вариантам синтеза облачных и грид-технологий.....	11
Таблица 2. Различия между производственными и исследовательскими грид-системами.....	38
Таблица 3. Список названий организаций и размещённых в них грид-сервисов распределённого полигона на базе ППО ЕМІ.....	40
Таблица 4. Компоненты полигона ТЗМОН.....	44
Таблица 5. Список проведённых курсов по обучению пользователей работе в средах gLite и ЕМІ.....	50
Таблица 6. Список проведённых курсов по обучению системных администраторов	

.....52

Таблица 7. Информация об адаптированных для РГС прикладных пакетах.....73

Приложение А. Программный код модуля расширения ВИГ РГС для ППП FDS

```
// Плагин к ВИГ для ППП FDS

// Тело плагина - безымянная функция
;(function(){

    // Под этим именем плагин будет виден в списке
    var name = "FDS";

    // Эта функция добавляет плагин в меню и возвращает DOM-
    узел типа DIV
    // Строить интерфейс плагина нужно именно в этом узле
    var dom = register_plugin(name);

    // Объект невидимого редактора заданий (DAG)
    var dag = new uw_dagedit();

    // Создаём нужные нам DOM-объекты
    // Журнал операций
    var log = document.createElement("textarea");
    log.disabled = true;
    log.style.width = "100%";
    log.style.overflow = "scroll";
    log.style.border = "1px solid";
    log.style.color = "black";
    log.style.backgroundColor = "transparent";
    log.rows = 10;
    // Функция для добавления записи в журнал
    var logwrite = function(text) {
        var date = new Date;
        var time = date.getHours() + ":" + (date.getMinutes())
```

```

< 10 ? "0" : "") + date.getMinutes();
    log.value += time + " - " + text + "\n";
}

// Поле ввода для описания задачи
var descr = document.createElement("input");
descr.style.width = "30%";

// Кнопка восстановления значения по-умолчанию
var descrdef = document.createElement("button");
descrdef.innerHTML = "Сбросить";
descrdef.onclick = function() {
    descr.value = "FDS (" + (new Date).toLocaleString() +
") "
}
// Устанавливаем изначальное значение
descrdef.onclick();

// Выпадающий список
var softenv = document.createElement("select");
softenv.style.width = "30%";
softenv.onchange = function() {
    // Запоминаем выбранное пользователем значение
    set_cookie("fds-softenv", softenv.value);
}
// Функция обратного вызова для заполнения списка окружений
var envlist_cb = function(ret) {
    // Список окружений
    var envlist = [];
    // Если вернулось число, то это код системной
ошибки HTTP
    if(typeof ret == "number") {
        logwrite("Ошибка " + ret + " получения списка

```

```

окружений.");
        return;
    } else {
        // Если status не 0, то возникла ошибка на уровне ПАИГ
        if(ret["status"] != 0) {
            logwrite("Ошибка обновления списка
окружений: " + ret["retMsg"] + ".");
            return;
        }
        try {
            var reply = JSON.parse(ret["reply"],
function (key, value) {
                var d;
                if (key === "softenv") {
                    d = value;
                    if(/^\+?fds/.test(d) && ! envlist[d])
envlist[d] = 1;

                    return d;
                }
            });

        } catch(e) {
            logwrite("Неверный отклик информационной
системы.");

            return;
        }
    }
    // Очищаем выпадающий список
while(softenv.length > 0) softenv.remove(0);
    // Читаем последнее запомненное значение
var oldval = get_cookie("fds-softenv");
    // Заполняем выпадающий список полученными
значениями

```

```

for(var i in envlist) {
    var opt = document.createElement("option");
    opt.text = i;
    opt.value = i;
    // Если совпадает с запомненным, то выбираем
    этот пункт по умолчанию.
    if(i == oldval) opt.selected = true;
    try {
        softenv.add(opt, null);
    } catch(e) {
        softenv.add(opt);
    }
}
logwrite("Список окружений обновлён.");
}

