

РАЗРАБОТКА ПЕРВОГО НАЦИОНАЛЬНОГО СТАНДАРТА ДЛЯ ОБЕСПЕЧЕНИЯ ИНТЕРОПЕРАБЕЛЬНОСТИ В ГРИД-СРЕДЕ

Е. Е. Журавлёв¹, В. Н. Корниенко², А. Я. Олейников, Т. Д. Широбокова²

¹Физический Институт им. П.Н. Лебедева РАН

²Институт радиотехники и электроники им. В.А.Котельникова РАН

Получена 21 февраля 2011 г.

Аннотация. Описана структура и изложено основное содержание проекта первого национального стандарта для обеспечения интероперабельности в Грид-среде.

Ключевые слова: Грид-среда, интероперабельность, стандарт, язык JSDL.

Abstract. The paper have deal with Structure and Content of first national Standard for providing Interoperablity Grid – environment.

Keywords: Grid-environment, Interoperability, Standard, JSDL.

Введение

В настоящее время активно разворачиваются работы по развитию Грид-технологий, созданию и применению Грид-систем. Вычислительные и информационные ресурсы Грид-среды реализуются на разнородных программно-аппаратных платформах, поэтому среда является гетерогенной и поэтому в ней возникает проблема интероперабельности разнородных платформ а также удаленных пользователей со средой. Проблема интероперабельности решается за счет применения принципов открытых систем, т.е. использования согласованных наборов стандартов, называемых профилями. За рубежом вопросами разработки стандартов для Грид-среды занимается целый ряд организаций, основной из которых выступает международная организация OpenGridForum (OGF) [1]. В работе OGF участвуют 50 стран, которые представляют коммерческие, правительственные

и научные организации. К настоящему времени разработано более 170 документов различного статуса.

Хотя в нашей стране активно разворачиваются работы по Грид-системам, но до настоящего времени не разработано не только ни одного профиля, но даже нет ни одного национального стандарта. Без национальных стандартов невозможно строить национальный сегмент мировой Грид-системы, поскольку согласно Федерального Закона «О техническом регулировании» [2] на территории РФ должны применяться национальные стандарты, гармонизированные с международными.

Проведя анализ имеющихся международных стандартов в области Грид-технологий, мы предложили в качестве первого национального стандарта разработать стандарт на язык описания представления задач, взяв за основу разработанный в 2008 г в рамках OpenGridForum язык описания задач JSDL (Job Submission Description Language). Этот документ называется GFD.136 Rec. Он основан на языке XML, содержит в себе описание элементов (и их допустимых значений), при помощи которых можно формировать как собственно задание, так и требования на программно-аппаратную платформу, на которой это задание может быть выполнено. JSDL хорошо зарекомендовал себя в мировой практике [3].

Таким образом, нами было принято решение о разработке на базе документа GFD.136 Rec. национального стандарта [4]. Авторы имеют опыт такой работы. Так, нами разработан документ ГОСТ Р 51954-2002 «Профиль прикладной среды организации вычислений на супер-ЭВМ».

Данная работа выполнена в рамках Программы Президиума РАН №1 и при поддержке РФФИ (грант № 09-07-00171)

Структура и содержание предлагаемого стандарта

Документ получил название «Язык описания представления задач. Версия 1.0» и оформлен, согласно требованиям ГОСТ 1.5-2001, в виде проекта национального стандарта. Содержание документа приведено в Приложении.

Документ содержит 80 стр. текста и включает Введение, 11 разделов, три обязательных приложения и одно информационное.

Во Введении к документу сказано, что язык описания представления задач, который для краткости будем обозначать JSDL, представляет собой язык для описания требований вычислительных задач к ресурсам, особенно в Грид-среде, хотя и не ограничивается ими. Нам представляется, что этот язык найдёт применение при проведении облачных вычислений [5], которые находят всё большее распространение. Язык JSDL содержит словарь и нормативную XML-схему, которые упрощают описание этих требований, используя набор элементов XML.