```

```

// Кнопка обновления списка окружений
var softdef = document.createElement("button");
softdef.innerHTML = "Обновить";
softdef.onclick = function() {
    // Массив запроса к ПАИГ
    var req = [];
    logwrite("Обновляем список окружений.");
    // Запрос к информационной системе
    // ada - глобальная переменная
    ui_req_a(envlist_cb, ada, "infoQuery", req);
}

```

```

// Загрузка списка
softdef.onclick();

```

```

// Входной файл
var np = document.createElement("input");

```



```

np.style.width = "50px";
np.value = "3";

function is_int(value){
    for (i = 0 ; i < value.length ; i++) {
        if ((value.charAt(i) < '0') || (value.charAt(i)
> '9')) return false
    }
    return true;
}

// Функция обратного вызова для запуска задания
var run_cb = function(ret) {
    // Если вернулось число, то это код системной ошибки
HTTP
    if(typeof ret == "number") {
        logwrite("При попытке запуска задания возникла
системная ошибка " + ret + ". Обратитесь к администратору
ВИГ.");
        // Иначе - массив с ответом от ПАИГ
    } else {
        if(ret["status"] == 0) {
            logwrite("Задание " + ret["jobName"] + " успешно
запущено.");
        } else {
            logwrite("Запуск задания " + ret["jobName"] + "
завершился с ошибкой " + ret["status"] + ": " +
ret["retMsg"] + ".");
        }
    }
}

// Входной файл

```

```

var infile = document.createElement("input");
infile.style.width = "50%";

// Кнопка выбора входного файла
var inselect = document.createElement("button");
inselect.innerHTML = "Выбрать";
inselect.onclick = function() {
// Открываем диалог выбора файла
    // Корень дерева каталогов
    file_select({ init_path: "files/",
// Начальное значение пути относительно корня
        path: infile.value.substring(0,
infile.value.lastIndexOf("/")),
// Функция обратного вызова при выборе файла
// Аргумент - путь к выбранному файлу относительно корня
        cb: function(file) {infile.value = file}
    });
}

// Выходной файл
var outfile = document.createElement("input");
outfile.style.width = "50%";
outfile.value = "output.tgz";

// Адрес сервера GridFTP
var gridftp = document.createElement("input");
gridftp.style.width = "50%";
gridftp.onchange = function() {
// Если указан внешний сервер GridFTP, то выбирать
// локальные файлы нет смысла
    inselect.disabled = this.value != "";
}

```

```

// Кнопка запуска задания
var start = document.createElement("button");
start.innerHTML = "Запустить";
// Функция запуска задания
start.onclick = function() {
  if (is_int(np.value) == false) { alert ("Введено
неправильное количество процессоров!"); exit; }
  // Сброс редактора в пустое состояние
  dag.clear();

  // Добавление задачи
  var task = dag.addnode();
  // Комментарий к задаче
  task.title = descr.value;
  // Задача будет иметь встроенное описание (не в файле)
  task.task = {};
  // Установка ожидаемого сервисом Pilot значения
  task.task.version = 2;
  // Количество процессоров
  task.task.count = parseInt(np.value);

  var nproc = softenv.value;
  var nprocexec = nproc.substr(1,nproc.length-1);
  var nprocver = nprocexec.split("-");
  var nprocreq = nprocver[0] + " == " + nprocver[1];

var runscript = "";
runscript = runscript + "#!/bin/sh\n";
runscript = runscript + "\n";
runscript = runscript + "id_my=$OMPI_COMM_WORLD_RANK\n";
runscript = runscript + "id_max=$OMPI_COMM_WORLD_SIZE\n";
runscript = runscript + "#echo $id_my >> /tmp/0000.txt\n";
runscript = runscript + "if [ $id_my = 0 ]\n";