Версия JSDL 1.0 даёт базовый словарь описания задачи для ее представления в Грид-среде. Этот базовый словарь включает понятия, используемые в ряде существующих систем: Condor, Globus Toolkit, Load Sharing Facility (LSF), Portable Batch System (PBS), (Sun) GridEngine (SGE), Uniform Interface to Computing Resources (Unicore).

Предполагается, что реализация более сложных функций будет обеспечена за счет не стандартизованного расширения версии JSDL 1.0. Часть таких функций могут стать нормативными стандартными расширениями языка в установленном порядке.

Элементы JSDL подразделяются на три следующие общие категории:

- требования на идентификацию задачи;
- требования на ресурсы;
- требования на данные.

Элементы языка JSDL версии 1.0 ограничены описанием требований задач на момент представления. Не существует элементов, определенных в спецификации JSDL версии 1.0 таким образом, чтобы они содержали информацию о задаче после того, как она была представлена. Такая информация, как уникальные идентификаторы задач или информация о статусе задачи, как правило, поддерживается основными системами управления задачами. Метод доступа к такого сорта информации может быть описан в

отдельных документах, либо добавлен в спецификацию JSDL версии 1.0 в качестве расширений.

Разделы 1, 2, 3 Область применения, Нормативные ссылки, Термины, определения и сокращения являются каноническими для любого национального стандарта.

Раздел 4 «Основные положения» содержит 2 подраздела: 4.1 «Спецификация JSDL» и 4.2 «4.2 «Соглашения, принятые в данном документе»

В подразделе 4.1 определены два основных сценария, исходя из которых разработана спецификация языка.

Согласно первому сценарию, многие организации, разрабатывающие Грид-системы, устанавливают различные системы управления задачами. При этом каждая система может иметь свой собственный язык для описания требований представления задач. Это делает взаимодействие между этими системами управления задачами сложным. Для того чтобы использовать такие различные системы управления задачами, организации вынуждены подготовить и поддерживать целый ряд различных документов, касающихся представления задач, по одному для каждой системы, каждый из которых описывает одно и то же представление. Стандартизованный язык, такой как JSDL, может быть легко перенесен на различные системы и, тем самым, может полностью решить эту проблему.

Согласно второму сценарию, Грид-среда, как заведомо гетерогенная, вовлекает во взаимодействие целый ряд систем управления заданиями различных типов. Рис.1 иллюстрирует один из возможных видов такой Грид-среды. Описание представления задачи (показано в виде документа JSDL на рис.1) в Грид-среде может быть преобразовано посредниками или дополнительно уточнено при помощи информации, первоначально отсутствовавшей у заявителя задачи. Это описание может быть передано между системами или реализовано на одном из ресурсов, удовлетворяющем требованиям данной задачи. Все эти взаимодействия могут осуществляться

автоматически, при помощи стандартного языка, который может быть легко транслирован на собственный язык каждой из систем.

На рис.1 в качестве примера показана часть систем, в которых язык JSDL может быть использован в Грид-среде.

Отметим, что использование JSDL не ограничивается только этими системами. Может существовать множество других потребителей, к которым могут относиться: системы учета, системы защиты, системы архивации, системы аудита и пр.

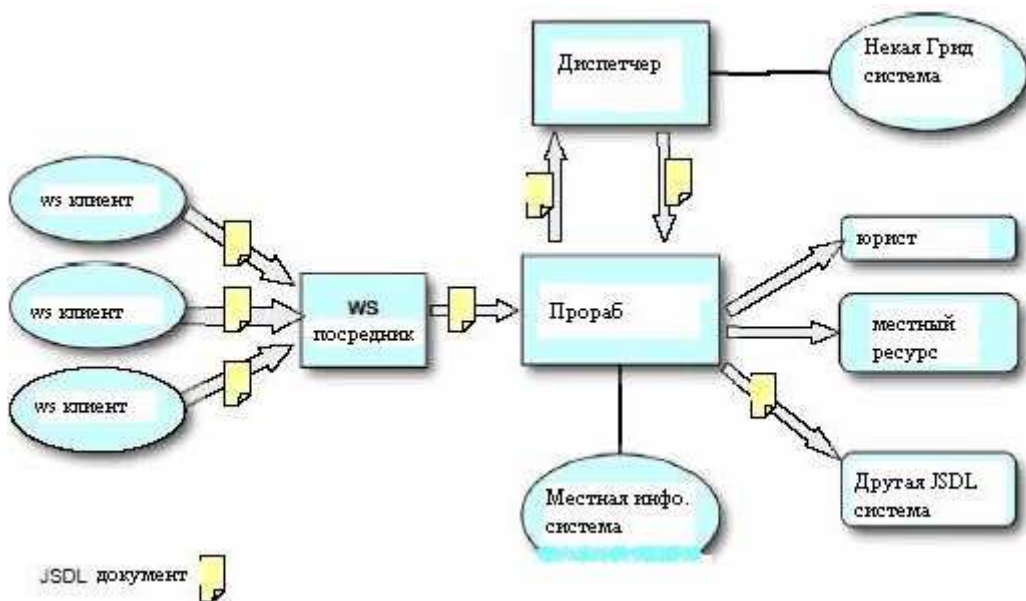


Рис.1. Роль JSDL в Грид-среде.

В п.4.2 «Соглашения, принятые в данном документе» даётся интерпретация ключевых слов, используемых в документе: «обязан», «недопустимо», «требуемый», «должен», «не позволяется», «рекомендуется», «может» и «опциональный» а также приводятся префиксы и пространства имен, используемые в данной спецификации. Так, термины «элемент JSDL» и «атрибут JSDL» указывают на то, что соответствующая языковая конструкция представлена в нормативной схеме JSDL либо как элемент языка XML, либо как атрибут языка XML.

Термин «документ JSDL» означает правильно отформатированный документ XML, который может быть проверен на соответствие нормативному определению схемы JSDL, которое содержится в Приложении А.

Ключевое слово «present», использованное для ссылки на элемент JSDL, подразумевает, что экземпляр элемента содержится в документе JSDL.

Ключевое слово «support» по отношению к потребительской системе, использующей JSDL и поддерживающей спецификацию JSDL и языковые конструкции, относится к возможности этой системы проводить синтаксический анализ документа JSDL. Т.е. потребительская система должна иметь возможность интерпретировать языковые конструкции и присваивать им семантические конструкции, описанные в этой спецификации, а также присваивать им значения, содержащиеся в документе JSDL. Таким образом, все потребляющие системы, совместимые с JSDL, должны поддерживать все языковые конструкции JSDL.

Ключевое слово «satisfy» по отношению к потребительской системе, удовлетворяющей документу JSDL, означает, что система может как поддерживать, так и удовлетворять все элементы, представленные в документе JSDL. Элемент считается удовлетворенным, если потребительская система может обеспечить надлежащее выполнение значения, присвоенного элементу в документе JSDL.

В Разделе 5 «Область действия JSDL» даётся обоснование области применимости JSDL 1.0.

JSDL – язык, используемый для описания требований представления индивидуальных задач. Он не служит для описания полного жизненного цикла задачи или взаимодействия между отдельными задачами. Таким образом, JSDL допускает, что процессы управления и исполнения задач, описанных и представленных с использованием JSDL, в частности в Грид-среде, могут требовать ряд других языков и протоколов. Например, для описания взаимодействия отдельных задач, которые представлены при помощи JSDL, и для описания взаимодействия этих задач с потребляемыми или производимыми

ими данными, следует использовать язык рабочего потока. В качестве другого примера рассматривается процесс переговоров, который необходим при размещении задачи в оптимальной среде ресурсов в соответствии с ее требованиями. В этом случае требуется использование удовлетворительного процесса переговоров и согласованных протоколов. Примером такого протокола является протокол, который определен в спецификации GGF WS-соглашения (WS-AG). На рис.2 приведены некоторые другие элементы, которые необходимы для представления, управления и выполнения задач, в особенности для Грид-систем. Многие из этих элементов являются языками и протоколами, которые уже отсутствуют, либо только разрабатываются в настоящее время, либо требуют создания спецификаций и проведения стандартизации.