```

```

runscript = runscript + "    then \n";
runscript = runscript + "        tar zxf input.tgz\n";
runscript = runscript + "        touch .mpi-arch.0\n";
runscript = runscript + "    else\n";
runscript = runscript + "        while true\n";
runscript = runscript + "        do\n";
runscript = runscript + "            if [ -f .mpi-arch.0 ]\n";
runscript = runscript + "                then break\n";
runscript = runscript + "                else sleep 3\n";
runscript = runscript + "            fi\n";
runscript = runscript + "        done\n";
runscript = runscript + "fi\n";
runscript = runscript + "fds_mpi_intel_linux_64 *.fds\n";
runscript = runscript + "if [ $id_my = 0 ]\n";
runscript = runscript + "    then\n";
runscript = runscript + "        i=0\n";
runscript = runscript + "        while true\n";
runscript = runscript + "        do\n";
runscript = runscript + "            if [ $i = $id_max ] \n";
runscript = runscript + "                then break\n";
runscript = runscript + "            fi\n";
runscript = runscript + "            if [ -f .mpi-arch.$i ]\n";
runscript = runscript + "                then \n";
runscript = runscript + "                    i=`expr $i +
1` \n";
runscript = runscript + "                else \n";
runscript = runscript + "                    sleep 3\n";
runscript = runscript + "                fi\n";
runscript = runscript + "            done\n";
runscript = runscript + "        else\n";
runscript = runscript + "            touch .mpi-arch.$id_my\n";
runscript = runscript + "fi\n";
runscript = runscript + "\n";

```

```

runscript = runscript + "if [ $id_my = 0 ]\n";
runscript = runscript + "    then tar --exclude '*.sh'
--exclude '*.tgz' --exclude '*.tar' --exclude '*.mpi-*'
--exclude '*.tmp' --exclude '*.fds' -cf output.tar . \n";
runscript = runscript + "    gzip output.tar \n";
runscript = runscript + "    mv output.tar.gz output.tgz
\n";
runscript = runscript + "        sleep 5\n";
runscript = runscript + "fi\n";
    // Сохраняем шелл-скрипт на сервере для последующего
доступа с ресурсов по gridftp.
        ui_file_save_s("files/fds_mpi_run.sh", runscript);

    task.task.executable = "/bin/sh";
    task.task.arguments = ["fds_mpi_run.sh"];
    // Требуемое окружение
    task.task.extensions = {"softenv" : softenv.value};
    // Входные файлы
    task.task.input_files = {"fds_mpi_run.sh" :
"fds_mpi_run.sh", "input.tgz" : infile.value};
    // Выходные файлы
    task.task.output_files = {"output.tgz" :
outfile.value};
    task.task.stderr = "stderr_fds.txt";
    task.task.stdout = "stdout_fds.txt";
    // Если указан внешний сервер GridFTP, то использовать его
        if(gridftp.value) task.task.default_storage_base =
gridftp.value;

        task.task.requirements = {"software": nprocreq};

    // Массив запроса к ПАИГ
    var req = [];

```

```

// Описание задания
req["definition"] = dag.text();
// Имя задания в списке
req["jobName"] = descr.value;
logwrite("Запускаем задание " + descr.value + ".");

// Запрос на запуск задания
// run_cb - наша функция обратного вызова
// ada - глобальная переменная
ui_req_a(run_cb, ada, "jobRun", req);
}

// Прицепляем всё к отведённому для плагина DIV.
dom.innerHTML = "<h1 style='align:center;font-size:24pt'>FDS</h1>";
dom.appendChild(document.createElement("p"));
dom.appendChild(document.createElement("h3"));
dom.lastChild.innerHTML = "Название задания:";
dom.appendChild(descr);
dom.appendChild(descrdef);
dom.appendChild(document.createElement("p"));
dom.appendChild(document.createElement("h3"));
dom.lastChild.innerHTML = "Требуемое окружение:";
dom.appendChild(softenv);
dom.appendChild(softdef);
dom.appendChild(document.createElement("p"));
dom.appendChild(document.createElement("h3"));
dom.lastChild.innerHTML = "Количество процессоров:";
dom.appendChild(np);
dom.appendChild(document.createElement("p"));
dom.appendChild(document.createElement("h3"));
dom.lastChild.innerHTML = "Архив входных данных:";
dom.appendChild(infile);

```

```
dom.appendChild(inselect);
dom.appendChild(document.createElement("p"));
dom.appendChild(document.createElement("h3"));
dom.lastChild.innerHTML = "Архив выходных данных:";
dom.appendChild(outfile);
dom.appendChild(document.createElement("p"));
dom.appendChild(start);
dom.appendChild(document.createElement("p"));
dom.appendChild(document.createElement("h2"));
dom.lastChild.innerHTML = "Журнал работы";
dom.lastChild.style.textAlign = "center";
dom.appendChild(document.createElement("p"));
dom.appendChild(log);

// Конец плагина.
}());
```

Приложение Б. Программный код модуля расширения ВИГ РГС для ППП ZondGeoStat

```
// Плагин к ВИГ для ППП ZondGeoStat

// Тело плагина - безымянная функция
;(function(){

// Имя плагина в меню ВИГ
    var name = "ZondGeoStat";

// Добавление плагина в меню и получение DOM-узла типа DIV
    var dom = register_plugin(name);

// Объект невидимого редактора заданий (DAG)
    var dag = new uw_dagedit();

// Создание нужных DOM-объектов

    // Журнал операций
    var log = document.createElement("textarea");
    log.disabled = true;
    log.style.width = "100%";
    log.style.overflow = "scroll";
    log.style.border = "1px solid";
    log.style.color = "black";
    log.style.backgroundColor = "transparent";
    log.rows = 10;
// debug
// log.rows = 30;

    // Функция для добавления записи в журнал
    var logwrite = function(text) {
```



```

var date = new Date;
var time = date.getHours() + ":" + (date.getMinutes()
< 10 ? "0" : "") + date.getMinutes();
log.value += time + " - " + text + "\n";
}

// Поле ввода для описания задачи
var descr = document.createElement("input");
descr.style.width = "30%";

// Кнопка восстановления значения по-умолчанию
var descrdef = document.createElement("button");
descrdef.innerHTML = "Сбросить";
descrdef.onclick = function() {
    descr.value = "ZondGeoStat (" + (new
Date).toLocaleString() + ")"
}

// Установка изначального значения
descrdef.onclick();
// Выпадающий список
var softenv = document.createElement("select");
softenv.style.width = "30%";
softenv.onchange = function() {
    // Сохранение выбранного пользователем значения
    set_cookie("zondgeostat-softenv", softenv.value);
}

// Функция обратного вызова для заполнения списка
окружений
var envlist_cb = function(ret) {
    // Список окружений
    var envlist = [];

```

```
// Если вернулось число, то это код системной
// ошибки HTTP
if(typeof ret == "number") {
    logwrite("Ошибка " + ret + " получения списка
окружений.");
    return;
} else {
    // Если status не 0, то возникла ошибка
    // на уровне ПАИГ.
    if(ret["status"] != 0) {
        // ПАИГ вернёт сообщение об ошибке в поле
        // retMsg
        logwrite("Ошибка обновления списка
окружений: " + ret["retMsg"] + ".");
        return;
    }
    try {
        var reply = JSON.parse(ret["reply"],
function (key, value) {
            var d;
            if (key === "softenv") {
                d = value;
                if(/^\\+?zondgeostat/.test(d) && !
envlist[d]) envlist[d] = 1;
                return d;
            }
        });
    } catch(e) {
        logwrite("Неверный отклик информационной
системы.");
        return;
    }
}
```

```

}
// Очистка выпадающего списка
while(softenv.length > 0) softenv.remove(0);
// Чтение последнего запомненного значения
var oldval = get_cookie("zondgeostat-softenv");
// Заполнение выпадающего списка полученными
// значениями
for(var i in envlist) {
    var opt = document.createElement("option");
    opt.text = i;
    opt.value = i;
    // Если совпадает с запомненным, то выбираем
    // этот пункт по умолчанию
    if(i == oldval) opt.selected = true;
    try {
        softenv.add(opt, null);
    } catch(e) {
        softenv.add(opt);
    }
}
logwrite("Список окружений обновлён.");
}

// Кнопка обновления списка окружений
var softdef = document.createElement("button");
softdef.innerHTML = "Обновить";
softdef.onclick = function() {
    // Массив запроса к ПАИГ
    var req = [];
    logwrite("Обновляем список окружений.");
    // Запрос к информационной системе
    // ada - глобальная переменная
    ui_req_a(envlist_cb, ada, "infoQuery", req);
}