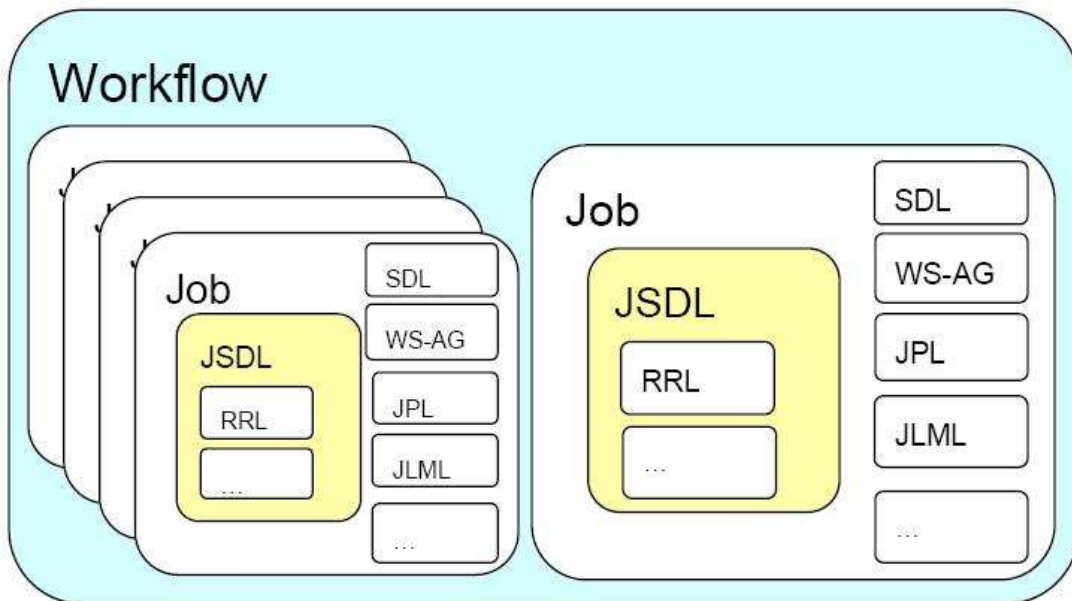


Рис.2. Элементы потока выполнения задач.

Раздел 5 содержит 5 подразделов.

В п.5.1 речь идет о языке описания потребностей в ресурсах (RRL). Спецификация JSDL могла бы использовать язык для описания потребностей задачи в ресурсах (RRL). Определение RRL должно быть дано в своей спецификации, и поэтому выходит за рамки спецификации JSDL. Однако, в отсутствие подходящего языка RRL, здесь задается основной набор элементов,

которые должны содержаться в RRL. Вполне вероятно, что будущие версии спецификации JSDL будут использовать стандартные спецификации RRL, если они будут доступны.

В п.5.2 описаны задачи, решаемые с использованием языка описания планирования (SDL). Планирование задач и описание требований планирования являются комплексными проблемами. Требования планирования распределяются по многим категориям. Они включают:

- Планирование времени. Время начала и окончания задачи, срок действия окон для исполнения задачи и т.д.
- Расписание, зависящее от данных. Планирование задач в зависимости от ожидаемого наличия входных данных и т.д.
- Планирование, зависящее от рабочего потока. Планирование задач в зависимости от порядка задач в рабочем потоке.

Кроме того, могут существовать зависимости между этими категориями в результате планирования требуемых ресурсов задач. Например, может существовать зависимость между заданным порядком задач в рабочем потоке и входными данными, необходимыми для выполнения запланированной задачи. Из-за сложности описания требований расписания, было принято решение вынести требования расписания за рамки JSDL. Следует, однако, отдавать себе отчет о важности языка описания расписания для размещения задачи на соответствующих ресурсах, управления и исполнения задач в таких потенциально сложных системах, как Грид-системы.

В п.5.3 описано соглашение о WEB-сервисах (WS-AG). Разработка соглашения между ресурсными WEB сервисами переднего плана и WEB сервисами представления задач в настоящее время лежат в рамках спецификации GGF WS-соглашения (WS-AG).

Мы рассматриваем WS-соглашения как важный новый стандарт представления ресурсов задачи. Однако, многие используемые в настоящий момент системы не имеют WEB-сервисов переднего плана и могут не требовать такого рода функциональности. Таким образом, важно, чтобы JSDL мог бы

быть использован как совместно с другими стандартами, так и самостоятельно. Другими словами, JSDL может быть использован как с WS-AG, так и без него.

Задачи, решаемые с использованием языка стратегии задач (JPL), приведены в п.5.4. Стратегия задач является очень широкой темой. Стратегии могут применяться во многих аспектах выполнения задачи. Пользователи могут хотеть применять стратегии для задач, собственники ресурсов также могут хотеть использовать стратегии для использования задачами своих ресурсов. Описание таких ограничений в стратегиях требует своего собственного языка описания и лежит вне рамок JSDL. Язык описания стратегий может рассматриваться как родственный JSDL.

В п.5.5 дано обоснование необходимости разработки языка управления жизненным циклом задач (JLML). Управление жизненным циклом задач является важным аспектом общего управления задачами. Это управление позволяет описывать состояния задач и директивы управления задачами. Мы видим необходимость разработки языка работы управления жизненным циклом задач, который мог бы описывать этот аспект управления задачами. Так как управление жизненным циклом задач связано с работой после представления задачи, язык управления жизненным циклом задач выходит за рамки JSDL.

Возможность применения JSDL в организации потока задач описана в п.5.6. Язык JSDL предназначен для описания шаблонов задач при представлении отдельных задач на соответствующих вычислительных ресурсах. Следует учитывать, что существует большая заинтересованность в развитии языков для описания параметрических задач, совместного распределения на вычислительных ресурсах, потоков задач и т.д.. Такие языки могут «знать о существовании JSDL», другими словами, они могут ссылаться на описания отдельных задач, выполненных на языке JSDL, для каждого рабочего узла. Мы видим, однако, что определение такого языка представляет собой отдельное направление деятельности, и может быть создана надстройка к стандарту JSDL и, следовательно, выходят за рамки JSDL.

Раздел 6 «Структура документа JSDL» содержит описание структуры документа, написанного на языке JSDL. Документ JSDL описывается с использованием языка XML и связан с нормативной схемой XML, содержащейся в Приложении А.

Документ JSDL организован следующим образом: корневой элемент JobDefinition содержит единственный обязательный дочерний элемент JobDescription. Элемент JobDescription содержит элементы, описывающие задачу: JobIdentification, Application, Resources, DataStaging. Определение псевдо-схемы следующее:

```
<JobDefinition>
  <JobDescription>
    <JobIdentification ... />?
    <Application ... />?
    <Resources ... />?
    <DataStaging ... />*
  </JobDescription>
  <xsd:any##other/>*
</JobDefinition>
```

Полные примеры документов JSDL приведены в Приложении В.

Спецификация JSDL не определяет значения по умолчанию для элементов, которые отсутствуют в документе JSDL. Типичные значения для таких элементов присваиваются системой потребления задач (потребляющей системой). Для получения более подробной информации следует обратиться к спецификации каждого элемента. Предполагается, что элементы JSDL, требующие описания для конкретного представления задачи, будут представлены совместно с соответствующими значениями в документе JSDL.