```

```
}

// Загрузка списка
softdef.onclick();

// Сторона
var n_start = document.createElement("input");
n_start.style.width = "50px";
n_start.value = "5";

// Шаг
var n_step = document.createElement("input");
n_step.style.width = "50px";
n_step.value = "2";

// Конец
var n_end = document.createElement("input");
n_end.style.width = "50px";
n_end.value = "10";

// Алгоритм
var alg = document.createElement("select");
alg.style.width = "350px";

var choice = document.createElement("option");
choice.value = "mean";
choice.appendChild(document.createTextNode("среднее
значение (mean)"));
alg.appendChild(choice);

choice = document.createElement("option");
choice.value = "minimum";
choice.appendChild(document.createTextNode("минимум
```

```
значений (minimum)"));
    alg.appendChild(choice);

    choice = document.createElement("option");
    choice.value = "maximum";
    choice.appendChild(document.createTextNode("максимум
значений (maximum)"));
    alg.appendChild(choice);

    choice = document.createElement("option");
    choice.value = "median";
    choice.appendChild(document.createTextNode("медиана
(median)"));
    alg.appendChild(choice);

    choice = document.createElement("option");
    choice.value = "sum";
    choice.appendChild(document.createTextNode("сумма
значений (sum)"));
    alg.appendChild(choice);

    choice = document.createElement("option");
    choice.value = "range";
    choice.appendChild(document.createTextNode("диапазон
значений (range)"));
    alg.appendChild(choice);

    choice = document.createElement("option");
    choice.value = "stddev";
    choice.appendChild(document.createTextNode("стандартное
отклонение (stddev)"));
    alg.appendChild(choice);
```

```

choice = document.createElement("option");
choice.value = "majority";
choice.appendChild(document.createTextNode("наиболее
часто встречающееся значение (majority)"));
alg.appendChild(choice);

choice = document.createElement("option");
choice.value = "minority";
choice.appendChild(document.createTextNode("наименее
часто встречающееся значение (minority)"));
alg.appendChild(choice);

choice = document.createElement("option");
choice.value = "variety";
choice.appendChild(document.createTextNode("число
уникальных значений (variety)"));
alg.appendChild(choice);

function is_int(value){
    for (i = 0 ; i < value.length ; i++) {
        if ((value.charAt(i) < '0') || (value.charAt(i)
> '9')) return false
    }
    return true;
}

// Функция обратного вызова для запуска задания
var run_cb = function(ret) {
// Если вернулось число, то это код системной ошибки
// HTTP
if(typeof ret == "number") {
    logwrite("При попытке запуска задания возникла