Все элементы, присутствующие в документе JSDL, обязаны быть выполнены для всего документа, который будет удовлетворен потребляющей системой. Конечно, существует возможность создания документов JSDL,

которые не могут быть удовлетворены. Например, документ JSDL, который содержит противоречивые потребности в ресурсах, не может быть удовлетворен. Однако следует отметить, что результат обработки документа JSDL потребляющей системой спецификацией JSDL не рассматривается.

В разделе 7 «Типы элементов JSDL» отмечается, что спецификация JSDL использует некоторое количество типов нормативных XML-схем. Она также использует ряд типов, специфичных для описания потребностей задач. Группа JSDL-WG основывается на определении ряда типов, приведенных в уже существующих стандартах, в частности, стандарте Общей информационной модели (CIM) и Интерфейсе переносимых операционных систем (POSIX). Поскольку не существует определения нормативных XML-схем для этих типов, одна из них представлена в этом документе. Типы операционных систем определены, основываясь на CIM. Некоторое число типов определяют специальное значение “other”. Это значение может быть использовано для введения элементов из других спецификаций.

В разделе 8 «Базовый набор элементов JSDL» описывается базовый набор элементов JSDL, который содержит семантику элементов, определенных в спецификации JSDL 1.0. Все элементы должны поддерживаться потребляющими системами, удовлетворяющими требованиям JSDL 1.0.

Всего в базовый набор входят 28 элементов, объединенных в ряд групп. Каждый элемент описывается единым образом, это описание включает: определение, множественность, тип, атрибуты, псевдо-схему и примеры.

В Разделе 9 «Расширения JSDL» подчеркивается, что язык JSDL задает общую структуру для определения требований размещения задач. Эта структура может быть расширена для обеспечения лучшего соответствия специфическим потребностям. Язык JSDL обеспечивает два механизма расширения: использование атрибутов и использование элементов. В общем случае, использование любого механизма расширения будет ограничивать интероперабельность, поэтому эти механизмы должны использоваться расчетливо и только в случае необходимости. Разработчикам,

заинтересованным в расширении JSDL, рекомендуется вначале убедиться в том, что никакая другая группа уже не создала расширение, которое уже учитывает выдвигаемые требования.

Если элементы или атрибуты, определенные посредством расширений, существуют в документе JSDL, они должны поддерживаться таким же способом, как и обычные элементы или атрибуты JSDL. Если элементы или атрибуты, определенные посредством расширений, существуют в документе JSDL, они должны быть удовлетворены для документа JSDL, подлежащему удовлетворению потребляющей системой в целом.

Как уже отмечалось, результат размещения документа JSDL на потребляющей системе не описывается данной спецификацией.

В Разделе 10 «Нормативные расширения JSDL» сказано, что нормативно определенным расширением в JSDL 1.0. является выполнение задач на POSIX-совместимых узлах.

Это нормативное расширение определяет схему, которая описывает приложения, исполняемые на узлах, удовлетворяющих требованиям POSIX. “jsdl-posix” - префикс пространства имен, используемый для данной схемы в данной спецификации.

В Разделе 11 «Вопросы безопасности» отмечается, что данная спецификация определяет язык для описания требований вычислительной задачи при ее представлении (размещении) в Грид-системе или какой-то другой системе управления выполнением задач. Предполагается, что размещение задачи должно быть безопасным, но механизм обеспечения безопасности лежит вне данной спецификации. Возможность описания прав, необходимых для выполнения задачи, является очень важной. Ожидается, что документы JSDL версии 1.0 будут, если это требуется, объединены с более специализированными языками описания политик и безопасности для четкого делегирования прав.

Другие вопросы безопасности, которые являются специфическими для определенных в данной спецификации элементов, обсуждаются подробно в каждом разделе, где описаны эти элементы.

Обязательными считаются: Приложение А, «Нормативная схема XML», Приложение Б, «Нормативная схема JSDL для приложений POSIX» и Приложение В «Нормативная схема JSDL, содержащая сводку исправлений». Приложение Г «Расширенные информационные примеры JSDL.» считается информационным.

Направления дальнейших работ

Разработанный авторами документ представляет собой первую редакцию стандарта. Так как статус национального стандарта придает Росстандарт, для придания разработанному проекту такого статуса предстоит провести целый ряд действий, определяемых в ГОСТ Р 1.2-2004 «Правила разработки, утверждения, обновления и отмены» [6].

ИРЭ им. В.А.Котельникова РАН с 2010 г. участвует в работе ТК22 «Информационные технологии», ведёт ПК125 «Взаимосвязь оборудования для информационных технологий», поэтому утверждение стандарта целесообразно осуществлять через этот ТК.

Процесс разработки и утверждения национального стандарта достаточно трудоёмкий и длительный – занимает время порядка 1 года. Очевидно, что когда возникнет необходимость создания профилей Грид-среды, в которые войдут многие стандарты, трудоёмкость процесса многократно вырастет.

По-видимому, представляется целесообразным пойти другим путём, воспользовавшись поправкой к ФЗ «О техническом регулировании» принятой 30.12.2009 г., согласно которой к документам в области стандартизации, используемым на территории Российской Федерации, относятся и надлежащим образом заверенные переводы на русский язык международных стандартов (ст.13).

Фактически, в данном случае идёт речь о создании русскоязычной версии стандартов OGF. При этом, конечно же, важно участвовать в работе OGF, и определить очередность создания переводов стандартов, т.е. создания основных направлений разработки Грид-стандартов (так называемой «дорожной карты») для российских специалистов и выработки единого глоссария для Грид-тематики. Все эти работы требуют, безусловно, коллективного участия всех заинтересованных специалистов.

Заключение.

На основании изложенного можно сделать следующие выводы:

1. Постепенный переход от фундаментальных исследований в области Грид-технологий к прикладным и создание национального сегмента мировой Грид-среды требует разработки национальных стандартов и профилей стандартов, гармонизированных с международными.
2. Первоочередным стандартом для обеспечения интероперабельности должен выступить стандарт на язык описания представления задач в GRID –среде.

Проект такого стандарта, гармонизированный со стандартом на язык JSDL, который создан международной организацией Open Grid Forum, разработан авторами статьи. В настоящее время происходит процесс придания проекту статуса национального стандарта.

Этот документ войдет в первоочередной профиль стандартов Грид-среды. Дальнейшие работы требуют создания нормативного документа «Дорожная карта по разработке стандартов GRID» ((Roadmap), документа, содержащего единый глоссарий для всех стандартов Грид.

Очевидно, что масштаб планируемых работ требует значительного увеличения финансирования и активного сотрудничества с международным сообществом.

Литература

1. <http://www.ogf.org/>
2. <http://www.consultant.ru/popular/techreg/>
3. <http://www.omii.ac.uk/attach/Presentations/781.pdf>
4. Журавлев Е.Е., Корниенко В.Н., Олейников А.Я. Вопросы стандартизации и обеспечения интероперабельности в GRID-системах. //Распределенные вычисления и Грид-технологии в науке и образовании: Труды 4-й междунар. конф. (Дубна, 28 июня – 3 июля, 2010 г.).- Дубна: ОИЯИ, Д-11-2010-140, 2010.- с. 364-372 ISBN 978-5-9530-0269-1
5. <http://netler.ru/pc/cloud.htm>
6. Вопросы выбора первоочередных русскоязычных стандартов информационных технологий. А. А. Каменщиков, А. Н. Кочуков, А. Я. Олейников, Т. Д. Широбокова. Журнал радиоэлектроники, № 9, сентябрь 2008 г. <http://jre.cplire.ru/jre/sep08/2/text.html>.