```

```
системная ошибка " + ret + ". Обратитесь к администратору
ВИГ.");
```

```
    } else { // Иначе - массив с ответом от ПАИГ.
        if(ret["status"] == 0) {
            logwrite("Задание " + ret["jobName"] + " успешно
запущено.");
        } else {
            logwrite("Запуск задания " + ret["jobName"] + "
завершился с ошибкой " + ret["status"] + ": " +
ret["retMsg"] + ".");
        }
    }
}
```

```
// Входной файл
var infile = document.createElement("input");
infile.style.width = "50%";

var inselect = document.createElement("button"); //
Кнопка выбора входного файла.
inselect.innerHTML = "Выбрать";
inselect.onclick = function() {
    // Открываем диалог выбора файла.
    // Корень дерева каталогов, выше которого не подняться
    file_select({ init_path: "files/",
// Начальное значение пути относительно корня
                path: infile.value.substring(0,
infile.value.lastIndexOf("/")),
// Функция обратного вызова при выборе файла
// Аргумент - путь к выбранному файлу относительно
// корня
                cb: function(file) {infile.value = file}
    });
};
```

```
}

// Выходной файл
var outfile = document.createElement("input");
outfile.style.width = "50%";

// Адрес сервера GridFTP
var gridftp = document.createElement("input");
gridftp.style.width = "50%";
gridftp.onchange = function() {
// Если указан внешний сервер GridFTP, то выбирать
// локальные файлы нет смысла
inselect.disabled = this.value != "";
}

// Кнопка запуска задания
var start = document.createElement("button");
start.innerHTML = "Запустить";
// Функция запуска задания
start.onclick = function() {
    logwrite("Запуск пошел... ");

    var nproc = softenv.value;
    var nprocexec = nproc.substr(1,nproc.length-1);
    var nprocver = nprocexec.split("-");
    var nprocreq = nprocver[0] + " == " + nprocver[1];

// Сброс редактора в пустое состояние
dag.clear();

runshell = "";
runshell = runshell + "#!/bin/bash\n";
runshell = runshell + "gdal_as -f config.xml\n";
```



```

ui_file_save_s("files/run_zgs.sh", runshell);

var f_end = (Number(n_end.value)-1)*Number(n_step.value);

for (var i = Number(n_start.value); i <=
Number(n_start.value)+f_end; i = i + Number(n_step.value))
{
    // Сохраняем файлы
    // Добавляем задачу
    var task = dag.addnode();
    task.title = descr.value + " zondgeostat (" + i +
    ")";
    task.task = {}; // Задача будет иметь встроенное
    описание (не в файле).
    task.task.version = 2;

var runscript = "";
runscript = runscript + "<?xml version='1.0' encoding='UTF-
8' ?>\n";
runscript = runscript + "<task>\n";
runscript = runscript + " <source>\n";
runscript = runscript + " <grid2d>\n";
runscript = runscript + "
<connection>input.asc</connection> \n";
runscript = runscript + " <data>\n";
runscript = runscript + " <band>1</band>\n";
runscript = runscript + " </data>\n";
runscript = runscript + " </grid2d>\n";
runscript = runscript + " </source>\n";

```

```

runscript = runscript + "<target>\n";
runscript = runscript + "  <grid2d>\n";
runscript = runscript + "
<connection>rez.tif</connection>\n";
runscript = runscript + "      <data>\n";
runscript = runscript + "          <name>output-band-
name</name>\n";
runscript = runscript + "          <type>double</type>\n";
runscript = runscript + "      </data>\n";
runscript = runscript + "      <format>gtiff</format>\n";
runscript = runscript + "      <no-data>\n";
runscript = runscript + "          <value>-100</value>\n";
runscript = runscript + "      </no-data>\n";
runscript = runscript + "  </grid2d>\n";
runscript = runscript + "</target>\n";
runscript = runscript + " <processing>\n";
runscript = runscript + "  <step exec='procname'>\n";
runscript = runscript + "      <function>"+ alg.value
+"</function>\n";
runscript = runscript + "      <parameters>\n";
runscript = runscript + "          <neighbourhood>\n";
runscript = runscript + "              <no-data>\n";
runscript = runscript + "
<policy>ignore</policy>\n";
runscript = runscript + "          </no-data>\n";
runscript = runscript + "          <units>cell</units>\n";
runscript = runscript + "          <rectangle>\n";
runscript = runscript + "              <width>" + i +
"</width>\n";
runscript = runscript + "              <height>" + i +
"</height>\n";
runscript = runscript + "          </rectangle>\n";
runscript = runscript + "      </neighbourhood>\n";

```

```

runscript = runscript + "    </parameters>\n";
runscript = runscript + "    </step> \n";
runscript = runscript + " </processing>\n";
runscript = runscript + "</task>\n";

    // Сохраняем скрипт на сервере для последующего
доступа с ресурсов по gridftp
        ui_file_save_s("files/zondgeostat" + i + ".xml",
runscript);

    var xml_file = "zondgeostat" + i + ".xml";

    task.task.executable = "/bin/bash";
    task.task.arguments = ["run_zgs.sh"];
    task.task.extensions = {"softenv" : softenv.value}; //
Требуемое окружение.
        task.task.input_files = {"config.xml" : xml_file ,
"input.asc" : infile.value, "run_zgs.sh" :
"run_zgs.sh"}; // Входные файлы.
        task.task.output_files = {"rez.tif" : outfile.value+ i
+ ".tif"}; // Выходные файлы.
        task.task.stderr = "stderr_zgs" + i + ".txt";
        task.task.stdout = "stdout_zgs" + i + ".txt";
        task.task.requirements = {"software": nprocreq};
        logwrite ("Записали конфиг..." + i);

}

    // Массив запроса к ПАИГ
    var req = [];
    // Описание задания
    req["definition"] = dag.text();
    // Имя задания в списке

```

```

req["jobName"] = descr.value;
logwrite("Запускаем задание " + descr.value + ".");
// Запрос на запуск задания
// run_cb - наша функция обратного вызова
// ada - глобальная переменная
ui_req_a(run_cb, ada, "jobRun", req);
}

// Прицепляем всё к отведённому для плагина DIV.
dom.innerHTML = "<h1 style='align:center;font-size:24pt'>ZondGeoStat</h1>";
dom.appendChild(document.createElement("p"));
dom.appendChild(document.createElement("h3"));
dom.lastChild.innerHTML = "Название задания:";
dom.appendChild(descr);
dom.appendChild(descrdef);
dom.appendChild(document.createElement("p"));
dom.appendChild(document.createElement("h3"));
dom.lastChild.innerHTML = "Требуемое окружение:";
dom.appendChild(softenv);
dom.appendChild(softdef);

dom.appendChild(document.createElement("p"));
dom.appendChild(document.createElement("h3"));
dom.lastChild.innerHTML = "Сторона первого (начального)
окна:";
dom.appendChild(n_start);

dom.appendChild(document.createElement("p"));
dom.appendChild(document.createElement("h3"));
dom.lastChild.innerHTML = "Шаг изменения стороны
окна:";
dom.appendChild(n_step);

```

```

dom.appendChild(document.createElement("p"));
dom.appendChild(document.createElement("h3"));
dom.lastChild.innerHTML = "Количество окон:";
dom.appendChild(n_end);

```

```

dom.appendChild(document.createElement("p"));
dom.appendChild(document.createElement("h3"));
dom.lastChild.innerHTML = "Функция статистики:";
dom.appendChild(alg);

```

```

dom.appendChild(document.createElement("p"));
dom.appendChild(document.createElement("h3"));
dom.lastChild.innerHTML = "Входные данные (файл):";
dom.appendChild(infile);
dom.appendChild(inselect);
dom.appendChild(document.createElement("p"));
dom.appendChild(document.createElement("h3"));
dom.lastChild.innerHTML = "Выходные данные (название
файла результата)";
dom.appendChild(outfile);
dom.appendChild(document.createElement("p"));
dom.appendChild(start);
dom.appendChild(document.createElement("p"));
dom.appendChild(document.createElement("h2"));
dom.lastChild.innerHTML = "Журнал работы";
dom.lastChild.style.textAlign = "center";
dom.appendChild(document.createElement("p"));
dom.appendChild(log);

```

```

// Конец плагина.
}());

```